

# INFORMATYKA

## **Program nauczania informatyki dla drugiego etapu edukacyjnego InstaKod Programowanie od podstaw**

Program zgodny z

Rozporządzeniem Ministra Edukacji Narodowej z dnia 14 lutego 2017 r.

w sprawie podstawy programowej wychowania przedszkolnego oraz podstawy programowej kształcenia ogólnego dla szkoły podstawowej, w tym dla uczniów z niepełnosprawnością intelektualną w stopniu umiarkowanym lub znacznym, kształcenia ogólnego dla branżowej szkoły I stopnia, kształcenia ogólnego dla szkoły specjalnej przysposabiającej do pracy oraz kształcenia ogólnego dla szkoły policealnej

instaKod

PROGRAMOWANIE OD PODSTAW

**SPIS TREŚCI**

UWAGI WSTĘPNE .....	4
REALIZACJA TREŚCI PODSTAWY PROGRAMOWEJ W KLASACH 4-6.....	6
CELE EDUKACYJNE W KLASACH 4-6.....	8
SZCZEGÓŁOWE CELE WYCHOWAWCZE:.....	8
SZCZEGÓŁOWE CELE KSZTAŁCENIA: .....	9
MATERIAŁ NAUCZANIA W KLASACH 4-6 .....	15
KLASA 4.....	15
KLASA 5.....	18
KLASA 6.....	22
ROZKŁAD MATERIAŁU W KLASACH 4-6 .....	25
RAMOWY ROZKŁAD MATERIAŁU - KLASA 4 .....	25
RAMOWY ROZKŁAD MATERIAŁU - KLASA 5 .....	26
RAMOWY ROZKŁAD MATERIAŁU - KLASA 6 .....	26
SZCZEGÓŁOWY ROZKŁAD MATERIAŁU -- KLASA 4.....	27
SZCZEGÓŁOWY ROZKŁAD MATERIAŁU – KLASA 5.....	28
SZCZEGÓŁOWY ROZKŁAD MATERIAŁU – KLASA 6.....	30
SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA.....	32
KLASA 4.....	33
KLASA 5.....	37
KLASA 6.....	42
OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA .....	47
KLASA 4.....	48
KLASA 5.....	56
KLASA 6.....	66
REALIZACJA TREŚCI PODSTAWY PROGRAMOWEJ W KLASACH 7-8.....	74
CELE EDUKACYJNE W KLASACH 7-8.....	76
SZCZEGÓŁOWE CELE WYCHOWAWCZE:.....	76
SZCZEGÓŁOWE CELE EDUKACYJNE: .....	77
MATERIAŁ NAUCZANIA W KLASACH 7-8 .....	82
KLASA 7.....	82
KLASA 8.....	86

ROZKŁAD MATERIAŁU W KLASACH 7-8 .....	89
RAMOWY ROZKŁAD MATERIAŁU KLASA 7.....	89
RAMOWY ROZKŁAD MATERIAŁU KLASA 8.....	90
SZCZEGÓŁOWY ROZKŁAD MATERIAŁU KLASA 7 .....	91
SZCZEGÓŁOWY ROZKŁAD MATERIAŁU KLASA 8.....	93
SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA.....	95
KLASA 7 .....	96
KLASA 8.....	101
OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA.....	107
KLASA 7.....	108
KLASA 8.....	116
METODY PRACY NA LEKCJACH .....	124
DODATEK A. LISTA ALGORYTMÓW REALIZOWANYCH W KLASACH 4-6.....	142
DODATEK B. LISTA ALGORYTMÓW REALIZOWANYCH W KLASACH 7-8.....	142
DODATEK C. LISTA INSTRUKCJI W JĘZYKU ASSEMBLY DLA KLAS 4-6.....	142
DODATEK D. LISTA INSTRUKCJI W JĘZYKU NIANIOLANG DLA KLAS 7-8 .....	14234
AUTORZY .....	142

### UWAGI WSTĘPNE

We współczesnym świecie charakteryzującym się szybkim postępem technologicznym, podstawy programowania na równi z arytmetyką stanowią niezbędną część wykształcenia. Nowa podstawa programowa dla II etapu edukacyjnego definiuje wymagany zakres pojęć programistycznych oraz umiejętności technicznych. Opanowanie wspomnianych wymagań konieczne jest by sprostać oczekiwaniom dla dalszych etapów edukacji. Nabyta wiedza oraz umiejętności przygotowują uczniów do świadomego korzystania z technologii oraz urządzeń cyfrowych w życiu codziennym.

**InstaKod** jest programem zgodnym z obowiązującą podstawą programową dla II etapu edukacyjnego. Może być realizowany jako kontynuacja dowolnego programu zgodnego z podstawą programową dla I etapu edukacyjnego, uwzględnia bowiem wszystkie zmiany wprowadzone przez najnowszą podstawę programową w nauczaniu informatyki w klasach I – III szkoły podstawowej.

Materiał programu został podzielony na trzy uzupełniające się działy:

1. wprowadzenie, w skład którego wchodzi zagadnienia z zakresu przestrzegania prawa i zasad bezpieczeństwa korzystania z technologii informacyjnych i informacji dostępnych w sieci,
2. podstawy programowania, w szczególności proces rozumienia, analizowania i rozwiązywania problemów (myślenia komputacyjnego),
3. technologie informacyjne wspierające naukę oraz ich zastosowanie do realizacji projektów z innych przedmiotów szkolnych.

Dzieląc materiał podstawy programowej na poszczególne lata, każdy rok szkolny w klasach 4-6 podzielono pół na pół między podstawy programowania oraz pozostałe dwie części. W klasach 7-8 każdy rok szkolny podzielono w stosunku 2/3 do 1/3. Taki podział zapewnia uczniom wystarczająco dużo czasu na zbudowanie solidnej podstawy w rozumieniu zagadnień związanych z programowaniem. Dzięki temu łatwiej będzie im sprostać wymaganiom z zakresu informatyki w kolejnym etapie edukacyjnym.

Program ułożono zgodnie z zasadą spiralności, różnicowania i zwiększania poziomu trudności zadań małymi krokami. Praktyczne zadania z zakresu podstaw programowania dostosowane zostały do różnych możliwości percepcyjnych i poziomu intelektualnego uczniów. Zadania w każdej lekcji zróżnicowane są poziomami trudności. Dzięki temu każdy uczeń może ćwiczyć nowo nabyte umiejętności na poziomie, odpowiadającym jego możliwościom w tej dziedzinie. Platforma internetowa umożliwi uczniom zdolnym pracę w szybszym tempie i dalszy rozwój m.in. w kierunku przygotowania do olimpiady informatycznej. Nauczyciel posiada opcję udostępnienia uczniowi zdań oraz materiałów dydaktycznych z wyższych poziomów.

Program stanowi część zestawu materiałów wspierających nauczyciela i ucznia, w skład których wchodzi: **podręczniki, zeszyty ćwiczeń, platforma edukacyjna oraz poradniki metodyczne**. Platforma edukacyjna InstaKod dostępna jest pod adresem [instakod.pl](http://instakod.pl). Nauczyciel na swoim **koncie** zakłada i zarządza **kontami uczniów, mając pełny wgląd w wyniki ich pracy**. Ewaluuje postępy uczniów za pomocą quizów oraz sprawdzianów on-line. W ramach wsparcia dla nauczycieli na platformie dostępne są również **szkolenia on-line** z nawigacji po platformie, wybranych języków programowania oraz metodyki nauczania.

W klasach 4-6 język wybrany do nauki podstaw programowania to Assembly. Jest językiem wizualnym, z założenia powstałym do celów edukacyjnych. Składa się jedynie z 10 instrukcji, wystarczających do wprowadzenia pojęć wejścia i wyjścia, operacji arytmetycznych, instrukcji warunkowej oraz pętli.

W klasach 7-8 język wybrany do nauki podstaw programowania to Nianiolang. Język ten jest językiem tekstowym. Biblioteka dostępnych w nim instrukcji ograniczona została do 50, co umożliwi skupienie się na istocie ćwiczonych pojęć. Zgodnie z zasadą spiralności, pojęcia programistyczne poznane i przećwiczone na etapie nauczania w klasach 4-6 w języku wizualnym są powtarzane i ćwiczone w języku tekstowym w klasach 7-8. Pojęcia rozbudowane są o tablice, napisy i funkcje. Uczniowie zdolni, po ukończeniu programu klas 7-8 są gotowi, by głębiej poznać algorytmikę i z łatwością przechodzą do programowania np. w języku C++ (jeden z języków Olimpiady Informatycznej).

Nauka technologii informacyjnych odbywa się dwutorowo. Funkcje poszczególnych aplikacji są omawiane i ćwiczone przy pomocy krótkich zadań. Następnie nowo nabyte umiejętności są praktycznie stosowane do realizacji projektów międzyprzedmiotowych. Projekty oceniane są z technicznego punktu widzenia na zajęciach informatyki i z merytorycznego punktu widzenia na przedmiocie, w którym realizowany jest projekt.

Wybór aplikacji dokonany został tak, by każdy uczeń był w stanie korzystać z nich w domu, niezależnie od tego czy ma dostęp do płatnego oprogramowania czy nie. Wybrane funkcje aplikacji prezentowane są w podręcznikach na przykładzie m.in. pakietu MS Office i darmowego oprogramowania (m.in. Libre Office oraz innych dostępnych online). Uczniowie nabywają umiejętności obsługi kilku aplikacji o zbliżonym zastosowaniu, poznając ich wady i zalety.

Dostęp do platformy InstaKod jest nieodpłatny dla wszystkich nauczycieli i uczniów realizujących zajęcia informatyki w oparciu o program i podręczniki InstaKod.

InstaKod Sp z o.o. jest właścicielem platformy instakod.pl oraz programu i uzupełniających materiałów. Publikując program na stronie [www.instakod.pl](http://www.instakod.pl), wyraża tym samym zgodę na bezpłatne wykorzystanie przez nauczycieli niniejszego programu do pracy z uczniami.

InstaKod Sp z o.o. wyraża także zgodę na tworzenie przez nauczycieli autorskich programów nauczania w oparciu o program nauczania *InstaKod* pod warunkiem, że w przygotowanym materiale zostanie zapisana informacja, iż powstał on na podstawie programu *InstaKod*.

## REALIZACJA TREŚCI PODSTAWY PROGRAMOWEJ W KLASACH 4-6

W tabeli przedstawiono informacje, w których klasach według programu *InstaKod* realizowane są poszczególne treści podstawy programowej.

Treści nauczania według podstawy programowej	Klasa		
	4	5	6
<b>I. Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:</b>			
1) Tworzy i porządkuje w postaci sekwencji (liniowo) lub drzewa (nieliniowo) informacje, takie jak:			
a. obrazki i teksty ilustrujące wybrane sytuacje;	+	+	+
b. obiekty z uwzględnieniem ich cech charakterystycznych.	+	+	+
2) Formułuje i zapisuje w postaci algorytmów, polecenia składające się na:			
a. rozwiązywanie problemów z życia codziennego i z różnych przedmiotów, np. Liczenie średniej, pisemne wykonanie działań arytmetycznych, takich jak dodawanie i odejmowanie;	+	+	+
b. osiągnięcie postawionego celu, w tym znalezienie elementu w zbiorze nieuporządkowanym lub uporządkowanym, znalezienie elementu najmniejszego i największego;	+	+	+
c. sterowanie robotem lub obiektem na ekranie.		+	
3) W algorytmicznym rozwiązywaniu problemu wyróżnia podstawowe kroki: określenie problemu, i celu do osiągnięcia, analiza sytuacji problemowej, opracowanie rozwiązania, sprawdzenie rozwiązania problemu dla przykładowych danych, zapisanie rozwiązania w postaci schematu lub programu.	+	+	+
<b>II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:</b>			
1) Projektuje, tworzy i zapisuje w wizualnym języku programowania:			
a. pomysły historyjek i rozwiązania problemów, w tym proste algorytmy z wykorzystaniem poleceń sekwencyjnych, warunkowych i iteracyjnych oraz zdarzeń jednoczesnych;	+	+	+
b. prosty program sterujący robotem lub innym obiektem na ekranie komputera.	+		
Testuje na komputerze swoje programy pod względem zgodności z przyjętymi założeniami i ewentualnie je poprawia, objaśnia przebieg działania programów.	+	+	+
2) Przygotowuje i prezentuje rozwiązania problemów posługując się podstawowymi aplikacjami (edytor tekstu oraz grafiki, arkusz kalkulacyjny, program do tworzenia prezentacji multimedialnej), na swoim komputerze lub w chmurze, wykazując się przy tym umiejętnościami:			
a. tworzenia ilustracji w edytorze grafiki: rysuje za pomocą wybranych narzędzi, przekształca obrazy, uzupełnia grafikę tekstem;	+	+	+
b. tworzenia dokumentów tekstowych: dobiera czcionkę, formatuje akapity, wstawia do tekstu ilustracje, napisy i kształty, tworzy tabele oraz listy numerowane i punktowane;	+	+	+
c. korzystania z arkusza kalkulacyjnego w trakcie rozwiązywania zadań związanych z prostymi obliczeniami: wprowadza dane do arkusza, formatuje komórki, definiuje proste formuły i dobiera wykresy do danych i celów obliczeń;	+	+	+
d. tworzenia krótkich prezentacji multimedialnych łączących tekst z grafiką, korzysta przy tym z gotowych szablonów lub projektuje według własnych pomysłów.	+	+	+
3) Gromadzi, porządkuje i selekcjonuje efekty swojej pracy oraz potrzebne zasoby w komputerze lub w innych urządzeniach oraz w środowiskach wirtualnych (w chmurze).		+	+

<b>III. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi. Uczeń:</b>			
1) Opisuje funkcje podstawowych elementów komputera i urządzeń zewnętrznych oraz:			
a. korzysta z urządzeń do nagrywania obrazów, dźwięków i filmów, w tym urządzeń mobilnych, takich jak, telefony komórkowe i aparaty fotograficzne;		+	+
b. wykorzystuje komputer lub inne urządzenie cyfrowe do gromadzenia, porządkowania i selekcjonowania własnych zasobów.		+	+
2) Wykorzystuje sieć komputerową (szkolną, internet):			
a. do wyszukiwania potrzebnych informacji i zasobów edukacyjnych nawigując między stronami;	+	+	+
b. jako medium komunikacyjne;		+	+
c. do pracy w wirtualnym środowisku (na platformie, w chmurze), stosując się do sposobów i zasad pracy w takim środowisku.	+	+	+
3) Organizuje w folderach swoje pliki zgromadzone w komputerze.	+	+	+
<b>IV. Rozwijanie kompetencji społecznych. Uczeń:</b>			
1) Uczestniczy w zespołowym rozwiązaniu problemu posługując się technologią taką jak: poczta elektroniczna, forum, wirtualne środowisko kształcenia, dedykowany portal edukacyjny.	+	+	+
2) Identyfikuje i docenia korzyści płynące ze współpracy nad wspólnym rozwiązywaniem problemów.	+	+	+
3) Respektuje zasadę równości w dostępie do technologii i do informacji, w tym w dostępie do komputerów w społeczności szkolnej.	+	+	+
4) Określa zawody i wymienia przykłady z życia codziennego, w których są wykorzystywane kompetencje informatyczne.	+	+	+
<b>V. Prawo i zasady bezpieczeństwa. Uczeń:</b>			
1) Rozumie, że niewłaściwe postępowanie w posługiwaniu się technologią i informacją rodzi negatywne konsekwencje.	+	+	+
2) Uznaje i respektuje prawo do prywatności danych i informacji oraz prawo do własności intelektualnej.	+	+	+
3) Zauważa zagrożenia związane z powszechnym dostępem do technologii oraz do informacji i wystrzega się ich.	+	+	+
4) Rozumie znaczenie profilaktyki antywirusowej i konieczność ochrony przed innymi zagrożeniami dla komputerów i informacji.	+	+	+

## CELE EDUKACYJNE W KLASACH 4-6

### SZCZEGÓŁOWE CELE WYCHOWAWCZE:

Informatyka poza kształceniem umiejętności technicznych pobudza uczniów do myślenia abstrakcyjnego, do utrzymania koncentracji przez dłuższy okres czasu oraz ćwiczenia czytania ze zrozumieniem. Świadomy nauczyciel informatyki rozumie wieloaspektowość wychowania informatycznego i stara się odzwierciedlić to w swoim nauczaniu, realizując wymienione poniżej cele wychowawcze..

- Ćwiczenie pamięci.
- Rozwijanie myślenia abstrakcyjnego.
- Rozwijanie dociekliwości.
- Ćwiczenie umiejętności czytania ze zrozumieniem.
- Rozwijanie umiejętności analizowania problemu i wyszukiwania kluczowych informacji.
- Ćwiczenie spójnego i precyzyjnego wyrażania myśli.
- Rozwijanie zainteresowań informatycznych.
- Kształtowanie umiejętności rozwiązywania zadań programistycznych.
- Kształcenie umiejętności klarownego przedstawiania swojego toku rozumowania.
- Rozwijanie potrzeby sprawdzania poprawności swojego toku rozumowania, wyciągania wniosków i poprawiania błędów.
- Wyrabianie potrzeby eksperymentowania z nowymi pojęciami.
- Rozwijanie samodzielności w zbieraniu informacji o danym temacie.
- Kształcenie nawyku systematycznej i zorganizowanej pracy.
- Rozwijanie potrzeby pracy w ciszy i skupieniu.
- Kształcenie kompetencji społecznych, w tym pracy zespołowej.
- Poszanowanie pracy innych.



SZCZEGÓŁOWE CELE KSZTAŁCENIA:

**KLASA 4**

**I. Wprowadzenie**

- 1) Prawo i zasady bezpieczeństwa
  - Poznanie zasad prawidłowego zachowania w pracowni komputerowej.
  - Poznanie zasad tworzenia i bezpiecznego korzystania z haseł.
  - Poznanie zasad wykorzystania internetu.
  - Poznanie zasad bezpiecznego korzystania z internetu.
- 2) System binarny
  - Rozumienie fundamentalnego znaczenia liczb w informatyce.
  - Rozwijanie umiejętności posługiwania się systemem binarnym.

**II. Kształtowanie pojęć informatycznych i rozwijanie umiejętności posługiwania się nimi**

- 1) Wprowadzenie do programowania
  - Poznanie zasad tworzenia programu komputerowego przy użyciu edytora wizualnego.
  - Pisanie prostego programu przy użyciu wizualnego języka programowania.
  - Rozumienie pojęć wejścia i wyjścia z programu.
  - Posługiwanie się instrukcją wyjścia w celu wyświetlenia na konsoli komunikatów.
  - Posługiwanie się instrukcją wejścia w celu pobrania danych od użytkownika.
  - Kształtowanie pojęcia zmiennej oraz świadomości stanu wartości zmiennych w danej chwili.
  - Rozumienie różnicy między wyjściem stałym i zmiennym.
- 2) Wprowadzenie do algorytmów
  - Rozwijanie intuicji związanych z podstawowymi pojęciami informatyki: kolejność liniowa, instrukcja, algorytm.
  - Analiza problemu, opracowanie rozwiązania, przetestowanie rozwiązania dla przykładowych danych.
  - Implementacja rozwiązania w języku wizualnym i sprawdzanie rozwiązania problemu dla przykładowych danych.
  - Analiza błędów.
  - Objaśnianie przebiegu działania programów.
  - Rysowanie diagramów rozwiązań.
  - Rozumienie różnicy między algorytmami liniowymi i algorytmami z warunkami.
- 3) Operacje arytmetyczne na zmiennych
  - Poznanie sposobów rozwiązywania zadań arytmetycznych z wykorzystaniem komputera.
  - Stosowanie w programach instrukcji służących do wykonywania prostych operacji arytmetycznych na zmiennych. Dodawanie i odejmowanie liczb całkowitych.
  - Rozumienie i używanie pojęć związanych z arytmetyką: suma i różnica.
  - Rozumienie potrzeby użycia zmiennej pomocniczej.
  - **Implementacja algorytmu znajdowania sumy i różnicy dla dwóch i trzech zmiennych.**
- 4) Instrukcja warunkowa
  - Zapisywanie i ocena prawdziwości wyrażeń logicznych.
  - Wykorzystanie instrukcji warunkowej do podejmowania decyzji.
  - Implementacja w programie prostego algorytmu z warunkami.
  - **Implementacja algorytmów znajdowania maksimum i minimum z dwóch liczb.**

**III. Poznawanie technologii informacyjnych**

- 1) Tworzenie rysunków za pomocą edytora graficznego.
  - Posługiwanie się edytorem graficznym w celu stworzenia prostego rysunku.
  - Użycie edytora graficznego do przygotowania projektu międzyprzedmiotowego.
- 2) Tworzenie dokumentów za pomocą edytora tekstu.
  - Posługiwanie się edytorem tekstu w celu napisania poprawnego językowo i stylistycznie tekstu.
  - Stosowanie podstawowych zasad formatowania i redagowania tekstu.
  - Użycie edytora tekstu do przygotowania projektu międzyprzedmiotowego.
- 3) Tworzenie prezentacji multimedialnych
  - Posługiwanie się programem do tworzenia prezentacji w celu stworzenia estetycznego i interesującego przekazu informacji.
  - Użycie programu do tworzenia prezentacji do przygotowania projektu międzyprzedmiotowego.
- 4) Obliczenia w arkuszu kalkulacyjnym  
Poznanie podstawowych zasad wykonywania prostych obliczeń za pomocą arkusza kalkulacyjnego.

## KLASA 5

## I. Wprowadzenie

- 1) Prawo i zasady bezpieczeństwa
  - Poznanie podstawowych elementów komputera.
  - Poznanie funkcji systemu operacyjnego.
  - Poznanie pojęcia piractwa komputerowego.
- 2) System binarny
  - Rozumienie fundamentalnego znaczenia liczb w informatyce.
  - Rozwijanie umiejętności posługiwania się systemem binarnym.

## II. Kształtowanie pojęć informatycznych i rozwijanie umiejętności posługiwania się nimi

- 1) Wprowadzenie do programowania
  - Rozwijanie intuicji związanej z pojęciami informatycznymi poznanymi w klasie czwartej.
  - Poznanie zasad tworzenia programu komputerowego przy użyciu edytora wizualnego.
  - Pisanie prostego programu przy użyciu wizualnego języka programowania.
  - Rozumienie pojęć wejścia i wyjścia z programu.
  - Posługiwanie się instrukcją wyjścia w celu wyświetlenia na konsoli komunikatów.
  - Posługiwanie się instrukcją wejścia w celu pobrania danych od użytkownika.
  - Kształtowanie pojęcia zmiennej oraz świadomości stanu wartości zmiennych w danej chwili.
  - Rozumienie różnicy między wyjściem stałym i zmiennym.
- 2) Wprowadzenie do algorytmów
  - Rozwijanie intuicji związanej z podstawowymi pojęciami informatyki: kolejność liniowa, instrukcja, algorytm.
  - Analiza problemu, opracowanie rozwiązania, przetestowanie rozwiązania dla przykładowych danych.
  - Implementacja rozwiązania w języku wizualnym i sprawdzanie rozwiązania problemu dla przykładowych danych.
  - Analiza błędów.
  - Objaśnianie przebiegu działania programów.
  - Rysowanie diagramów rozwiązań.
  - Rozumienie różnicy między algorytmami liniowymi i algorytmami z warunkami.
- 3) Operacje arytmetyczne na zmiennych
  - Poznanie sposobów rozwiązywania zadań arytmetycznych z wykorzystaniem komputera.
  - Stosowanie w programach instrukcji służących do wykonywania prostych operacji arytmetycznych na zmiennych. Dodawanie i odejmowanie liczb całkowitych.
  - Rozumienie i używanie pojęć związanych z arytmetyką: suma i różnica.
  - Rozumienie potrzeby użycia zmiennej pomocniczej.
  - **Implementacja algorytmu znajdowania sumy i różnicy dla dwóch i trzech zmiennych.**
- 4) Instrukcja warunkowa
  - Zapisywanie i ocena prawdziwości wyrażeń logicznych.
  - Wykorzystanie instrukcji warunkowej do podejmowania decyzji.
  - Implementacja w programie prostego algorytmu z warunkami.
  - **Implementacja algorytmów znajdowania maksimum i minimum z dwóch liczb.**
- 5) Pętla pojedyncza
  - Implementacja pętli pojedynczej przy użyciu instrukcji warunkowej.
  - Stosowanie pętli do wypisywania wzorów na konsolę oraz modyfikacji pobranych od użytkownika danych.

**III. Poznawanie technologii informacyjnych**

- 1) Tworzenie rysunków za pomocą edytora graficznego.
  - Posługiwanie się edytorem graficznym w celu przetwarzania i obróbki zdjęć.
  - Użycie edytora graficznego do przygotowania projektu międzyprzedmiotowego.
- 2) Tworzenie dokumentów za pomocą edytora tekstu.
  - Umiejętność opracowania tekstów w edytorze tekstu.
  - Stosowanie podstawowych zasad formatowania i redagowania tekstu.
  - Użycie edytora tekstu do przygotowania projektu międzyprzedmiotowego.
- 3) Obliczenia w arkuszu kalkulacyjnym
  - Umiejętność prezentowania danych liczbowych w tabelach.
  - Użycie arkusza kalkulacyjnego do przygotowania projektu międzyprzedmiotowego.
- 4) Tworzenie prezentacji multimedialnych
  - Posługiwanie się programem do tworzenia prezentacji w celu stworzenia estetycznego i interesującego przekazu informacji.
  - Użycie programu do tworzenia prezentacji do przygotowania projektu międzyprzedmiotowego.
- 5) Tworzenie animacji w języku Scratch
  - Umiejętność tworzenia programów z wykorzystaniem sterowania duszkiem na ekranie komputera, w wizualnym języku programowania.
- 6) Dzielenie się plikami w Chmurze
  - Poznanie, czym jest chmura i jakie są jej zastosowania.
  - Korzystanie z chmury w celu wymiany plików.
- 7) Nagrywanie i edycja filmów
  - Korzystanie z urządzeń do nagrywania obrazów, dźwięków i filmów, w tym z urządzeń mobilnych.
  - Umiejętność edycji i montażu filmów.

## KLASA 6

## I. Wprowadzenie

- 1) Prawo i zasady bezpieczeństwa
  - Poznanie różnych zagrożeń dla komputera.
  - Poznanie zasad działania oprogramowania antywirusowego.
- 2) System binarny
  - Rozumienie fundamentalnego znaczenia liczb w informatyce.
  - Rozwijanie posługiwania się systemem binarnym.
  - Poznanie pojęć sortowania i grupowania liczb.
  - Wprowadzenie pojęcia przeszukiwania binarnego.

## II. Kształtowanie pojęć informatycznych i rozwijanie umiejętności posługiwania się nimi

- 1) Wprowadzenie do programowania
  - Rozwijanie intuicji związanych z pojęciami informatycznymi poznanymi w klasach czwartej i piątej.
  - Poznanie zasad tworzenia programu komputerowego przy użyciu edytora wizualnego.
  - Pisanie prostego programu przy użyciu wizualnego języka programowania.
  - Rozumienie pojęć wejścia i wyjścia z programu.
  - Posługiwanie się instrukcją wyjścia w celu wyświetlenia na konsoli komunikatów.
  - Posługiwanie się instrukcją wejścia w celu pobrania danych od użytkownika.
  - Kształtowanie pojęcia zmiennej oraz świadomości stanu wartości zmiennych w danej chwili.
  - Rozumienie różnicy między wyjściem stałym i zmiennym.
- 2) Wprowadzenie do algorytmów
  - Rozwijanie intuicji związanej z podstawowymi pojęciami informatyki: kolejność liniowa, instrukcja, algorytm.
  - Analiza problemu, opracowanie rozwiązania, przetestowanie rozwiązania dla przykładowych danych.
  - Implementacja rozwiązania w języku wizualnym i sprawdzanie rozwiązania problemu dla przykładowych danych.
  - Analiza błędów.
  - Objaśnianie przebiegu działania programów.
  - Rysowanie diagramów rozwiązań.
  - Rozumienie różnicy między algorytmami liniowymi i algorytmami z warunkami.
- 3) Operacje arytmetyczne na zmiennych
  - Poznanie sposobów rozwiązywania zadań arytmetycznych z wykorzystaniem komputera.
  - Stosowanie w programach instrukcji służących do wykonywania prostych operacji arytmetycznych na zmiennych. Dodawanie i odejmowanie liczb całkowitych.
  - Rozumienie i używanie pojęć związanych z arytmetyką: suma i różnica.
  - Rozumienie potrzeby użycia zmiennej pomocniczej.
  - **Implementacja algorytmu znajdowania sumy i różnicy dla dwóch i trzech zmiennych.**
  - **Implementacja algorytmu znajdowania wielokrotności z zapisu binarnego**
- 4) Instrukcja warunkowa oraz warunki zagnieżdżone
  - Zapisywanie i ocena prawdziwości wyrażeń logicznych.
  - Wykorzystanie instrukcji warunkowej do podejmowania decyzji.
  - Zapisywanie w postaci programu złożonego algorytmu z zagnieżdżonymi warunkami.
  - **Implementacja algorytmów znajdowania maksimum i minimum z dwóch liczb.**
  - **Implementacja algorytmów znajdowania maksimum i minimum z trzech liczb.**
  - **Implementacja algorytmu sortowania bąbelkowego trzech liczb.**
  - **Implementacja algorytmu znajdowania liczby środkowej z trzech liczb.**
  - Rozwiązywanie zadań tekstowych wymagających użycia instrukcji warunkowej.

- 5) Pętla pojedyncza oraz pętle podwójne
- Implementacja pętli pojedynczej przy użyciu instrukcji warunkowej.
  - Stosowanie pętli do wypisywania wzorów na konsolę oraz modyfikacji pobranych od użytkownika danych.
  - **Implementacja algorytmu znajdowania iloczynu dwóch liczb.**
  - **Implementacja algorytmu obliczania kwadratu liczby.**
  - **Implementacja algorytmu znajdowania ilorazu dwóch liczb.**
  - **Implementacja algorytmu obliczania średniej z trzech liczb.**
  - **Implementacja algorytmu znajdowania reszty z dzielenia.**
  - **Implementacja algorytmu sprawdzania parzystości danej liczby.**
  - Implementacja pętli zagnieżdżonej przy dwukrotnym użyciu instrukcji warunkowej.
  - Stosowanie pętli zagnieżdżonej do wypisywania na konsolę prostokątów i trójkątów złożonych z pojedynczych znaków.

### III. Poznawanie technologii informacyjnych

- 1) Tworzenie rysunków za pomocą edytora graficznego.
  - Posługiwanie się edytorem graficznym w celu przetwarzania i obróbki zdjęć.
  - Użycie edytora graficznego do przygotowania projektu międzyprzedmiotowego.
- 2) Tworzenie dokumentów za pomocą edytora tekstu.
  - Umiejętność opracowania tekstów w edytorze tekstu.
  - Stosowanie podstawowych zasad formatowania i redagowania tekstu.
  - Użycie edytora tekstu do przygotowania projektu międzyprzedmiotowego.
- 3) Dzielenie się plikami w Chmurze
  - Korzystanie z chmury w celu wymiany plików.
  - Umiejętność zespołowej pracy nad projektem.
- 4) Obliczenia w arkuszu kalkulacyjnym
  - Umiejętność korzystania z wbudowanych funkcji arkusza kalkulacyjnego.
  - Umiejętność prezentowania danych za pomocą różnych rodzajów wykresów.
  - Użycie arkusza kalkulacyjnego do przygotowania projektu międzyprzedmiotowego.
- 5) Tworzenie prezentacji multimedialnych
  - Posługiwanie się programem do tworzenia prezentacji w celu stworzenia estetycznego i interesującego przekazu informacji.
  - Użycie programu do tworzenia prezentacji do przygotowania projektu na lekcję historii.
- 6) Nagrywanie i edycja filmów
  - Korzystanie z urządzeń do nagrywania obrazów, dźwięków i filmów, w tym urządzeń mobilnych.
  - Umiejętność edycji i montażu filmów.
  - Użycie programu do edycji filmów do przygotowania projektu międzyprzedmiotowego.

## MATERIAŁ NAUCZANIA W KLASACH 4-6

Materiał nauczania na poziomie każdej klasy podzielony jest na 3 uzupełniające się działy:

- I. **wprowadzenie**, w skład którego wchodzi zagadnienia z zakresu przestrzegania prawa i bezpieczeństwa korzystania z technologii informacyjnych i informacji dostępnych w sieci,
- II. **podstawy programowania**, w szczególności proces rozumienia, analizowania i rozwiązywania problemów (myślenia komputacyjnego),
- III. **technologie informacyjne** wspierające naukę oraz ich zastosowanie do realizacji projektów z innych przedmiotów szkolnych.

Program został zbudowany w oparciu o zasadę spiralności. Treści, które uczniowie poznają w jednej klasie utrwalane i rozbudowywane są w klasie następnej. W ten sposób uczniowie posiadają głębsze rozumienie omawianych pojęć. Nauczyciele pracujący szybciej niż sugeruje to program, mogą bez problemu omawiać materiał z klasy następnej. Sugerowane jest przenoszenie uczniów najzdolniejszych do kolejnych poziomów klas po zrealizowaniu przez nich treści z poziomu ich klasy.

### KLASA 4

<b>I. Wprowadzenie</b>	
<b>Prawo i zasady bezpieczeństwa</b>	
Treści	Komentarze
Zasady pracy w pracowni komputerowej	Zasady pracy z komputerami. Regulamin szkolnej pracowni komputerowej.
Hasła	Pojęcie hasła. Zasady tworzenia bezpiecznego hasła. Zasady bezpiecznego korzystania z haseł.
Korzystanie z internetu	Pojęcie internetu. Możliwości i zagrożenia jakie niesie ze sobą powszechny dostęp do sieci.
Bezpieczeństwo w internecie	Wyjaśnienie potrzeby bezpiecznego korzystania z internetu. Zasady netykiety.
<b>System binarny</b>	
Treści	Komentarze
Systemy pozycyjne	Pojęcie systemu pozycyjnego na przykładzie systemów dziesiętnego i binarnego. Wyjaśnienie potrzeby użycia systemu binarnego w komputerach
Konwersja między systemami binarnym i dziesiętnym	Zapisywanie liczb naturalnych w zakresie do 32 w systemie binarnym. Odczytywanie liczb zapisanych w systemie binarnym.

II. Programowanie	
Wprowadzenie do programowania	
Treści	Komentarze
Zmienna	Wprowadzenie pojęcia zmiennej na przykładzie pudełka przechowującego liczby.
Wejście do programu	Wprowadzenie pojęcia wejścia do programu na przykładzie liczb pobieranych z klawiatury.
Stan programu	Wprowadzenie pojęcia stanu programu.
Wyjście stałe i zmienne	Wprowadzenie pojęcia wyjścia z programu na przykładzie konsoli. Rozróżnienie między wyjściem zmiennym i stałym.
Operacje arytmetyczne	
Treści	Komentarze
Dodawanie i odejmowanie stałych od zmiennej	Znajdowanie liczby, która jest od danej liczby o 3 większa, o 2 mniejsza, itd., dla jednej, dwóch i trzech liczb.
Suma i różnica dwóch zmiennych. Operacje arytmetyczne na dwóch i trzech zmiennych.	Obliczanie wartości prostych wyrażeń arytmetycznych. Rozbudowywanie rozumienia stanu programu w oparciu o zmienną pomocniczą. <b>Algorytm znajdowania sumy i różnicy dla dwóch i trzech zmiennych.</b>
Algorytm liniowy	Wprowadzenie do algorytmów liniowych.
Instrukcja warunkowa	
Treści	Komentarze
Operatory logiczne	Wprowadzenie sześciu operatorów relacji: równy =, różny ≠, większy >, większy bądź równy ≥, mniejszy < oraz mniejszy bądź równy ≤.
Porównywanie dwóch liczb	Przykłady porównań oraz ocena prawdziwości danego porównania.
Instrukcja warunkowa z porównaniem między zmienną a stałą	Uzależnienie wykonania instrukcji bądź bloku instrukcji od prawdziwości danego porównania między zmienną a stałą.
Instrukcja warunkowa z porównaniem między dwiema zmiennymi	Uzależnienie wykonania instrukcji bądź bloku instrukcji od prawdziwości danego porównania między zmienną a stałą. Porównanie dwóch zmiennych w celu sprawdzenia czy są równe. <b>Algorytm znajdowania minimum z dwóch liczb. Algorytm znajdowania maksimum z dwóch liczb.</b>
Algorytm z warunkami	Wprowadzenie pojęcia algorytmu z warunkami. Rozróżnienie między algorytmem liniowym i algorytmem z warunkami.



<b>III. Technologie informacyjne</b>	
<b>Edytor graficzny</b>	
Treści	Komentarze
Obsługa narzędzi głównych	Rysowanie za pomocą ołówka, zmiana wielkości oraz koloru ołówka. Wypełnianie tła kolorem. Korygowanie pomyłek przy użyciu gumki. Narzędzie wycinania. Pędzle. Znajomość ikon narzędzi głównych. Kopiowanie, wklejanie oraz obracanie i zmienianie rozmiarów elementów.
Kształty	Przykłady kształtów: Owal, Trójkąt, Prostokąt. Rysowanie linii i krzywych.
Pole tekstowe	Wzbogacenie rysunków o napisy przy pomocy pola tekstowego.
<b>Edytor tekstowy</b>	
Treści	Komentarze
Kombinacje klawiszowe	Podstawowe skróty klawiszowe. Litery diakrytyczne. Wielkie litery. Znaki interpunkcyjne. Klawisze specjalne.
Formatowanie tekstu	Wprowadzenie pojęcia formatowanie tekstu. Pogrubienie, kursywa i podkreślenie.
Listy jednopoziomowe	Wprowadzenie pojęcia listy na przykładzie listy jednopoziomowej. Przykłady wykorzystania listy.
Wyrównywanie tekstu	Wyrównywanie tekstu do lewej, do środka i do prawej. Justowanie.
Akapity	Wprowadzenie pojęcia akapitu. Znajomość ikony służącej do użycia akapitu w tekście.
Reguły pisania tekstu	Podstawowe zasady edytorskie dotyczące składni tekstu, odstępu po znakach itd.
<b>Program do tworzenia prezentacji</b>	
Treści	Komentarze
Slajdy	Tworzenie nowych slajdów, edycja oraz usuwanie slajdów. Zmiana koloru tła slajdu. Dodawanie tytułu oraz tekstu do slajdu. Formatowanie tekstu.
Ilustracje	Źródła ilustracji wykorzystywanych w prezentacji. Dodawanie grafik do slajdów. Zmiana wielkości i położenia grafik. Przestrzeganie zasad korzystania z zasobów internetu.
Zasady tworzenia prezentacji	Tworzenie zwięzłych, spójnych i interesujących prezentacji. Dbanie o poprawność wypowiedzi.
<b>Arkusz kalkulacyjny</b>	
Treści	Komentarze
Podstawowe pojęcia	Wprowadzenie terminów: komórki, kolumny i wiersze.
Formatowanie	Zmiana tła komórki i zbioru komórek. Rodzaje obramowań.
Dane liczbowe i tekstowe	Rozróżnienie między komórkami liczbowymi a komórkami tekstowymi.
Operacje arytmetyczne	Dodawanie, odejmowanie, mnożenie oraz dzielenie wartości liczbowych w komórkach.

## KLASA 5

<b>I. Wprowadzenie</b>	
<b>Prawo i zasady bezpieczeństwa</b>	
Treści	Komentarze
Elementy komputera.	Zewnętrzne elementy komputera: monitor, mysz i klawiatura. Wewnętrzna budowa komputera: płyta główna, procesor, pamięć operacyjna, dysk twardy. Pamięć ROM i RAM.
System operacyjny.	Funkcje systemu operacyjnego. Środowisko tekstowe i graficzne. BIOS. Sekwencja zdarzeń zachodzących w komputerze między uruchomieniem, a pojawieniem się środowiska graficznego.
Piractwo	Zagrożenia związane z piractwem komputerowym.
<b>System binarny</b>	
Treści	Komentarze
Korekcja błędów.	Poprawianie błędów w zapisie liczby w systemie binarnym.

II. Programowanie	
Operacje arytmetyczne	
Treści	Komentarze
Dodawanie i odejmowanie stałych od zmiennej.	Znajdowanie liczby, która jest od danej liczby o 3 większa, o 2 mniejsza, itd., dla jednej, dwóch i trzech liczb. Rozwiązywanie prostych zadań tekstowych wymagających użycia operacji arytmetycznych.
Suma i różnica dwóch zmiennych. Operacje arytmetyczne na dwóch i trzech zmiennych.	Obliczanie wartości prostych wyrażeń arytmetycznych. Poszerzanie rozumienia stanu programu w oparciu o zmienną pomocniczą. <b>Algorytm znajdowania sumy i różnicy dla dwóch i trzech zmiennych.</b>
Algorytm liniowy	Poszerzanie rozumienia algorytmów liniowych.
Instrukcja warunkowa	
Treści	Komentarze
Instrukcja warunkowa z porównaniem między zmienną a stałą	Uzależnienie wykonania jednej instrukcji bądź bloku instrukcji i innej instrukcji bądź innego bloku instrukcji od spełnienia danego warunku. Rozwiązywanie prostych zadań tekstowych wymagających użycia instrukcji warunkowej.
Instrukcja warunkowa z porównaniem między dwiema zmiennymi	Uzależnienie wykonania instrukcji bądź bloku instrukcji od prawdziwości danego porównania między zmienną a stałą. Porównanie dwóch zmiennych w celu sprawdzenia czy są równe. <b>Algorytm znajdowania minimum z dwóch liczb. Algorytm znajdowania maksimum z dwóch liczb.</b>
Łączenie warunków	Łączenie ze sobą dwóch instrukcji warunkowych. Sprawdzanie czy dana liczba należy do przedziału.
Algorytm z warunkami	Poszerzanie rozumienia algorytmów z warunkami.

Pętla pojedyncza	
Treści	Komentarze
Pętla nieskończona	Wprowadzenie pojęcia pętli na przykładzie pętli nieskończonej.
Licznik	Użycie zmiennej, jako licznika wykonanych kroków iteracji.
Sposoby zakończenia iteracji	Różnice w zakończeniu iteracji gdy liczba powtórzeń jest z góry określona lub zależna od warunku.

<b>III. Technologie informacyjne</b>	
<b>Chmura</b>	
Treści	Komentarze
Wprowadzenie do pracy w chmurze	Założenie konta e-mail. Tworzenie konta na wybranej platformie internetowej. Nawigacja po platformie. Udostępnianie plików w ramach platformy.
<b>Filmy</b>	
Treści	Komentarze
Windows Zdjęcia	Uruchamianie programu, interfejs i funkcjonalności.
Przesyłanie plików multimedialnych	Przesyłanie zdjęć czy filmów między urządzeniami, na przykład między smartphonem i komputerem, korzystając z chmury czy USB.
Dodawanie klatek i dźwięku	Dodawanie własnych i ściągniętych z legalnych źródeł zdjęć, jako klatek filmu, edycja przejść między klatkami i ich długości, dodawanie fragmentów filmów jako klatek i ścieżki dźwiękowej.
Praca w grupach	Realizowany w grupach projekt na lekcję języka polskiego, nauka pracy w grupach z pomocą chmury.
<b>Edytor tekstowy</b>	
Treści	Komentarze
Ilustracje	Wstawianie, zmiana położenia oraz wielkości obrazów.
Listy	Tworzenie list punktowych oraz wielopoziomowych numerowanych. Zmiana punktatorów i sposobów numeracji.
Tabele	Wstawianie i formatowanie tabeli.
Napisy i kształty	Wstawianie napisów i kształtów.
<b>Arkusz kalkulacyjny</b>	
Treści	Komentarze
Tabele	Wstawianie i formatowanie tabeli. Tworzenie tabel na podstawie podanych informacji oraz danych znalezionych w internecie.
Formatowanie komórek	Zmiana rodzaju komórek. Zmiana kroju czcionki, koloru oraz wielkości tekstu w komórce. Scalanie i dzielenie komórek.
Formatowanie kolumn i wierszy	Dodawanie i usuwanie kolumn i wierszy. Zmiana szerokości kolumn, wysokości wierszy.
Operacje matematyczne	Wykonywanie operacji matematycznych na całych kolumnach i wierszach.

<b>Program do tworzenia prezentacji</b>	
Treści	Komentarze
Efekty przejścia między slajdami	Korzystanie z efektów przejścia między slajdami.
Szablon	Tworzenie własnego szablonu prezentacji.

<b>Scratch</b>	
Treści	Komentarze
Wprowadzenie do Scratcha	Wejście na platformę Scratch. Nawigowanie po platformie. Podstawy języka. Tworzenie nowego projektu.
Sterowanie postacią	Tworzenie programu sterującego postacią: <ul style="list-style-type: none"> <li>• Góra, dół, prawa, lewa</li> <li>• Obrót w lewo, prawo</li> </ul>

## KLASA 6

<b>I. Wprowadzenie</b>	
<b>Prawo i zasady bezpieczeństwa</b>	
Treści	Komentarze
Złośliwe oprogramowanie	Typy złośliwego oprogramowania. Wirusy, robaki, konie trojańskie, spyware i adware.
Oprogramowanie antywirusowe	Pojęcie oprogramowania antywirusowego. Zasady działania oprogramowania antywirusowego.
<b>System binarny</b>	
Treści	Komentarze
Rewers liczby	Suma liczby i jej rewesu.
Dzielenie całkowitoliczbowe przez 2	Odcięcie ostatniej cyfry liczby w zapisie binarnym.
Dodawanie w systemie binarnym	Dodawanie dwóch liczb w systemie binarnym.
Grupowanie i porządkowanie liczb	Wprowadzenie pojęć: porządek losowy, porządek rosnący. Sortowanie rosnąco liczb. Grupowanie liczb wg cyfry danej pozycji.
Przeszukiwanie binarne	Wprowadzenie pojęcia przeszukiwania binarnego. Przeszukiwanie binarne zbioru uporządkowanego rosnąco.

II Programowanie	
Operacje arytmetyczne	
Treści	Komentarze
Dodawanie i odejmowanie stałych od zmiennej	Znajdowanie liczby, która jest od danej liczby o 3 większa, o 2 mniejsza, itd., dla jednej, dwóch i trzech liczb. Rozwiązywanie prostych zadań tekstowych wymagających użycia operacji arytmetycznych.
Suma i różnica dwóch zmiennych. Operacje arytmetyczne na dwóch i trzech zmiennych.	Obliczanie wartości prostych wyrażeń arytmetycznych. Poszerzanie rozumienia stanu programu w oparciu o zmienną pomocniczą. <b>Algorytm znajdowania sumy i różnicy dla dwóch i trzech zmiennych.</b>
Wielokrotności z zapisu binarnego	Obliczanie wielokrotności danej liczby przy użyciu sumy iloczynów tej liczby oraz potęg liczby 2. <b>Algorytm znajdowania wielokrotności z zapisu binarnego.</b>
Algorytm liniowy	Poszerzanie rozumienia algorytmów liniowych.

Instrukcja warunkowa oraz warunki zagnieżdżone	
Treści	Komentarze
Instrukcja warunkowa z porównaniem między zmienną a stałą	Uzależnienie wykonania jednej instrukcji bądź bloku instrukcji i innej instrukcji bądź innego bloku instrukcji od spełnienia danego warunku. Rozwiązywanie prostych zadań tekstowych wymagających użycia instrukcji warunkowej.
Instrukcja warunkowa z porównaniem między dwiema zmiennymi	Uzależnienie wykonania instrukcji bądź bloku instrukcji od prawdziwości danego porównania między zmienną a stałą. Porównanie dwóch zmiennych w celu sprawdzenia czy są równe. <b>Algorytm znajdowania minimum z dwóch liczb. Algorytm znajdowania maksimum z dwóch liczb.</b>
Łączenie warunków	Łączenie ze sobą dwóch instrukcji warunkowych. Sprawdzanie czy dana liczba należy do przedziału.
Zagnieżdżanie warunków	<b>Algorytm znajdowania minimum z trzech liczb. Algorytm znajdowania maksimum z trzech liczb. Algorytm sortowania bąbelkowego trzech liczb. Algorytm znajdowania liczby środkowej z trzech liczb.</b>
Algorytm z warunkami	Poszerzanie rozumienia algorytmów z warunkami.
Pętla pojedyncza i pętla w pętli	
Treści	Komentarze
Licznik	Użycie zmiennej jako licznika wykonanych kroków iteracji.
Sposoby zakończenia iteracji	Różnice w zakończeniu iteracji, gdy liczba powtórzeń z góry określona lub zależna od warunku.
Algorytmy mnożenia oraz dzielenia	<b>Algorytm znajdowania iloczynu dwóch liczb. Algorytm obliczania kwadratu liczby. Algorytm znajdowania ilorazu dwóch liczb. Algorytm obliczania średniej z trzech liczb. Algorytm znajdowania reszty z dzielenia. Algorytm sprawdzania parzystości danej liczby.</b>
Pętla w pętli	Wyświetlanie na konsoli nieskomplikowanych wzorów przy użyciu pętli zagnieżdżonej, w tym: kwadratów, trójkątów i prostokątów.

<b>III Technologie informacyjne</b>	
<b>Chmura</b>	
Treści	Komentarze
Obsługa aplikacji w chmurze	Tworzenie i edycja prezentacji, dokumentów tekstowych i arkusza kalkulacyjnego on-line w grupie.
<b>Edytor graficzny</b>	
Treści	Komentarze
Podstawy programu (np. GIMP)	Interfejs, obsługa i funkcjonalność programu
Źródła zdjęć	Zgrywanie zdjęć z urządzenia zewnętrznego (aparat, telefon) na dysk. Pobieranie zdjęć z internetu z poszanowaniem praw autorskich. Dodawanie zdjęć lub ilustracji do programu.
Podstawowe narzędzia	Narzędzia zaznaczania: ogólne, prostokątne, eliptyczne, wycinanie i kopiowanie. Narzędzia rysownicze: ołówek, pędzel, wypełnienie, gumka.
Edycja zdjęć	Przycinanie, zmiana wielkości, zmiana jasności oraz faktury. „Rysowanie” na zdjęciach.
<b>Edytor tekstowy</b>	
Treści	Komentarze
Formatowanie tekstu	Przekreślenie, indeks dolny i górny. Podpis pod ilustracją i tabelą. Nowe znaki specjalne.
Formatowanie dokumentu	Zmiana rozmiaru i orientacji strony. Zmiana marginesów.
Drukowanie	Przygotowanie dokumentu do druku.
<b>Arkusz kalkulacyjny</b>	
Treści	Komentarze
Serie danych	Wprowadzanie do arkusza dużych serii danych, w tym automatyczne wypełnianie kolumn lub wierszy.
Funkcje	Wprowadzenie do funkcji w arkuszu kalkulacyjnym. Funkcja Suma.
Wykresy	Wykresy: liniowy, kolumnowy, kołowy. Tworzenie i zastosowanie wykresów. Dobieranie wykresów do danych.
Pasek formuły	Wprowadzenie do paska formuły. Pisanie i korzystanie z prostych formuł. Kopiowanie formuł.
<b>Program do tworzenia prezentacji</b>	
Treści	Komentarze
Tabele	Dodawanie tabel do slajdów.
Wykresy	Dodawanie wykresów do slajdów.



## ROZKŁAD MATERIAŁU W KLASACH 4-6

Poniżej zostały zamieszczone tabele przedstawiające bloki tematyczne realizowane w danej klasie wraz z liczbą godzin potrzebną do ich realizacji. Dla każdej klasy przygotowano również szczegółowy rozkład materiału.

Rok szkolny liczy około 190 dni lekcyjnych. Przewidując 1 godzinę zajęć informatyki tygodniowo otrzymujemy 38 lekcji informatyki rocznie. Biorąc pod uwagę, że w niektórych tygodniach lekcje nie będą mogły się odbyć ze względu na okoliczności losowe, wycieczki szkolne itp., na zrealizowanie materiału w każdej klasie zostały zarezerwowane **32** godziny.

### RAMOWY ROZKŁAD MATERIAŁU - KLASA 4

Blok tematyczny	Liczba godzin
<b>I. Wprowadzenie</b>	
Prawo i zasady bezpieczeństwa	2
System binarny	1
<b>II. Programowanie</b>	
Wprowadzenie do programowania	3
Operacje arytmetyczne	9
Instrukcja warunkowa	5
<b>III. Technologie informacyjne</b>	
Edytor graficzny	3
Edytor tekstowy	3
Program do tworzenia prezentacji	3
Arkusze kalkulacyjny	2

## RAMOWY ROZKŁAD MATERIAŁU - KLASA 5

Blok tematyczny	Liczba godzin
<b>I. Wprowadzenie</b>	
Prawo i zasady bezpieczeństwa	2
System binarny	1
<b>II. Programowanie</b>	
Operacje arytmetyczne	5
Instrukcja warunkowa	7
Pętla pojedyncza	4
<b>III. Technologie informacyjne</b>	
Chmura	2
Filmy	2
Edytor tekstowy	2
Arkusze kalkulacyjne	3
Program do tworzenia prezentacji	2
Scratch	1

## RAMOWY ROZKŁAD MATERIAŁU - KLASA 6

Blok tematyczny	Liczba godzin
<b>I. Wprowadzenie</b>	
Prawo i zasady bezpieczeństwa	2
System binarny	2
<b>II. Programowanie</b>	
Operacje arytmetyczne oraz suma zapisu binarnego	2
Instrukcja warunkowa oraz warunki wielokrotne	7
Pętla pojedyncza i pętla w pętli	7
<b>III. Technologie informacyjne</b>	
Chmura	1
Edytor graficzny	2
Edytor tekstowy	2
Arkusze kalkulacyjne	4
Program do tworzenia prezentacji	2

## SZCZEGÓŁOWY ROZKŁAD MATERIAŁU – KLASA 4

Dział / Rozdział	Temat i opis	Liczba godzin
Dział I	Wprowadzenie	
Rozdział 1	Bezpieczna praca z komputerem	1
Rozdział 2	Korzystanie z internetu	1
Rozdział 3	Zapis liczby w pamięci komputera, system binarny	1
Dział II	Podstawy programowania	
Rozdział 1	Wejście i wyjście z programu, wprowadzenie	2
Rozdział 2	Wejście i wyjście z programu, stałe i zmienne	1
Rozdział 3	Operacje arytmetyczne, zwiększanie i zmniejszanie wartości pudełka o stałą	1
Rozdział 4	Operacje arytmetyczne, zwiększanie i zmniejszanie wartości pudełka, uogólnienie	2
Rozdział 5	Operacje arytmetyczne, dodawanie i odejmowanie na 2-3 pudełkach	1
Rozdział 6	Operacje arytmetyczne, dodawanie i odejmowanie na 2-3 pudełkach, uogólnienie	2
Rozdział 7	Suma i różnica w pudełku pomocniczym, przypisanie wartości (ustaw), wprowadzenie	1
Rozdział 8	Suma i różnica w pudełku pomocniczym, uogólnienie	2
Rozdział 9	Instrukcja warunkowa, gdy warunek spełniony podejmij działanie, inaczej zakończ program	3
Rozdział 10	Instrukcja warunkowa, gdy warunek spełniony podejmij działanie I, inaczej podejmij działanie II	2
Dział III	Technologie informacyjne	
Rozdział 1	Tworzenie grafiki w programie MS Paint Rysowanie linii i krzywych	1
	Tworzenie grafiki w programie MS Paint Rysowanie kształtów. Powielanie wzorów. Pola tekstowe	1
	Tworzenie grafiki w programie MS Paint Projekt międzyprzedmiotowy	1
Rozdział 2	Edytor tekstowy MS Word / LibreOffice Writer Polskie litery. Znaki specjalne. Skróty klawiszowe.	1
	Edytor tekstowy MS Word / LibreOffice Writer Formatowanie czcionek. Listy numerowane.	1
	Edytor tekstowy MS Word / LibreOffice Writer Formatowanie tekstu. Akapity, wcięcia, interlinie. Projekt międzyprzedmiotowy	1
Rozdział 3	Prezentacje w MS Powerpoint / LibreOffice Impress Dodawanie i usuwanie slajdu. Wstawianie tekstu.	1
	Prezentacje w MS Powerpoint / LibreOffice Impress Wklejanie ilustracji. Zasady tworzenia prezentacji. Tryb prezentacji.	1
	Projekt międzyprzedmiotowy	1

**ROZKŁAD MATERIAŁU W KLASACH 4-6**

Rozdział 4	Arkusz kalkulacyjny MS Excel / LibreOffice Calc Komórki, kolumny, wiersze i ich formatowanie.	1
	Arkusz kalkulacyjny MS Excel / LibreOffice Calc Liczby w komórkach, operacje dodawania, odejmowania, mnożenia i dzielenia dwóch komórek.	1
	Podsumowanie i ocenianie	1
	<b>RAZEM</b>	<b>32</b>

**SZCZEGÓLOWY ROZKŁAD MATERIAŁU – KLASA 5**

Dział / Rozdział	Temat i opis	Liczba godzin
<b>Dział I</b>	<b>Wprowadzenie</b>	
Rozdział 1	Bezpieczna praca z komputerem	1
Rozdział 2	Korzystanie z internetu	1
Rozdział 3	Korekcja błędów. Bity parzystości. Sumy kontrolne.	1
<b>Dział II</b>	<b>Podstawy programowania</b>	
Rozdział 1	Wejście i wyjście. Operacje arytmetyczne na jednej zmiennej	1
Rozdział 2	Suma i różnica w pudełku pomocniczym, przypisanie wartości (ustaw)	1
Rozdział 3	Operacje arytmetyczne na 3 wczytanych liczbach - suma i różnica w pudełku pomocniczym	2
Rozdział 4	Operacje arytmetyczne - zadania tekstowe	1
Rozdział 5	Instrukcja warunkowa, gdy warunek spełniony podejmij działanie I, inaczej podejmij działanie II	1
Rozdział 6	Instrukcja warunkowa (przedziały)	2
Rozdział 7	Instrukcja warunkowa (dwa pudełka, max, min)	2
Rozdział 8	Instrukcja warunkowa - zadania tekstowe	2
Rozdział 9	Pętle - wprowadzenie	2
Rozdział 10	Pętle - wielokrotne wypisywanie *a, *a*	2
<b>Dział III</b>	<b>Technologie informacyjne</b>	
Rozdział 1	Chmura Google Stworzenie konta Google (założenie maila) i nawigowanie po platformie Dysk Google Zalety pracy w chmurze Dzielenie się plikami	2
Rozdział 2	Film w programie Windows Zdjęcia Uruchamianie i interfejs programu Dodawanie zdjęć, edycja przejść i długości klatek	1
	Film w programie Windows Zdjęcia Dodawanie ścieżki dźwiękowej Dodawanie fragmentów filmów Projekt w grupach - film „Mity Greckie” Projekt V.1.	1

**ROZKŁAD MATERIAŁU W KLASACH 4-6**

Rozdział 3	Edytor tekstowy Listy punktowe, zmiana sposobu punktacji i numeracji Listy wielopoziomowe numerowane, zmiana sposobu numeracji między poziomami	1
	Edytor tekstowy Wklejanie ilustracji i tabel, zmiana położenia i wielkości Dodawanie napisów i kształtów Projekt na język polski, komiks „Polskie legendy w dzisiejszym świecie”, Projekt V.2.	1
Rozdział 4	Arkusz kalkulacyjny Tworzenie tabeli na podstawie podanych informacji Wyszukiwanie potrzebnych informacji w internecie Formatowanie komórek: zmiana rodzaju komórki, formatowanie tekstu, scalanie i dzielenie komórek	1
	Arkusz kalkulacyjny Formatowanie arkusza: dodawanie i usuwanie kolumn i wierszy, zmiana szerokości kolumn i wysokości wierszy Wykonywanie operacji matematycznych na całych kolumnach i wierszach Projekt na lekcję przyrody „Polska i jej miejsce w Europie” Projekt V.3.	1
	Arkusz kalkulacyjny Projekt na lekcję przyrody „Polska i jej miejsce w Europie” Projekt V.3.	1
Rozdział 5	Prezentacje Edycja przejść między slajdami, tworzenie własnego szablonu	1
	Prezentacja podsumowująca projekt V.3. „Polska i jej miejsce w Europie”	1
Rozdział 6	Scratch Nawigowanie po platformie, podstawy języka Scratch Prosty program realizujący zadanie związane ze sterowaniem postacią na ekranie komputera	1
	Podsumowanie i ocenianie	1
	<b>RAZEM</b>	<b>32</b>

**Dla uczniów szybciej realizujących program dalej przejście do zadań na poziomie klasy 6 kontynuacja na nowym koncie ucznia**

## SZCZEGÓŁOWY ROZKŁAD MATERIAŁU – KLASA 6

Dział / Rozdział	Temat i opis	Liczba godzin
Dział I	Wprowadzenie	
Rozdział 1	Bezpieczna praca z komputerem	1
Rozdział 2	Korzystanie z internetu	1
Rozdział 3	System binarny, wybrane operacje. Rewers liczby Obcięcie ostatniej cyfry liczby w zapisie binarnym Dodawanie w systemie binarnym	1
	Przeszukiwanie binarne Ulica Losowa, Ulica Rosnąca, Osiedle Zhaszowane.	1
Dział II	Podstawy programowania	
Rozdział 1	Operacje arytmetyczne, suma i różnica w pudełku pomocniczym	1
Rozdział 2	Wielokrotności, suma z zapisu binarnego	1
Rozdział 3	Instrukcja warunkowa, gdy warunek spełniony podejmij działanie I, inaczej podejmij działanie II (przedziały i min, max z 2 liczb)	1
Rozdział 4	Pętle proste	1
Rozdział 5	Pętle z wczytaniem licznikiem, mnożenie i kwadrat liczby	2
Rozdział 6	Pętle z wczytaniem licznikiem, dzielenie i reszta z dzielenia, liczenie średniej z 2 i 3 liczb	2
Rozdział 7	Wielokrotne warunki, warunki proste na 3 pudełka	2
Rozdział 8	Wielokrotne warunki, wyszukiwanie największej, najmniejszej z 3 liczb max, min proste	2
Rozdział 9	Wielokrotne warunki, wyszukiwanie największej, najmniejszej z 3 liczb, max, min i modyfikacje	2
Rozdział 10	Pętla za pętlą i pętle zagnieżdżone, rysowanie liter I, L, C oraz prostokątów	2
Dział III	Technologie informacyjne	
Rozdział 1	Chmura Google Nawigowanie po platformie Dokumenty Google Prezentacje – Google Slides Edytor tekstu – Google Docs Arkusze kalkulacyjny – Google Sheets	1
Rozdział 2	Program graficzny GIMP Interfejs i podstawowe narzędzia graficzne programu GIMP Dodawanie zdjęć i ilustracji	1
	Edycja zdjęć w programie GIMP Edycja podstawowych parametrów zdjęć Korzystanie z elementów graficznych Projekt na plastykę, Projekt VI.1. plakat „Bał karnawałowy	1

**ROZKŁAD MATERIAŁU W KLASACH 4-6**

Rozdział 3	Edytor tekstowy Podpis pod ilustracją i tabelą Dodawanie nowych znaków specjalnych Formatowanie tekstu: przekreślenie, indeks dolny i górny	1
	Edytor tekstowy Formatowanie dokumentu: zmiana rozmiaru i orientacji strony, zmiana marginesów Przygotowanie dokumentu do druku Projekt na lekcję języka polskiego, artykuł „Ciekawe wydarzenie” Projekt VI.2	1
Rozdział 4	Arkusz kalkulacyjny Wprowadzanie serii danych Wprowadzenie funkcji: Suma	1
	Arkusz kalkulacyjny Wykresy: liniowy, kolumnowy, kołowy; tworzenie i zastosowanie Dobieranie wykresów do danych	1
	Arkusz kalkulacyjny Czym jest pasek formuły Pisanie i korzystanie z prostych formuł	1
	Arkusz kalkulacyjny Projekt na lekcję przyrody – „Układ Słoneczny” Projekt VI.3.	1
Rozdział 5	Prezentacje Wstawianie wykresów i tabel do prezentacji	1
	Prezentacje Prezentacja podsumowująca projekt VI.3. „Układ Słoneczny”	1
	Podsumowanie i ocenianie	1
	<b>RAZEM</b>	<b>32</b>

**Dla uczniów szybciej realizujących program, dodatkowo w zakresie podstaw programowania:**

Rozdział 11*	Pętla z wczytaniem licznikiem, mnożenie i kwadrat liczby, rozszerzenie	1
Rozdział 12*	Rozdział 12* Pętla z wczytaniem licznikiem, dzielenie i reszta z dzielenia, liczenie średniej z 2 i 3 liczb, rozszerzenie	1
Rozdział 13*	Rozdział 13* Pętla za pętlą, rozszerzenie	1
Rozdział 14*	Rozdział 14* Pętla zagnieżdżona, prostokąty, rozszerzenie	
Rozdział 15*	Rozdział 15* Pętla zagnieżdżona, trójkąty	
Rozdział 16*	Rozdział 16* Wielokrotne warunki, wyszukaj największej, najmniejszej z 3 liczb, max, min i modyfikacje	
Rozdział 17*	Pętla zagnieżdżona z warunkami	

**dalej przejście do zadań na poziomie klasy 7**

\*Temat nieobowiązkowy (wykraczający poza podstawę programową).

## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA

Zadaniem nauczyciela jest przekazanie takiego zakresu programu, na jaki pozwalają indywidualne możliwości danego ucznia. Podczas realizacji lekcji informatyki należy przekazywać wiedzę w zrozumiały dla uczniów sposób, w szczególności podając przykłady praktycznego zastosowania omawianych tematów:

- pokazując konkretne zastosowywania poznanych umiejętności do praktycznych zagadnień, tłumaczenie na przykładach z życia codziennego, (np. projekty międzyprzedmiotowe w zakresie technologii informacyjnych),
- pokazując związek między opanowaniem podstaw programowania, a rozumieniem działania aplikacji komputerowych w ich podstawach, jako narzędzia kontroli i skutecznego ich wykorzystania do przyszłej pracy,
- prezentując alfabetyzację informatyczną jako współczesny odpowiednik podstawowych umiejętności czytania ze zrozumieniem i liczenia.

Zrozumienie przez uczniów danego tematu osiągame jest poprzez ćwiczenia praktyczne. W zakresie podstaw programowania uczniowie rozwiązują problemy i zadania informatyczne w środowisku programistycznym, które dostępne jest na koncie ucznia na platformie instakod.pl. Ćwiczenia te nie tylko budują lepsze rozumienie zagadnień programistycznych oraz sprawność w posługiwaniu się językiem, ale również wspomagają umiejętność czytania ze zrozumieniem, koncentrację i wyobraźnię. Zadaniem nauczyciela jest omówienie na tablicy zadań przykładowych oraz zadań, które sprawiają uczniom szczególne trudności. Rozwiązania tych zadań powinny być budowane wspólnie z uczniami, w szczególności analiza i dyskusja nad błędami jest dobrym środkiem dydaktycznym.

Rozwiązywanie przykładów wspólnie z uczniami, poprzez zadawanie prostych pytań naprowadzających w kierunku znalezienia prawidłowego wyniku, pozwala im lepiej zrozumieć dane zagadnienie. W szczególności bardzo korzystne jest angażowanie uczniów najslabszych poprzez poprowadzenie ich do rozwiązania ścieżką dedukcji (dostosowując pytania do ich poziomu).

Nauka podstaw programowania powinna odbywać się dwutorowo: na kartce papieru, bez możliwości interakcji z komputerem oraz przy wsparciu komputera i środowiska programistycznego.

Nauczyciel w pierwszej kolejności zapisuje wspólnie wymyślone z uczniami rozwiązania na tablicy, prezentując algorytmy i rozwiązania za pomocą różnorodnych diagramów i schematów. Pozwala to na wdrażanie uczniów do intencji działań, w odróżnieniu od pracy metodą prób i błędów (interakcji ze środowiskiem programistycznym). Równoległe głębsze rozumienie wprowadzanych zagadnień ćwiczone może być przy pomocy papierowych kart pracy. W skład kart pracy wchodzi m.in. ćwiczenia polegające na znajdowaniu błędów w kodzie programu, analizie stanów wartości początkowych i końcowych dla danego programu oraz rysowaniu diagramów rozwiązań.

Uczniowie rozwiązują zestawy zadań informatycznych dostępnych na platformie InstaKod w wizualnym środowisku informatycznym. Przechodzą przez ścieżkę pełnego rozwiązania: od analizy i zrozumienia treści zadania, poprzez przemyślenie prawidłowego wyjścia dla konkretnych danych wejściowych, zapisanie rozwiązania w formie diagramu, sprawdzenie jego skuteczności, implementację, automatyczne testowanie, analizę wyników testów oraz korektę błędów.

Praca z podręcznikiem jest również bardzo korzystna. Kształtuje umiejętność czytania ze zrozumieniem oraz wyszukiwania najistotniejszych informacji. Umiejętność przeanalizowania w podręczniku prezentowanych przez nauczyciela na lekcji rozwiązań przykładów pozwala uczniom na pracę we własnym tempie, uczniom słabszym na powtórny analizę przykładów, a uczniom zdolnym na samodzielną pracę nad kolejnymi tematami.



## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA

Praca w grupach pozwala uczniom budować umiejętności społeczne. W szczególności dotyczy to grupowych projektów międzyprzedmiotowych.

### KLASA 4

I. Wprowadzenie	
Prawo i zasady bezpieczeństwa	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• omawia zasady pracy w pracowni komputerowej</li> <li>• omawia zasady tworzenia i korzystania z haseł</li> <li>• omawia zagadnienie korzystania z internetu</li> <li>• omawia zasady bezpieczeństwa w internecie</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z zasadami pracy w pracowni komputerowej</li> <li>• zapoznają się z zasadami tworzenia i korzystania z haseł</li> <li>• zapoznają się z zagadnieniem korzystania z internetu</li> <li>• zapoznają się z zasadami bezpieczeństwa w internecie</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>
System binarny	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia dlaczego liczby są istotne w informatyce</li> <li>• omawia na przykładach liczb (np. 3, 7, 13) zasady ich zapisu w systemie binarnym</li> <li>• wyjaśnia czym dla komputera jest stan 1 (z prądem) oraz 0 (bez prądu)</li> </ul>	<ul style="list-style-type: none"> <li>• dokonują konwersji liczby zapisanej w systemie binarnym na system dziesiętny</li> <li>• dokonują konwersji liczby zapisanej w systemie dziesiętnym na system binarny</li> </ul>
II. Programowanie	
Wprowadzenie do programowania	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia czym są programy komputerowe oraz kim są programiści</li> <li>• wyjaśnia rolę języków programowania</li> <li>• wyjaśnia na przykładzie czym jest wejście oraz wyjście z programu</li> <li>• omawia instrukcje wejścia i wyjścia języka wizualnego Assembly</li> <li>• wyjaśnia czym jest zmienna na przykładzie pudełka przechowującego liczby</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji wejścia i wyjścia</li> <li>• wyjaśnia czym jest stan programu</li> <li>• zwraca uwagę uczniów na różnicę między wyjściem stałym i zmiennym</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji wejścia i wyjścia oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• poznają fundamentalne dla programowania pojęcia, wejście, wyjście, zmienna oraz stan programu</li> <li>• weryfikują i uzupełniają swoją dotychczasową wiedzę na temat programowania: czym jest i na czym polega</li> <li>• zapoznają się z instrukcjami wejścia i wyjścia w języku wizualnym Assembly</li> <li>• objaśniają działanie poznanych instrukcji</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów z lekcji</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

Operacje arytmetyczne	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• omawia instrukcje dodawania i odejmowania w języku wizualnym Assembly</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji dodawania i odejmowania</li> <li>• wyjaśnia na przykładzie czym jest algorytm liniowy</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji dodawania i odejmowania oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia potrzebę korzystania ze zmiennej pomocniczej przy bardziej złożonych obliczeniach</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji dodawania i odejmowania oraz zmiennej pomocniczej</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji dodawania, odejmowania i zmiennej pomocniczej oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z instrukcjami dodawania i odejmowania w języku wizualnym Assembly</li> <li>• objaśniają działanie poznanych instrukcji</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują na papierowych kartach pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• najpierw w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do przykładów podanych przez nauczyciela, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z koncepcją zmiennej pomocniczej</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do kolejnych przykładów podanych przez nauczyciela, a następnie implementują swoje rozwiązania na komputerach</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów z lekcji</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA

Instrukcje warunkowe	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia czym są wyrażenia logiczne oraz zapisuje sześć operatorów porównania</li> <li>• omawia instrukcję warunkową w języku wizualnym Assembly</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między zmienną i stałą</li> <li>• wyjaśnia na przykładzie czym jest Algorytm z warunkami</li> <li>• wyjaśnia różnicę między algorytmem liniowym i drzewkowym</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla warunku między zmienną a stałą oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z sześcioma operatorami porównania</li> <li>• samodzielnie rozwiązują na papierze karty pracy mające utrwalić poznane operatory porównań oraz ocenę prawdziwości wyrażenia logicznego</li> <li>• zapoznają się z instrukcją warunkową w języku wizualnym Assembly</li> <li>• zapoznają się z porównaniem zmiennej i stałej za pomocą instrukcji warunkowej</li> <li>• objaśniają działanie poznanej instrukcji</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w kartach pracy zadania mające utrwalić poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z porównaniem dwóch zmiennych za pomocą instrukcji warunkowej</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy kolejne zadania mające utrwalić poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

<b>III Technologie informacyjne</b>	
<b>Edytor graficzny</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia użyteczność edytora graficznego</li> <li>• pokazuje jak korzystać z podstawowych narzędzi edytora graficznego</li> <li>• wyjaśnia sposób korzystania z gotowych kształtów</li> <li>• pokazuje jak korzystać z pola tekstowego</li> <li>• pokazuje jak zapisać gotowy plik na USB</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję plastyki</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą rysunki korzystając z narzędzi edytora graficznego</li> <li>• tworzą rysunki składające się z figur geometrycznych</li> <li>• korzystają z pola tekstowego oraz formatują napisany tekst</li> <li>• zapisują gotowy plik na USB</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję plastyki</li> </ul>

<b>Edytor tekstowy</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia użyteczność edytora tekstowego</li> <li>• wyjaśnia, na przykładzie, jak pisać i formatować tekst</li> <li>• pokazuje jak tworzyć jednopoziomowe listy numerowane</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję języka polskiego</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z zasadami pisania i formatowania tekstu</li> <li>• tworzą i formatują prosty tekst</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję języka polskiego</li> </ul>

<b>Program do tworzenia prezentacji</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia użyteczność programu do tworzenia prezentacji</li> <li>• wyjaśnia na przykładzie jak dodawać i uzupełniać slajdy</li> <li>• omawia zasady przedstawiania prezentacji</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję historii</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z zasadami tworzenia prezentacji</li> <li>• tworzą prezentację z wieloma slajdami</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję historii</li> </ul>

<b>Arkusz kalkulacyjny</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia użyteczność arkusza kalkulacyjnego</li> <li>• wyjaśnia pojęcia: komórka, wiersz i kolumna</li> <li>• pokazuje jak wprowadzać dane do arkusza oraz wykonywać operacje arytmetyczne</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję przyrody</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z podstawowymi pojęciami związanymi z arkuszami kalkulacyjnymi</li> <li>• wprowadzają dane do arkusza i wykonują proste operacje arytmetyczne</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję przyrody</li> </ul>

## KLASA 5

<b>I. Wprowadzenie</b>	
<b>Prawo i zasady bezpieczeństwa</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• omawia zewnętrzne elementy komputera</li> <li>• omawia budowę komputera</li> <li>• wyjaśnia przeznaczenie systemu operacyjnego</li> <li>• wyjaśnia różnicę między środowiskiem graficznym i tekstowym</li> <li>• zlecając uczniom włączenie komputerów omawia proces ich uruchamiania</li> <li>• wyjaśnia czym jest piractwo komputerowe i dlaczego jest zabronione</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• wymieniają zewnętrzne elementy komputera</li> <li>• zapoznają się z budową komputera</li> <li>• zapoznają się z przeznaczeniem systemu operacyjnego</li> <li>• zapoznają się z różnicami między środowiskiem graficznym i tekstowym</li> <li>• śledząc, po włączeniu komputera obraz na ekranie monitora i słuchając wyjaśnień nauczyciela zapoznają się z procesem uruchomienia systemu operacyjnego</li> <li>• zapoznają się z pojęciem piractwa komputerowego</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

<b>System binarny</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina dlaczego liczby są istotne w informatyce</li> <li>• przypomina czym dla komputera jest stan 1 (z prądem) oraz 0 (bez prądu)</li> <li>• powtarza omówione już w poprzednim roku zasady zapisu liczb w systemie binarnym, podaje przykłady zapisów</li> <li>• omawia na przykładzie, jak dokonywać korekcyj błędów zapisu liczby w systemie binarnym</li> </ul>	<ul style="list-style-type: none"> <li>• dokonują konwersji liczby zapisanej w systemie binarnym na system dziesiętny</li> <li>• dokonują konwersji liczby zapisanej w systemie dziesiętnym na system binarny</li> <li>• dokonują korekcyj błędów zapisu liczby w systemie binarnym</li> </ul>

II. Programowanie	
Operacje arytmetyczne	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina instrukcje dodawania i odejmowania w języku wizualnym Assembly</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji dodawania i odejmowania</li> <li>• przypomina czym jest algorytm liniowy</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji dodawania i odejmowania oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• przypomina potrzebę korzystania ze zmiennej pomocniczej przy bardziej złożonych obliczeniach</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji dodawania i odejmowania oraz zmiennej pomocniczej</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji dodawania, odejmowania i zmiennej pomocniczej oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• przypominają sobie instrukcje dodawania i odejmowania w języku wizualnym Assembly</li> <li>• objaśniają działanie poznanych instrukcji</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania mające utrwalić poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do wykonanych przykładów, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działania swoich programów</li> <li>• przypominają sobie czym jest zmienna pomocnicza</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów z lekcji</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

Instrukcje warunkowe	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina czym są wyrażenia logiczne oraz zapisuje sześć operatorów porównania</li> <li>• przypomina instrukcję warunkową w wizualnym języku Assembly</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między zmienną i stałą</li> <li>• przypomina czym jest Algorytm z warunkami</li> <li>• przypomina różnicę między algorytmem liniowym i drzewkowym</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla warunku między zmienną i stałą oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• wyjaśnia na przykładzie w jaki sposób łączyć ze sobą warunki</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla dwóch warunków oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• przypominają sobie sześć operatorów porównania</li> <li>• samodzielnie rozwiązują w karcie pracy zadania mające utrwalić poznane operatory porównań oraz ocenę prawdziwości wyrażenia logicznego</li> <li>• przypominają sobie zasadę działania instrukcji warunkowej w języku wizualnym Assembly</li> <li>• przypominają sobie jak dokonuje się porównania zmiennej i stałej za pomocą instrukcji warunkowej</li> <li>• objaśniają działanie instrukcji warunkowej</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• samodzielnie, w zeszytach rysują drzewkowe algorytmy rozwiązania zadań, analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z porównaniem dwóch zmiennych za pomocą instrukcji warunkowej</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania mające utrwalić poznane koncepcje oraz instrukcje</li> <li>• samodzielnie, w zeszytach rysują drzewkowe algorytmy rozwiązania zadań, analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA

<b>Pętla pojedyncza</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• na przykładzie pętli nieskończonej wprowadza pojęcie pętli</li> <li>• omawia konstrukcję pętli w języku wizualnym Assembly</li> <li>• wyjaśnia na przykładzie jak wyglądają algorytmy drzewkowe dla pętli pojedynczej</li> <li>• omawia pojęcie licznika</li> <li>• omawia sposoby zakończenia iteracji</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą pojedynczą</li> <li>• zleca uczniom rozwiązanie zadań z pętlą pojedynczą oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z pojęciami: pętla, pętla nieskończona, licznik.</li> <li>• zapoznają się ze sposobem konstrukcji pętli w języku wizualnym Assembly</li> <li>• objaśniają zasadę działania pętli w języku wizualnym Assembly</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli</li> <li>• samodzielnie, w zeszytach rysują drzewkowe algorytmy rozwiązań zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

### III. Technologie informacyjne

#### **Edytor graficzny**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak korzystać z edytora graficznego w celu edycji zdjęć</li> <li>• zleca wykonanie projektu na lekcję plastyki</li> </ul>	<ul style="list-style-type: none"> <li>• dodają własne zdjęcia i edytują je korzystając z narzędzi graficznych</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na plastykę</li> </ul>

#### **Edytor tekstowy**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak w dodawać ilustracje</li> <li>• pokazuje jak tworzyć listy punktowe oraz wielopoziomowe listy numerowane</li> <li>• pokazuje jak dodawać tabele</li> <li>• zleca wykonanie projektu na lekcję języka polskiego</li> </ul>	<ul style="list-style-type: none"> <li>• dodają ilustracje</li> <li>• tworzą listy punktowe i wielopoziomowe</li> <li>• dodają tabele</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję języka polskiego</li> </ul>



## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA

### Arkusz kalkulacyjny

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak tworzyć tabele na podstawie dostępnych danych</li> <li>• pokazuje jak wizualizować dane z tabel za pomocą wykresów słupkowych</li> <li>• na przykładzie funkcji SUMA i ŚREDNIA objaśnia czym jest funkcja w arkuszu kalkulacyjnym</li> <li>• zleca uczniom wykonywanie ćwiczeń opisanych w podręczniku, monitoruje ich pracę</li> <li>• zleca uczniom wykonanie projektu na lekcję przyrody</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą tabele na podstawie dostępnych danych</li> <li>• tworzą wykresy słupkowe na podstawie danych w arkuszu</li> <li>• korzystają z funkcji SUMA i ŚREDNIA w celu wykonania prostych obliczeń</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję przyrody</li> </ul>

### Program do tworzenia prezentacji

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak dodać efekty przejść między slajdami</li> <li>• wydaje uczniom polecenie samodzielnego wykonywania ćwiczeń opisanych w podręczniku</li> <li>• monitoruje pracę uczniów</li> <li>• zleca wykonanie projektu na lekcję historii</li> </ul>	<ul style="list-style-type: none"> <li>• korzystają z efektów przejść między slajdami oraz animacji</li> <li>• wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję historii</li> </ul>

### Chmura

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia użyteczność chmury</li> <li>• pokazuje jak wejść na platformę oraz założyć konto w wybranej usłudze internetowej</li> <li>• zleca uczniom wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• zakładają konta w wybranej przez nauczyciela usłudze internetowej</li> <li>• udostępniają pliki w ramach platformy</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

### Scratch

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia czym jest Scratch</li> <li>• pokazuje jak wejść na platformę Scratch</li> <li>• pokazuje jak tworzyć własną postać animowaną</li> <li>• zleca uczniom wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą własną postać animowaną</li> <li>• sterują ruchem po ekranie stworzonej postaci</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

## KLASA 6

<b>I. Wprowadzenie</b>	
<b>Prawo i zasady bezpieczeństwa</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• omawia czym są wirusy i inne typy złośliwego oprogramowania</li> <li>• omawia zasady działania oprogramowania antywirusowego</li> <li>• zleca uczniom wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z typami złośliwego oprogramowania</li> <li>• zapoznają się z zasadami działania oprogramowania antywirusowego</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

<b>System binarny</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina dlaczego liczby są istotne w informatyce</li> <li>• przypomina czym dla komputera jest stan 1 (z prądem) oraz 0 (bez prądu)</li> <li>• omawia, na przykładach, zasady zapisu liczb w systemie binarnym</li> <li>• omawia, na przykładach, czym jest rewers liczby w systemie binarnym</li> <li>• pokazuje na przykładach, że suma liczby o zadanej liczbie cyfr i jej rewersu jest zawsze stała</li> <li>• omawia na przykładzie jak dodać do siebie dwie liczby w systemie binarnym</li> <li>• omawia na przykładach ulicy Losowej i Rosnącej oraz osiedla Zhaszowanego czym są porządek losowy, rosnący i grupowanie wg cyfry danej pozycji liczby</li> <li>• omawia czym jest algorytm przeszukiwania binarnego zbioru uporządkowanego rosnąco</li> </ul>	<ul style="list-style-type: none"> <li>• dokonują konwersji liczby zapisanej w systemie binarnym na system dziesiętny</li> <li>• dokonują konwersji liczby zapisanej w systemie dziesiętnym na system binarny</li> <li>• dokonują rewersu liczby zapisanej w systemie binarnym</li> <li>• dzielą całkowitoliczbowo liczbę w systemie binarnym na 2</li> <li>• dodają dwie liczby w systemie binarnym</li> <li>• rozróżniają porządek losowy, rosnący, malejący oraz grupowanie względem wybranej pozycji w liczbie</li> <li>• sortują liczby rosnąco oraz grupują wg cyfry danej pozycji</li> <li>• przeszukują binarnie zbiór uporządkowany rosnąco</li> </ul>

II Programowanie	
Operacje arytmetyczne	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina instrukcje dodawania i odejmowania w języku wizualnym Assembly</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji dodawania i odejmowania</li> <li>• przypomina czym jest algorytm liniowy</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji dodawania i odejmowania oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• przypomina potrzebę korzystania ze zmiennej pomocniczej przy bardziej złożonych obliczeniach</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji dodawania i odejmowania oraz zmiennej pomocniczej</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji dodawania, odejmowania i zmiennej pomocniczej oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• przypominają sobie instrukcje dodawania i odejmowania w języku wizualnym Assembly</li> <li>• objaśniają działanie poznanych instrukcji</li> <li>• analizują sposób rozwiązania przykładowego wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działania swoich programów</li> <li>• przypominają sobie czym jest zmienna pomocnicza</li> <li>• analizują sposób rozwiązania przykładowego wybranego przez nauczyciela</li> <li>• samodzielnie w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

Instrukcje warunkowe oraz warunki zagnieżdżone	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina instrukcję warunkową w języku wizualnym Assembly</li> <li>• przypomina czym jest Algorytm z warunkami</li> <li>• przypomina różnicę między algorytmem liniowym i drzewkowym</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla dwóch warunków oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• pokazuje jak rysować diagram drzewkowy dla warunków zagnieżdżonych</li> <li>• rozwiązuje na tablicy przykładowe zadanie z warunkami zagnieżdżonymi dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z warunkami zagnieżdżonymi oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• przypominają sobie zasadę działania instrukcji warunkowej w języku wizualnym Assembly</li> <li>• objaśniają działanie instrukcji warunkowej</li> <li>• analizują sposób rozwiązania przykładowego wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w ka pracy mające utrwalić poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się ze sposobem rozwiązywania zadań z zagnieżdżonymi warunkami</li> <li>• analizują sposób rozwiązania przykładowego wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

Pętla pojedyncza i pętla w pętli	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina zasadę działania pętli w języku wizualnym Assembly</li> <li>• przypomina jak wyglądają algorytmy drzewkowe dla pętli pojedynczej</li> <li>• przypomina sposoby zakończenia iteracji</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą pojedynczą</li> <li>• zleca uczniom rozwiązanie zadań z pętli pojedynczej oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia algorytmy mnożenia i dzielenia za pomocą pętli</li> <li>• rozwiązuje na tablicy przykładowe zadanie na policzenie iloczynu lub ilorazu</li> <li>• zleca uczniom rozwiązanie zadań wymagających wykorzystania poznanych algorytmów oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• wyjaśnia zasadę konstrukcji pętli w pętli w języku wizualnym Assembly</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą w pętli</li> <li>• zleca uczniom rozwiązanie zadań z pętlą w pętli oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• przypominają sobie zasadę działania pętli w pętli w języku wizualnym Assembly</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z algorytmami mnożenia i dzielenia za pomocą pętli</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się ze sposobem konstrukcji pętli w pętli w języku wizualnym Assembly</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA

<b>II. Technologie informacyjne</b>	
<b>Edytor graficzny</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak korzystać z edytora graficznego w celu edycji zdjęć</li> <li>• zleca wykonanie projektu na lekcję plastyki</li> </ul>	<ul style="list-style-type: none"> <li>• dodają własne zdjęcia i edytują je korzystając z narzędzi graficznych</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję plastyki</li> </ul>
<b>Edytor tekstowy</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak w formatować tekst</li> <li>• pokazuje jak formatować dokument</li> <li>• pokazuje jak przygotować dokument do druku</li> <li>• zleca wykonanie projektu na lekcję języka polskiego</li> </ul>	<ul style="list-style-type: none"> <li>• formatują tekst</li> <li>• formatują dokument</li> <li>• przygotowują dokument do druku</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję języka polskiego</li> </ul>
<b>Chmura</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak korzystać w chmurze z edytora tekstu, programu do tworzenia prezentacji i arkusza kalkulacyjnego</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą w chmurze arkusze kalkulacyjne, dokumenty tekstowe oraz prezentacje</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>
<b>Arkusz kalkulacyjny</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• objaśnia czym jest pasek formuły</li> <li>• na przykładzie funkcji SUMA i ŚREDNIA objaśnia czym jest funkcja w arkuszu kalkulacyjnym</li> <li>• omawia zalety różnych typów wykresów</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję przyrody</li> </ul>	<ul style="list-style-type: none"> <li>• korzystają z funkcji SUMA i ŚREDNIA w celu wykonania prostych obliczeń</li> <li>• dodają różne typy wykresów w zależności od danych</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję przyrody</li> </ul>
<b>Program do tworzenia prezentacji</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak dodać tabele i wykresy do slajdu</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję historii</li> </ul>	<ul style="list-style-type: none"> <li>• dodają tabele i wykresy do slajdów</li> <li>• wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję historii</li> </ul>

## OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Ocenianie jest jednym z ważniejszych elementów pracy nauczyciela. Ocena pełni zarówno rolę informacyjną, jak i motywacyjną. Ocena informuje ucznia i rodzica o bieżących postępach w nauce, pokazuje z czym uczeń ma trudności, a co opanował w pełni. Ocena powinna uwzględniać nie tylko wyniki nauki, ale również wkład pracy włożonej przez ucznia i relatywny wzrost jego umiejętności. Dzięki temu nawet najslabsi uczniowie będą zmotywowani do dalszego wysiłku, wiedząc, że mają szansę na otrzymanie dobrego stopnia. Uczniom zdolnym należy proponować ambitniejsze zadania i szybsze tempo pracy, by mogli wykazać się swoimi umiejętnościami i rozwijać swój potencjał.

System oceniania powinien być jasny zarówno dla uczniów jak i ich rodziców. Powinien być przedstawiony na początku roku szkolnego i należy wystrzegać się jego zmian w trakcie trwania danego etapu nauczania. W systemie oceniania powinny być uwzględnione następujące kryteria:

- wiedza oraz umiejętności techniczne,
- zaangażowanie oraz aktywność na lekcjach,
- systematyczność,
- wysiłek włożony w naukę.

Uwzględniając te kryteria program *InstaKod* ustala poniższe zasady oceniania uczniów na lekcjach informatyki.

Na ocenę dopuszczającą zasługuje uczeń, który:

- wykonał minimum wysiłku w zrozumieniu najprostszych zagadnień omawianych w danym temacie.
- w zakresie podstaw programowania:
  - rozwiązał na swoim koncie zadania przykładowe, mając dostęp do wzorcowych rozwiązań.
  - potrzebuje pomocy nauczyciela w rozwiązywaniu prostych zadań, nie mając dostępu do wzorcowych rozwiązań.

Na ocenę dostateczną zasługuje uczeń, który:

- wykonał wysiłek w zrozumieniu najprostszych zagadnień omawianych w danym temacie.
- w zakresie podstaw programowania
  - rozwiązał na swoim koncie zadania przykładowe, mając dostęp do wzorcowych rozwiązań.
  - samodzielnie rozwiązał na swoim koncie najprostsze zadania (zadania z poziomu 1).

Na ocenę dobrą zasługuje uczeń, który:

- rozumie i potrafi własnymi słowami wyjaśnić omawiane zagadnienia.
- systematycznie odrabia pracę domową.
- w zakresie podstaw programowania samodzielnie rozwiązał na swoim koncie zadania najprostsze oraz średniej trudności (zadania z poziomu 1 i 2).

Na ocenę bardzo dobrą zasługuje uczeń, który:

- rozumie i potrafi własnymi słowami wyjaśnić omawiane zagadnienia.
- systematycznie odrabia pracę domową.
- jest zaangażowany w lekcję i aktywnie w niej uczestniczy.
- w zakresie podstaw programowania samodzielnie rozwiązał na swoim koncie zadania ze wszystkich trzech poziomów trudności.

Na ocenę celującą zasługuje uczeń, który:

- rozumie i potrafi własnymi słowami wyjaśnić omawiane zagadnienia.
- systematycznie odrabia pracę domową.
- w zakresie podstaw programowania samodzielnie rozwiązał na swoim koncie zadania ze wszystkich trzech poziomów trudności oraz zadania z gwiazdką
- wykazał się wysokim poziomem opanowania wiedzy oraz aktywnością na lekcjach
- w szczególności na ocenę celującą zasługują uczniowie, którzy przeszli do wyższych poziomów klas w *InstaKod* po zrealizowaniu na ocenę bardzo dobrą poziomu swojej klasy oraz uczniowie, którzy otrzymali wyróżnienia w konkursach informatycznych.

Kierując się powyższymi zasadami oceniania, skonstruowano tabele wymagań na poszczególne oceny. Mają one charakter orientacyjny.

KLASA 4

<b>I. Wprowadzenie</b>				
<b>Prawo i zasady bezpieczeństwa</b>				
Na ocenę <b>2</b> uczniów:	Na ocenę <b>3</b> uczniów:	Na ocenę <b>4</b> uczniów:	Na ocenę <b>5</b> uczniów:	Na ocenę <b>6</b> uczniów:
z pomocą nauczyciela tworzy bezpieczne hasło	samodzielnie tworzy bezpieczne hasło	omawia zasady pracy w pracowni komputerowej; omawia zasady bezpieczeństwa w internecie; samodzielnie tworzy bezpieczne hasło	omawia zasady pracy w pracowni komputerowej; omawia zagadnienie korzystania z internetu; omawia zasady bezpieczeństwa w internecie; samodzielnie tworzy bezpieczne hasło	omawia zasady pracy w pracowni komputerowej; omawia zagadnienie korzystania z internetu; omawia zasady bezpieczeństwa w internecie; rozszerza wspomniane zagadnienia o treści nie wspomniane w podręczniku; samodzielnie tworzy bezpieczne hasło
<b>System binarny</b>				
Na ocenę <b>2</b> uczniów:	Na ocenę <b>3</b> uczniów:	Na ocenę <b>4</b> uczniów:	Na ocenę <b>5</b> uczniów:	Na ocenę <b>6</b> uczniów:
potrafi rozpoznać liczbę zapisaną w systemach binarnym oraz dziesiętnym; z pomocą nauczyciela dokonuje konwersji liczb z systemu dziesiętnego na binarny w zakresie do 16	potrafi rozpoznać liczbę zapisaną w systemach binarnym oraz dziesiętnym; samodzielnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 16; z pomocą nauczyciela dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 32	wyjaśnia, czym dla komputera są stany 0 i 1; samodzielnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 32; z pomocą nauczyciela dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny większych od 32	rozumie istotę liczb w informatyce; rozumie czym jest system pozycyjny; płynnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 64	rozumie istotę liczb w informatyce; rozumie czym jest system pozycyjny; płynnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny większych od 64



OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

II. Programowanie				
Wprowadzenie do programowania				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
<p>z pomocą nauczyciela loguje się na platformę;</p> <p>rozdziela instrukcje wejścia i wyjścia z programu;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji</p>	<p>samodzielnie loguje się na platformę;</p> <p>rozdziela instrukcje wejścia i wyjścia z programu;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji;</p> <p>z pomocą nauczyciela rozwiązuje trudniejsze zadania</p>	<p>samodzielnie loguje się na platformę;</p> <p>zna i potrafi objaśnić działanie instrukcji wejścia i wyjścia;</p> <p>rozumie różnice między wyjściem stałym i zmiennym;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji;</p> <p>z pomocą nauczyciela rozwiązuje trudniejsze zadania</p>	<p>samodzielnie loguje się na platformę;</p> <p>zna i potrafi objaśnić działanie instrukcji wejścia i wyjścia;</p> <p>rozumie różnice między wyjściem stałym i zmiennym;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p>rozumie pojęcie stanu programu;</p> <p>objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>samodzielnie loguje się na platformę;</p> <p>zna i potrafi objaśnić działanie instrukcji wejścia i wyjścia;</p> <p>rozumie różnice między wyjściem stałym i zmiennym;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>rozumie pojęcie stanu programu;</p> <p>objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Operacje arytmetyczne				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>rozdziela pojęcia sumy i różnicy i ich instrukcje;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy</p>	<p>rozdziela pojęcia sumy i różnicy i ich instrukcje;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności</p>	<p>zna i potrafi objaśnić pojęcia sumy i różnicy i działanie ich instrukcji;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudniejsze, w tym zadania wymagające użycia pudełka pomocniczego</p>	<p>zna i potrafi objaśnić pojęcia sumy i różnicy i działanie ich instrukcji;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna i potrafi objaśnić pojęcia sumy i różnicy i działanie ich instrukcji;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p> objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Instrukcje warunkowe				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>rozdzieli sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p>	<p>rozdzieli sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające porównania między dwoma zmiennymi;</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające porównania między dwoma zmiennymi;</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p> płynnie rozwiązuje zadania proste oraz o średniej trudności, w tym zadania wymagające porównania między dwoma zmiennymi;</p> <p>objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

III. Technologie informacyjne				
Edytor graficzny				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
<p>z pomocą nauczyciela otwiera edytor graficzny;</p> <p>z pomocą nauczyciela tworzy prosty rysunek z użyciem narzędzi malarskich (Ołówek, Pędzel, Linia, Gumka)</p> <p>z pomocą nauczyciela tworzy prosty rysunek z użyciem figur geometrycznych (prostokąty, kwadraty, okręgi i trójkąty)</p>	<p>samodzielnie otwiera edytor graficzny;</p> <p>wyjaśnia, do czego służy edytor graficzny;</p> <p>samodzielnie tworzy prosty rysunek z użyciem narzędzi malarskich (Ołówek, Pędzel, Linia, Gumka)</p> <p>samodzielnie tworzy prosty rysunek z użyciem figur geometrycznych (prostokąty, kwadraty, okręgi i trójkąty)</p> <p>z pomocą nauczyciela korzysta z pola tekstowego;</p> <p>z pomocą nauczyciela wykonuje operacje na fragmencie rysunku: zaznacza, wycina, kopiuje i wkleja go w inne miejsce na tym samym rysunku</p>	<p>samodzielnie otwiera edytor graficzny;</p> <p>potrafi podać przykłady zastosowania edytora graficznego;</p> <p>samodzielnie tworzy prosty rysunek z użyciem narzędzi malarskich (Ołówek, Pędzel, Linia, Gumka)</p> <p>samodzielnie tworzy prosty rysunek z użyciem figur geometrycznych (prostokąty, kwadraty, okręgi i trójkąty)</p> <p>samodzielnie korzysta z pola tekstowego;</p> <p>samodzielnie wykonuje operacje na fragmencie rysunku: zaznacza, wycina, kopiuje i wkleja go w inne miejsce na tym samym rysunku;</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem</p>	<p>samodzielnie otwiera edytor graficzny;</p> <p>potrafi podać przykłady zastosowania edytora graficznego;</p> <p>samodzielnie tworzy prosty rysunek z użyciem narzędzi malarskich (Ołówek, Pędzel, Linia, Gumka)</p> <p>samodzielnie tworzy prosty rysunek z użyciem figur geometrycznych (prostokąty, kwadraty, okręgi i trójkąty)</p> <p>samodzielnie korzysta z pola tekstowego;</p> <p>samodzielnie wykonuje operacje na fragmencie rysunku: zaznacza, wycina, kopiuje i wkleja go w inne miejsce na tym samym rysunku;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera edytor graficzny;</p> <p>potrafi podać przykłady zastosowania edytora graficznego;</p> <p>samodzielnie tworzy prosty rysunek z użyciem narzędzi malarskich (Ołówek, Pędzel, Linia, Gumka)</p> <p>samodzielnie tworzy prosty rysunek z użyciem figur geometrycznych (prostokąty, kwadraty, okręgi i trójkąty)</p> <p>samodzielnie korzysta z pola tekstowego;</p> <p>samodzielnie wykonuje operacje na fragmencie rysunku: zaznacza, wycina, kopiuje i wkleja go w inne miejsce na tym samym rysunku;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p> <p>przygotowuje rysunki na konkursy informatyczne</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Edytor tekstowy</b>				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>z pomocą nauczyciela otwiera edytor tekstowy;</p> <p>samodzielnie zmienia rozmiar i kolor czcionki;</p> <p>samodzielnie usuwa znaki za pomocą klawisza Backspace</p> <p>z pomocą nauczyciela pisze krótki tekst, wielkie i małe litery oraz polskie znaki diakrytyczne;</p>	<p>samodzielnie otwiera edytor tekstowy;</p> <p>samodzielnie zmienia czcionkę oraz jej rozmiar i kolor;</p> <p>samodzielnie usuwa znaki za pomocą klawisza Backspace</p> <p>samodzielnie pisze krótki tekst, wielkie i małe litery oraz polskie znaki diakrytyczne;</p> <p>z pomocą nauczyciela wstawia akapity oraz wyrównuje je do lewej, do prawej, do środka;</p>	<p>samodzielnie otwiera edytor tekstowy;</p> <p>potrafi podać przykłady zastosowania edytora tekstowego;</p> <p>samodzielnie zmienia czcionkę oraz jej rozmiar i kolor;</p> <p>samodzielnie usuwa znaki za pomocą klawisza Backspace</p> <p>samodzielnie pisze krótki tekst, wielkie i małe litery oraz polskie znaki diakrytyczne;</p> <p>samodzielnie wstawia akapity oraz wyrównuje je do lewej, do prawej, do środka;</p> <p>samodzielnie prawidłowo stosuje spacje przy znakach interpunkcyjnych;</p> <p>z pomocą nauczyciela stosuje listy jednopoziomowe</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem</p>	<p>samodzielnie otwiera edytor tekstowy;</p> <p>potrafi podać przykłady zastosowania edytora tekstowego;</p> <p>samodzielnie zmienia czcionkę oraz jej rozmiar i kolor;</p> <p>samodzielnie usuwa znaki za pomocą klawisza Backspace</p> <p>samodzielnie pisze krótki tekst, wielkie i małe litery oraz polskie znaki diakrytyczne;</p> <p>samodzielnie wstawia akapity oraz wyrównuje je do lewej, do prawej, do środka;</p> <p>samodzielnie prawidłowo stosuje spacje przy znakach interpunkcyjnych;</p> <p>samodzielnie stosuje listy jednopoziomowe</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera edytor tekstowy;</p> <p>potrafi podać przykłady zastosowania edytora tekstowego;</p> <p>samodzielnie zmienia czcionkę oraz jej rozmiar i kolor;</p> <p>samodzielnie usuwa znaki za pomocą klawisza Backspace</p> <p>samodzielnie pisze krótki tekst, wielkie i małe litery oraz polskie znaki diakrytyczne;</p> <p>samodzielnie wstawia akapity oraz wyrównuje je do lewej, do prawej, do środka;</p> <p>samodzielnie prawidłowo stosuje spacje przy znakach interpunkcyjnych;</p> <p>samodzielnie stosuje listy jednopoziomowe</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p> <p>pisze tekst, stosując poprawnie poznane zasady redagowania tekstu</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Program do tworzenia prezentacji				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>z pomocą nauczyciela otwiera program do tworzenia prezentacji;</p> <p>samodzielnie dodaje tytuł i wypełnia pole tekstowe na slajdzie;</p> <p>z pomocą nauczyciela tworzy prostą prezentację składającą się z trzech slajdów</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>samodzielnie dodaje tytuł, wypełnia pole tekstowe oraz zmienia tło slajdu;</p> <p>z pomocą nauczyciela tworzy prostą prezentację składającą się z pięciu slajdów;</p> <p>z pomocą nauczyciela uruchamia pokaz slajdów</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie dodaje tytuł, wypełnia pole tekstowe oraz zmienia tło slajdu;</p> <p>samodzielnie tworzy prostą prezentację składającą się z pięciu slajdów;</p> <p>samodzielnie uruchamia pokaz slajdów;</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy estetyczną i spójną logicznie prezentację;</p> <p>samodzielnie wyszukuje w internecie informacje potrzebne do stworzenia prezentacji;</p> <p>prawidłowo rozmieszcza elementy na slajdzie;</p> <p>samodzielnie uruchamia pokaz slajdów;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy estetyczną i spójną logicznie prezentację;</p> <p>samodzielnie wyszukuje w internecie informacje potrzebne do stworzenia prezentacji;</p> <p>prawidłowo rozmieszcza elementy na slajdzie;</p> <p>samodzielnie uruchamia pokaz slajdów;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>korzysta z opcji niewspomnianych w podręczniku</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Arkusz kalkulacyjny				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>z pomocą nauczyciela otwiera arkusz kalkulacyjny;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>z pomocą nauczyciela wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>z pomocą nauczyciela formatuje tło komórki</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>rozdziela pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>rozumie, czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania arkusza kalkulacyjnego;</p> <p>samodzielnie wyjaśnia pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>samodzielnie wyjaśnia czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania arkusza kalkulacyjnego;</p> <p>samodzielnie wyjaśnia pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>samodzielnie wyjaśnia czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania arkusza kalkulacyjnego;</p> <p>samodzielnie wyjaśnia pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>samodzielnie wyjaśnia czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p> <p>korzysta z opcji niewspomnianych w podręczniku</p>

KLASA 5

I. Wprowadzenie				
Prawo i zasady bezpieczeństwa				
Na ocenę <b>2</b> uczeń:	Na ocenę <b>3</b> uczeń:	Na ocenę <b>4</b> uczeń:	Na ocenę <b>5</b> uczeń:	Na ocenę <b>6</b> uczeń:
<p>rozdziela zewnętrzne elementy komputera;</p> <p>wie, że pobieranie nielegalnie oprogramowania, gier, muzyki i filmów jest zabronione</p>	<p>samodzielnie wymienia zewnętrzne elementy komputera;</p> <p>samodzielnie wymienia niektóre spośród elementów budowy komputera;</p> <p>rozdziela środowisko tekstowe i graficzne;</p> <p>wie, że pobieranie nielegalnie oprogramowania, gier, muzyki i filmów jest zabronione</p>	<p>samodzielnie wymienia zewnętrzne elementy komputera;</p> <p>samodzielnie wymienia elementy budowy komputera;</p> <p>wyjaśnia czym jest system operacyjny;</p> <p>wyjaśnia różnicę między środowiskiem tekstowym i graficznym;</p> <p>wie, że pobieranie nielegalnie oprogramowania, gier, muzyki i filmów jest zabronione</p>	<p>samodzielnie opisuje elementy komputera;</p> <p>samodzielnie omawia przeznaczenie elementów budowy komputera;</p> <p>wyjaśnia czym jest system operacyjny;</p> <p>wyjaśnia różnicę między środowiskiem tekstowym i graficznym;</p> <p>omawia proces uruchomienia komputera;</p> <p>wie, że pobieranie nielegalnie oprogramowania, gier, muzyki i filmów jest zabronione</p>	<p>samodzielnie opisuje elementy komputera;</p> <p>samodzielnie omawia przeznaczenie elementów budowy komputera;</p> <p>wyjaśnia czym jest system operacyjny;</p> <p>wyjaśnia różnicę między środowiskiem tekstowym i graficznym;</p> <p>omawia proces uruchomienia komputera;</p> <p>rozszerza wspomniane zagadnienia o treści nie wspomniane w podręczniku;</p> <p>wie, że pobieranie nielegalnie oprogramowania, gier, muzyki i filmów jest zabronione</p>



OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

System binarny				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
<p>potrafi rozpoznać liczbę zapisaną w systemach binarnym oraz dziesiętnym;</p> <p>z pomocą nauczyciela dokonuje konwersji liczb z systemu dziesiętnego na binarny w zakresie do 16</p>	<p>potrafi rozpoznać liczbę zapisaną w systemach binarnym oraz dziesiętnym;</p> <p>samodzielnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 16;</p> <p>z pomocą nauczyciela dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 32;</p> <p>z pomocą nauczyciela dokonuje korekcji błędów zapisu liczby w systemie binarnym w zakresie do 16</p>	<p>wyjaśnia, czym dla komputera są stany 0 i 1;</p> <p>samodzielnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 32;</p> <p>z pomocą nauczyciela dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny większych od 32;</p> <p>samodzielnie dokonuje korekcji błędów zapisu liczby w systemie binarnym w zakresie do 16</p> <p>z pomocą nauczyciela dokonuje korekcji błędów zapisu liczby w systemie binarnym w zakresie do 16;</p>	<p>rozumie istotę liczb w informatyce;</p> <p>rozumie czym jest system pozycyjny;</p> <p>płynnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 64;</p> <p>samodzielnie dokonuje korekcji błędów zapisu liczby w systemie binarnym w zakresie do 32;</p>	<p>rozumie istotę liczb w informatyce;</p> <p>rozumie czym jest system pozycyjny;</p> <p>płynnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny większych od 64;</p> <p>samodzielnie dokonuje korekcji błędów zapisu liczby w systemie binarnym w zakresie przekraczającym 32</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

II. Programowanie				
Operacje arytmetyczne				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
<p>rozdziela instrukcje sumy i różnicy;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy</p>	<p>rozdziela instrukcje sumy i różnicy;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności</p>	<p>zna i potrafi objaśnić działanie instrukcji sumy i różnicy;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudniejsze, w tym zadania wymagające użycia pudełka pomocniczego</p>	<p>zna i potrafi objaśnić działanie instrukcji sumy i różnicy;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna i potrafi objaśnić działanie instrukcji sumy i różnicy;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p> objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Instrukcje warunkowe				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>rozdzieli sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą</p>	<p>rozdzieli sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające porównania między dwoma zmiennymi</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające porównania między dwoma zmiennymi</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności, w tym zadania wymagające porównania między dwoma zmiennymi;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p> objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Pętla pojedyncza</b>				
Na ocenę <b>2</b> uczniów:	Na ocenę <b>3</b> uczniów:	Na ocenę <b>4</b> uczniów:	Na ocenę <b>5</b> uczniów:	Na ocenę <b>6</b> uczniów:
<p>potrafi rozróżnić pętlę skończoną i nieskończoną;</p> <p>z pomocą nauczyciela tworzy prosty program realizujący pętlę pojedynczą</p>	<p>potrafi rozróżnić pętlę skończoną i nieskończoną;</p> <p>potrafi wskazać, która zmienna służy jako licznik;</p> <p>samodzielnie tworzy prosty program realizujący pętlę pojedynczą;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności</p>	<p>rozumie i potrafi wyjaśnić czym jest pętla pojedyncza;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>samodzielnie tworzy prosty i średniozaawansowany program wymagający użycia pętli pojedynczej;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudniejsze</p>	<p>rozumie i potrafi wyjaśnić czym jest pętla pojedyncza;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p>objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>rozumie i potrafi wyjaśnić czym jest pętla pojedyncza;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

III. Technologie informacyjne				
Edytor graficzny				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
z pomocą nauczyciela otwiera edytor graficzny; z pomocą nauczyciela dodaje zdjęcia do programu graficznego	samodzielnie otwiera edytor graficzny; samodzielnie dodaje zdjęcia do programu graficznego; z pomocą nauczyciela dokonuje prostej obróbki zdjęć w programie graficznym	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego do obróbki zdjęć; samodzielnie dodaje zdjęcia do programu graficznego; samodzielnie dokonuje prostej obróbki zdjęć w programie graficznym; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego do obróbki zdjęć; samodzielnie dodaje zdjęcia do programu graficznego; samodzielnie dokonuje prostej obróbki zdjęć w programie graficznym; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego do obróbki zdjęć; samodzielnie dodaje zdjęcia do programu graficznego; samodzielnie dokonuje prostej obróbki zdjęć w programie graficznym; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; przygotowuje rysunki na konkursy informatyczne

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Edytor tekstowy</b>				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela otwiera edytor tekstowy; z pomocą nauczyciela wkleja do tekstu ilustracje, napisy i kształty;	samodzielnie otwiera edytor tekstowy; samodzielnie wkleja do tekstu ilustracje, napisy i kształty; z pomocą nauczyciela wstawia listy punktowe oraz listy wielopoziomowe numerowane;	samodzielnie otwiera edytor tekstowy; potrafi podać przykłady zastosowania list wielopoziomowych w edytorze tekstowym; samodzielnie wkleja do tekstu ilustracje, napisy i kształty; z pomocą nauczyciela wstawia listy punktowe oraz listy wielopoziomowe numerowane; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem	samodzielnie otwiera edytor tekstowy; potrafi podać przykłady zastosowania list wielopoziomowych w edytorze tekstowym; samodzielnie wkleja do tekstu ilustracje, napisy i kształty; z pomocą nauczyciela wstawia listy punktowe oraz listy wielopoziomowe numerowane; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera edytor tekstowy; potrafi podać przykłady zastosowania list wielopoziomowych w edytorze tekstowym; samodzielnie wkleja do tekstu ilustracje, napisy i kształty; z pomocą nauczyciela wstawia listy punktowe oraz listy wielopoziomowe numerowane; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; samodzielnie pisze teksty prawidłowo stosując poznane dotychczas narzędzia

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Arkusz kalkulacyjny				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
<p>z pomocą nauczyciela otwiera arkusz kalkulacyjny;</p> <p>z pomocą nauczyciela dodaje i usuwa kolumny i wiersze;</p> <p>z pomocą nauczyciela zmienia szerokość kolumn i wysokość wierszy</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>samodzielnie dodaje i usuwa kolumny i wiersze;</p> <p>samodzielnie zmienia szerokość kolumn i wysokość wierszy;</p> <p>z pomocą nauczyciela formatuje tekst, scala i dzieli komórki;</p> <p>z pomocą nauczyciela tworzy tabelę na podstawie podanych informacji</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania tabel z danymi w arkuszu kalkulacyjnym;</p> <p>samodzielnie dodaje i usuwa kolumny i wiersze;</p> <p>samodzielnie zmienia szerokość kolumn i wysokość wierszy;</p> <p>samodzielnie formatuje tekst, scala i dzieli komórki;</p> <p>samodzielnie tworzy tabelę na podstawie podanych informacji;</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania tabel z danymi w arkuszu kalkulacyjnym;</p> <p>samodzielnie dodaje i usuwa kolumny i wiersze;</p> <p>samodzielnie zmienia szerokość kolumn i wysokość wierszy;</p> <p>samodzielnie formatuje tekst, scala i dzieli komórki;</p> <p>samodzielnie tworzy tabelę na podstawie podanych informacji;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania tabel z danymi w arkuszu kalkulacyjnym;</p> <p>samodzielnie dodaje i usuwa kolumny i wiersze;</p> <p>samodzielnie zmienia szerokość kolumn i wysokość wierszy;</p> <p>samodzielnie formatuje tekst, scala i dzieli komórki;</p> <p>samodzielnie tworzy tabelę na podstawie podanych informacji;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p> <p>korzysta z opcji niewspomnianych w podręczniku</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Program do tworzenia prezentacji</b>				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>z pomocą nauczyciela otwiera program do tworzenia prezentacji;</p> <p>samodzielnie tworzy prosty slajd, dodaje tytuł oraz ilustracje</p> <p>z pomocą nauczyciela tworzy prostą prezentację składającą się z trzech slajdów</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>samodzielnie dodaje tytuł, wypełnia pole tekstowe oraz zmienia tło slajdu;</p> <p>z pomocą nauczyciela tworzy prostą prezentację składającą się z pięciu slajdów;</p> <p>z pomocą nauczyciela dodaje przejścia między slajdami;</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy prostą prezentację składającą się z pięciu slajdów;</p> <p>samodzielnie dodaje przejścia między slajdami;</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy prostą prezentację składającą się z pięciu slajdów;</p> <p>samodzielnie dodaje przejścia między slajdami;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy estetyczną i spójną logicznie prezentację;</p> <p>samodzielnie wyszukuje w internecie informacje potrzebne do stworzenia prezentacji;</p> <p>samodzielnie dodaje przejścia między slajdami;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>korzysta z opcji niewspomnianych w podręczniku</p>
<b>Chmura</b>				
<p>z pomocą nauczyciela zakłada konto w wybranej usłudze oferującej pracę w chmurze</p>	<p>samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze;</p> <p>z pomocą nauczyciela nawiguje po platformie</p>	<p>samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze;</p> <p>potrafi podać przykłady zastosowania chmury;</p> <p>samodzielnie nawiguje po platformie;</p> <p>z pomocą nauczyciela udostępnia pliki w ramach platformy</p>	<p>samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze;</p> <p>potrafi podać przykłady zastosowania chmury;</p> <p>samodzielnie nawiguje po platformie;</p> <p>samodzielnie udostępnia pliki w ramach platformy</p>	<p>samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze;</p> <p>potrafi podać przykłady zastosowania chmury;</p> <p>samodzielnie nawiguje po platformie;</p> <p>samodzielnie udostępnia pliki w ramach platformy;</p> <p>samodzielnie korzysta z chmury w trakcie pracy nad projektem grupowym</p>



OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Scratch</b>				
Na ocenę <b>2</b> uczeń:	Na ocenę <b>3</b> uczeń:	Na ocenę <b>4</b> uczeń:	Na ocenę <b>5</b> uczeń:	Na ocenę <b>6</b> uczeń:
Z pomocą nauczyciela loguje się na platformę Scratch; Z pomocą nauczyciela steruje postacią po ekranie	Samodzielnie loguje się na platformę Scratch; Z pomocą nauczyciela steruje postacią po ekranie	Samodzielnie loguje się na platformę Scratch; samodzielnie steruje postacią po ekranie	Samodzielnie loguje się na platformę Scratch; samodzielnie steruje postacią po ekranie tworzy własne historyjki i realizuje je w języku Scratch	Samodzielnie loguje się na platformę Scratch; samodzielnie steruje postacią po ekranie tworzy własne historyjki i realizuje je w języku Scratch; samodzielnie tworzy skomplikowane animacje

KLASA 6

I. Wprowadzenie				
Prawo i zasady bezpieczeństwa				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela wymienia typy złośliwego oprogramowania	samodzielnie wymienia typy złośliwego oprogramowania	samodzielnie wymienia typy złośliwego oprogramowania i wyjaśnia czym się charakteryzują	samodzielnie wymienia typy złośliwego oprogramowania i wyjaśnia czym się charakteryzują; samodzielnie wyjaśnia zasady działania oprogramowania antywirusowego	samodzielnie wymienia typy złośliwego oprogramowania i wyjaśnia czym się charakteryzują; samodzielnie wyjaśnia zasady działania oprogramowania antywirusowego; rozszerza wspomniane zagadnienia o treści nie wspomniane w podręczniku;
System binarny				
potrafi rozpoznać liczbę zapisaną w systemach binarnym oraz dziesiętnym; samodzielnie dokonuje konwersji liczb z systemu dziesiętnego na binarny w zakresie do 16; z pomocą nauczyciela dokonuje rewersu liczby w systemie binarnym	potrafi rozpoznać liczbę zapisaną w systemach binarnym oraz dziesiętnym; samodzielnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 32; samodzielnie dokonuje rewersu liczby w systemie binarnym; z pomocą nauczyciela dodaje dwie liczby w systemie binarnym w zakresie do 16	wyjaśnia, czym dla komputera są stany 0 i 1; samodzielnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 32; samodzielnie dokonuje rewersu liczby w systemie binarnym; samodzielnie dodaje dwie liczby w systemie binarnym w zakresie do 16	rozumie istotę liczb w informatyce; rozumie czym jest system pozycyjny; płynnie dokonuje konwersji liczb z systemu dziesiętnego na binarny oraz z binarnego na dziesiętny w zakresie do 64; samodzielnie dokonuje rewersu liczby w systemie binarnym; samodzielnie dodaje dwie liczby w systemie binarnym w zakresie do 32	rozumie istotę liczb w informatyce oraz czym jest system pozycyjny; płynnie dokonuje konwersji liczb między system dziesiętnym a binarnym większych od 64; samodzielnie dokonuje rewersu liczby w systemie binarnym; rozumie czym jest porządek losowy i rosnący; samodzielnie dokonuje przeszukiwania binarnego na zbiorze uporządkowanym rosnąco; samodzielnie dodaje dwie liczby w systemie binarnym w zakresie przekraczającym 32
II. Programowanie				

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Operacje arytmetyczne				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
<p>rozdziela instrukcje sumy i różnicy;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy</p>	<p>rozdziela instrukcje sumy i różnicy;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności</p>	<p>zna i potrafi objaśnić działanie instrukcji sumy i różnicy;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji sumy i różnicy;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudniejsze, w tym zadania wymagające użycia pudełka pomocniczego</p>	<p>zna i potrafi objaśnić działanie instrukcji sumy i różnicy;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna i potrafi objaśnić działanie instrukcji sumy i różnicy;</p> <p>rozumie czym jest pudełko pomocnicze i kiedy należy je stosować;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p> objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Instrukcje warunkowe i warunki zagnieżdżone				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>rozróżnia sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą w programie z trzema zmiennymi</p>	<p>rozróżnia sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między dwoma zmiennymi;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające znalezienia największej oraz najmniejszej z trzech liczb</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p> <p>samodzielnie tworzy program znajdujący największą oraz najmniejszą z trzech wczytanych liczb;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające znalezienia liczby środkowej z trzech wczytanych liczb</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p>płynnie rozwiązuje zadania proste oraz średniej trudności, w tym zadania wymagające znalezienia liczby środkowej z trzech wczytanych liczb;</p> <p>objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p>płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Pętla pojedyncza i pętla w pętli</b>				
Na ocenę <b>2</b> uczeń:	Na ocenę <b>3</b> uczeń:	Na ocenę <b>4</b> uczeń:	Na ocenę <b>5</b> uczeń:	Na ocenę <b>6</b> uczeń:
<p>potrafi rozróżnić pętlę skończoną i nieskończoną;</p> <p>z pomocą nauczyciela tworzy prosty program realizujący pętlę pojedynczą;</p> <p>z pomocą nauczyciela implementuje algorytmy mnożenia oraz dzielenia</p>	<p>potrafi rozróżnić pętlę skończoną i nieskończoną;</p> <p>potrafi wskazać, która zmienna służy jako licznik;</p> <p>samodzielnie tworzy prosty program realizujący pętlę pojedynczą;</p> <p>z pomocą nauczyciela implementuje algorytmy mnożenia oraz dzielenia</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności</p>	<p>rozumie i potrafi wyjaśnić czym jest pętla pojedyncza;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>samodzielnie tworzy prosty i średniozaawansowany program wymagający użycia pętli pojedynczej;</p> <p>samodzielnie implementuje algorytmy mnożenia oraz dzielenia;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudniejsze, w tym zadania wymagające użycia pętli w pętli</p>	<p>rozumie i potrafi wyjaśnić czym jest pętla pojedyncza;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>wyjaśnia czym jest pętla w pętli i kiedy należy ją stosować;</p> <p>samodzielnie implementuje algorytmy mnożenia oraz dzielenia;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>rozumie i potrafi wyjaśnić czym jest pętla pojedyncza;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>wyjaśnia czym jest pętla w pętli i kiedy należy ją stosować;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p> objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

III. Technologie informacyjne				
Edytor graficzny				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
z pomocą nauczyciela otwiera edytor graficzny; z pomocą nauczyciela dodaje zdjęcia do programu graficznego	samodzielnie otwiera edytor graficzny; samodzielnie dodaje zdjęcia do programu graficznego; z pomocą nauczyciela dokonuje prostej obróbki zdjęć w programie graficznym	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego do obróbki zdjęć; samodzielnie dodaje zdjęcia do programu graficznego; samodzielnie dokonuje prostej obróbki zdjęć w programie graficznym; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego do obróbki zdjęć; samodzielnie dodaje zdjęcia do programu graficznego; samodzielnie dokonuje prostej obróbki zdjęć w programie graficznym; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego do obróbki zdjęć; samodzielnie dodaje zdjęcia do programu graficznego; samodzielnie dokonuje prostej obróbki zdjęć w programie graficznym; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; przygotowuje rysunki na konkursy informatyczne

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Edytor tekstowy</b>				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
z pomocą nauczyciela otwiera edytor tekstowy; z pomocą nauczyciela dodaje podpis pod ilustracją i tabelą	samodzielnie otwiera edytor tekstowy; samodzielnie dodaje podpis pod ilustracją i tabelą; z pomocą nauczyciela formatuje dokument, zmienia rozmiar i orientację strony, zmienia marginesy; z pomocą nauczyciela drukuje dokument	samodzielnie otwiera edytor tekstowy; potrafi podać przykłady zastosowania tabel w edytorze tekstowym; samodzielnie dodaje podpis pod ilustracją i tabelą; samodzielnie formatuje dokument, zmienia rozmiar i orientację strony, zmienia marginesy; samodzielnie drukuje dokument; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem;	samodzielnie otwiera edytor tekstowy; potrafi podać przykłady zastosowania tabel w edytorze tekstowym; samodzielnie dodaje podpis pod ilustracją i tabelą; samodzielnie formatuje dokument, zmienia rozmiar i orientację strony, zmienia marginesy; samodzielnie drukuje dokument; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera edytor tekstowy; potrafi podać przykłady zastosowania tabel w edytorze tekstowym; samodzielnie dodaje podpis pod ilustracją i tabelą; samodzielnie formatuje dokument, zmienia rozmiar i orientację strony, zmienia marginesy; samodzielnie drukuje dokument; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; samodzielnie pisze teksty prawidłowo stosując poznane dotychczas narzędzia

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

<b>Chmura</b>				
Na ocenę <b>2</b> uczniów:	Na ocenę <b>3</b> uczniów:	Na ocenę <b>4</b> uczniów:	Na ocenę <b>5</b> uczniów:	Na ocenę <b>6</b> uczniów:
z pomocą nauczyciela zakłada konto w wybranej usłudze oferującej pracę w chmurze	samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze; z pomocą nauczyciela nawiguje po platformie	samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze; potrafi podać przykłady zastosowania chmury; samodzielnie nawiguje po platformie; z pomocą nauczyciela korzysta z edytora tekstowego, programu do tworzenia prezentacji oraz arkusza kalkulacyjnego	samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze; potrafi podać przykłady zastosowania chmury; samodzielnie nawiguje po platformie; samodzielnie korzysta z edytora tekstowego, programu do tworzenia prezentacji oraz arkusza kalkulacyjnego	samodzielnie zakłada konto w wybranej usłudze oferującej pracę w chmurze; potrafi podać przykłady zastosowania chmury; samodzielnie nawiguje po platformie; samodzielnie korzysta z edytora tekstowego, programu do tworzenia prezentacji oraz arkusza kalkulacyjnego; samodzielnie korzysta z chmury w trakcie pracy nad projektem grupowym
<b>Arkusz kalkulacyjny</b>				
z pomocą nauczyciela otwiera arkusz kalkulacyjny; z pomocą nauczyciela wprowadza serię danych; z pomocą nauczyciela tworzy wykres liniowy	samodzielnie otwiera arkusz kalkulacyjny; samodzielnie wprowadza serię danych; samodzielnie tworzy wykres liniowy, słupkowy oraz kołowy; z pomocą nauczyciela korzysta z paska formuły	samodzielnie otwiera arkusz kalkulacyjny; potrafi podać przykłady zastosowania różnych wykresów w arkuszu kalkulacyjnym; samodzielnie wprowadza serię danych; samodzielnie tworzy wykres liniowy, słupkowy oraz kołowy; samodzielnie korzysta z paska formuły; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem	samodzielnie otwiera arkusz kalkulacyjny; potrafi podać przykłady zastosowania różnych wykresów w arkuszu kalkulacyjnym; samodzielnie wprowadza serię danych; samodzielnie tworzy wykres liniowy, słupkowy oraz kołowy; samodzielnie korzysta z paska formuły; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera arkusz kalkulacyjny; potrafi podać przykłady zastosowania różnych wykresów w arkuszu kalkulacyjnym; samodzielnie wprowadza serię danych; samodzielnie tworzy wykres liniowy, słupkowy oraz kołowy; samodzielnie korzysta z paska formuły; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności korzysta z opcji niewspomnianych w podręczniku



OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 4-6 I PROPOZYCJE METOD OCENIANIA

Program do tworzenia prezentacji				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>z pomocą nauczyciela otwiera program do tworzenia prezentacji;</p> <p>samodzielnie tworzy prosty slajd, dodaje tytuł oraz ilustracje;</p> <p>z pomocą nauczyciela wstawia tabelę oraz wykres do prezentacji</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>samodzielnie dodaje tytuł, wypełnia pole tekstowe oraz zmienia tło slajdu;</p> <p>samodzielnie wstawia tabelę oraz wykres do prezentacji;</p> <p>z pomocą nauczyciela tworzy prostą prezentację składającą się z pięciu slajdów;</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy prostą prezentację składającą się z pięciu slajdów;</p> <p>samodzielnie wstawia tabelę oraz wykres do prezentacji;</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem;</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy prostą prezentację składającą się z pięciu slajdów;</p> <p>samodzielnie wstawia tabelę oraz wykres do prezentacji;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera program do tworzenia prezentacji;</p> <p>potrafi podać przykłady zastosowania programu do tworzenia prezentacji;</p> <p>samodzielnie tworzy estetyczną i spójną logicznie prezentację;</p> <p>samodzielnie wyszukuje w internecie informacje potrzebne do stworzenia prezentacji;</p> <p>samodzielnie wstawia tabelę oraz wykres do prezentacji;</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>korzysta z opcji niewspomnianych w podręczniku</p>

## REALIZACJA TREŚCI PODSTAWY PROGRAMOWEJ W KLASACH 7-8

W tabeli przedstawiono informacje, w których klasach według programu *InstaKod*

Treści nauczania według podstawy programowej	Klasa 7	Klasa 8
<b>I. Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:</b>		
1) Formułuje problem w postaci specyfikacji (opisuje dane i wyniki) i wyróżnia kroki w algorytmicznym rozwiązywaniu problemów. Stosuje różne sposoby przedstawiania algorytmów, w tym w języku naturalnym, w postaci schematów blokowych, listy kroków.	+	+
2) Stosuje przy rozwiązywaniu problemów podstawowe algorytmy:		
a. na liczbach naturalnych: bada podzielność liczb, wyodrębnia cyfry danej liczby, przedstawia działanie algorytmu Euklidesa w obu wersjach iteracyjnych (z odejmowaniem i z resztą z dzielenia);	+	
b. wyszukiwania i porządkowania: wyszukuje element w zbiorze uporządkowanym i nieuporządkowanym oraz porządkuje elementy w zbiorze metodą przez proste wybieranie i zliczanie.		+
3) Przedstawia sposoby reprezentowania w komputerze wartości logicznych, liczb naturalnych (system binarny), znaków (kody ASCII) i tekstów.	+	+
4) Rozwija znajomość algorytmów, korzystając z dostępnego oprogramowania do demonstracji działania algorytmów, wykonuje eksperymenty z algorytmami z wykorzystaniem takiego oprogramowania dla różnych danych.	+	+
5) Prezentuje przykłady zastosowań informatyki w innych dziedzinach, w zakresie pojęć, obiektów oraz algorytmów.	+	+
<b>II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:</b>		
1) Projektuje, tworzy i testuje programy w procesie rozwiązywania problemów. W programach stosuje: instrukcje wejścia/wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe instrukcje iteracyjne, funkcje oraz zmienne i tablice. W szczególności programuje algorytmy z części I.2).	+	+
2) Projektuje, tworzy i testuje oprogramowanie sterujące robotem lub innym obiektem na ekranie lub w rzeczywistości.		+
3) Korzystając z aplikacji komputerowych przygotowuje dokumenty i prezentacje, także w chmurze, na potrzeby rozwiązywanych problemów i własnych prac z różnych dziedzin (przedmiotów), dostosowuje format i wygląd opracowań do ich treści i przeznaczenia, wykazując się przy tym umiejętnościami:		
a. tworzenia estetycznych kompozycji graficznych: tworzy kolaże, wykonuje zdjęcia i poddaje je obróbce zgodnie z przeznaczeniem, nagrywa krótkie filmy oraz poddaje je podstawowej obróbce cyfrowej;	+	+
b. tworzenia różnych dokumentów: formatuje i łączy teksty, wstawia symbole, obrazy, tabele, korzysta z szablonów dokumentów, dłuższe dokumenty dzieli na strony;	+	+

REALIZACJA TREŚCI PODSTAWY PROGRAMOWEJ W KLASACH 7-8

Treści nauczania według podstawy programowej	Klasa 7	Klasa 8
c. rozwiązywania zadań rachunkowych z programu nauczania dla szkoły podstawowej z różnych przedmiotów, z codziennego życia oraz implementacji wybranych algorytmów w arkuszu kalkulacyjnym: posługuje się adresami bezwzględnymi, względnymi i mieszanymi, podstawowymi funkcjami, tworzy serie danych, gromadzi dane i przedstawia je w postaci tabel i różnych typów wykresów, wykonuje podstawowe operacje na danych zgromadzonych w jednej tabeli, w tym porządkuje i filtruje dane;	+	+
d. tworzenia prezentacji multimedialnej wykorzystując tekst, grafikę, animację, dźwięk i film, stosuje hiperłącza;	+	+
e. tworzenia prostej strony internetowej zawierającej; tekst, grafikę, hiperłącza, stosuje przy tym podstawowe polecenia języka HTML.	+	+
4) Zapisuje efekty swojej pracy w różnych formatach i przygotowuje wydruki.	+	+
5) Wyszukuje w sieci informacje potrzebne do realizacji wykonywanego zadania, stosując złożone postaci zapytań i korzysta z zaawansowanych możliwości wyszukiwarek.	+	+
<b>III. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi. Uczeń:</b>		
1) Schematycznie przedstawia budowę i funkcjonowanie sieci komputerowej, szkolnej, domowej i sieci Internet.	+	
2) Rozwija umiejętności korzystania z różnych urządzeń do tworzenia elektronicznych wersji tekstów, obrazów, dźwięków, filmów i animacji.	+	+
3) Poprawnie posługuje się terminologią związaną z informatyką i technologią.	+	+
<b>IV. Rozwijanie kompetencji społecznych. Uczeń:</b>		
1) Bierze udział w różnych formach współpracy, jak: programowanie w parach lub w zespole, realizacja projektów, uczestnictwo w zorganizowanej grupie uczących się. Projektuje, tworzy i prezentuje efekty wspólnej pracy.	+	+
2) Ocenia krytycznie informacje i ich źródła, w szczególności w sieci, pod względem rzetelności i wiarygodności w odniesieniu do rzeczywistych sytuacji, docenia znaczenie otwartych zasobów w sieci i korzysta z nich.		+
3) Przedstawia główne etapy w historycznym rozwoju informatyki i technologii.	+	
4) Określa zakres kompetencji informatycznych, niezbędnych do wykonywania różnych zawodów, rozważa i dyskutuje wybór dalszego i pogłębionego kształcenia, również w zakresie informatyki.		+
<b>V. Prawo i zasady bezpieczeństwa. Uczeń:</b>		
1) Opisuje kwestie etyczne związane z wykorzystaniem komputerów i sieci komputerowych, takie jak: bezpieczeństwo, cyfrowa tożsamość, prywatność, własność intelektualna, równy dostęp do informacji i dzielenie się informacją.	+	
2) Postępuje etycznie w pracy z informacjami.	+	+
3) Rozróżnia typy licencji na oprogramowanie: oprogramowanie otwarte ( <i>open source</i> ), wolne ( <i>free</i> ), w domenie publicznej ( <i>public domain</i> ) oraz rozróżnia typy licencji na zasoby w sieci.	+	

## CELE EDUKACYJNE W KLASACH 7-8

### SZCZEGÓŁOWE CELE WYCHOWAWCZE:

Są tożsame z celami, które stawiane są na poziomie nauczania informatyki w klasach 4-6 (str. 8), zakładając, że są one uniwersalne. Realizacja tych celów na każdym poziomie nauczania uwzględnia wyniki osiągnięte na poprzednim poziomie. Dla przypomnienia wyliczono je poniżej.

- Ćwiczenie pamięci.
- Rozwijanie myślenia abstrakcyjnego.
- Rozwijanie dociekliwości.
- Ćwiczenie umiejętności czytania ze zrozumieniem.
- Rozwijanie umiejętności analizowania problemu i wyszukiwania kluczowych informacji.
- Ćwiczenie spójnego i precyzyjnego wyrażania myśli.
- Rozwijanie zainteresowań informatycznych.
- Kształtowanie umiejętności rozwiązywania zadań programistycznych.
- Kształcenie umiejętności klarownego przedstawiania swojego toku rozumowania.
- Rozwijanie potrzeby sprawdzania poprawności swojego toku rozumowania, wyciągania wniosków i poprawiania błędów.
- Wyrabianie potrzeby eksperymentowania z nowymi pojęciami.
- Rozwijanie samodzielności w zbieraniu informacji o danym temacie.
- Kształcenie nawyku systematycznej i zorganizowanej pracy.
- Rozwijanie potrzeby pracy w ciszy i skupieniu.
- Kształcenie kompetencji społecznych, w tym pracy zespołowej.
- Poszanowanie pracy innych.

## SZCZEGÓŁOWE CELE EDUKACYJNE:

## KLASA 7

## I. Wprowadzenie

- 1) Prawo i zasady bezpieczeństwa
  - Poznanie zagadnień związanych z prywatnością i tożsamością w internecie, własnością intelektualną i informacją.
  - Umiejętność skutecznego wyszukiwania informacji z pomocą wyszukiwarki internetowej, weryfikacja ich treści i źródeł.
- 2) Poznawanie zasad działania sieci komputerowej oraz różnych typów licencji
  - Poznanie budowy i zasad funkcjonowania różnych rodzajów sieci, w tym: sieci komputerowej, domowej, szkolnej i sieci Internet.
  - Przesyłanie plików między różnymi urządzeniami multimedialnymi.
  - Poznanie różnych typów licencji na oprogramowanie i zasoby w sieci, takich jak oprogramowanie open source.

## II. Kształtowanie pojęć informatycznych i rozwijanie umiejętności posługiwania się nimi

- 1) Wprowadzenie do programowania
  - Rozwijanie intuicji związanych z pojęciami informatycznymi poznanymi w klasach IV-VI.
  - Poznanie zasad tworzenia programu komputerowego w dydaktycznym środowisku programowania (Nianiolang).
  - Pisanie prostego programu przy użyciu tekstowego języka programowania.
  - Rozumienie pojęć wejścia i wyjścia z programu.
  - Posługiwanie się instrukcją wyjścia w celu wyświetlenia na konsoli komunikatów.
  - Posługiwanie się instrukcją wejścia w celu pobrania danych od użytkownika.
  - Kształtowanie pojęcia zmiennej oraz świadomości stanu wartości zmiennych w danej chwili.
  - Rozumienie różnicy między wyjściem stałym i zmiennym.
  - Rozumienie czym są i dlaczego są wymagane trzy różne etapy tworzenia programu: algorytm, implementacja i testowanie.
- 2) Wprowadzenie do algorytmów
  - Rozwijanie intuicji związanych z podstawowymi pojęciami informatyki: kolejność liniowa, instrukcja, algorytm.
  - Analiza problemu, opracowanie rozwiązania, przetestowanie rozwiązania dla przykładowych danych.
  - Implementacja rozwiązania w języku tekstowym i sprawdzanie rozwiązania problemu dla przykładowych danych.
  - Analiza błędów.
  - Objaśnianie przebiegu działania programów.
  - Rysowanie diagramów rozwiązań.
  - Rozumienie różnicy między algorytmami liniowymi i algorytmami z warunkami.
- 3) Operacje arytmetyczne na zmiennych
  - Poznanie sposobów rozwiązywania zadań arytmetycznych z wykorzystaniem komputera.
  - Stosowanie w programach instrukcji służących do wykonywania prostych operacji arytmetycznych na zmiennych. Dodawanie, odejmowanie, mnożenie, dzielenie oraz znajdowanie reszty z dzielenia na liczbach całkowitych.
  - Rozumienie i używanie pojęć związanych z arytmetyką: suma, różnica, iloczyn oraz iloraz.
  - Rozumienie potrzeby użycia zmiennej pomocniczej
- 4) Instrukcja warunkowa
  - Zapisywanie i ocena prawdziwości wyrażeń logicznych.
  - Wykorzystanie instrukcji warunkowej do podejmowania decyzji.
  - **Implementacja algorytmów znajdowania maksimum i minimum z dwóch liczb.**
  - **Implementacja algorytmu badania podzielności liczby.**
  - **Implementacja algorytmów znajdowania maksimum i minimum z trzech liczb.**

- **Implementacja algorytmu sortowania bąbelkowego trzech liczb.**
  - **Implementacja algorytmu znajdowania liczby środkowej z trzech liczb.**
  - Łączenie warunków.
- 5) Pętla pojedyncza oraz pętla w pętli
- Rozumienie różnicy między pętlą warunkową i iteracyjną.
  - Zapisywanie algorytmów do zadań wymagających użycia pętli.
  - Stosowanie pętli do modyfikacji danych pobranych od użytkownika.
  - **Implementacja algorytmu znajdowania minimum ze strumienia.**
  - **Implementacja algorytmu znajdowania maksimum ze strumienia.**
  - **Implementacja algorytmu wyodrębniania cyfr danej liczby.**
  - **Implementacja algorytmu naiwnego testu pierwszości.**
  - **Implementacja algorytmu Euklidesa w wersji z odejmowaniem.**
  - **Implementacja algorytmu Euklidesa w wersji z resztą z dzielenia.**
  - **Implementacja algorytmu znajdowania NWW przy pomocy algorytmu Euklidesa.**
  - Stosowanie pętli zagnieżdżonej do wypisywania wzorów na konsolę.
- 6) Tablice
- Rozumienie czym są struktury danych.
  - Wykorzystanie tablicy jako sposobu reprezentacji złożonych danych.
  - Wczytywanie i wypisywanie elementów tablicy.

### III. Poznanie technologii informacyjnych

- 1) Tworzenie rysunków za pomocą edytora graficznego
  - Edycja wybranych parametrów zdjęć z pomocą edytora graficznego.
  - Wykorzystanie elementów graficznych do edycji zdjęć.
  - Łączenie kilku zdjęć w jednym projekcie.
  - Użycie edytora graficznego do przygotowania projektu na lekcję plastyki.
- 2) Tworzenie dokumentów za pomocą edytora tekstu
  - Korzystanie z gotowych szablonów przy tworzeniu dokumentu tekstowego.
  - Łączenie kilku dokumentów tekstowych w jeden.
  - Umiejętne wykorzystanie symboli, tabeli, obrazów.
  - Formatowanie dokumentu tekstowego w zadany sposób.
- 3) Obliczenia w arkuszu kalkulacyjnym
  - Wykorzystanie różnych rodzajów wykresów do zobrazowania danych liczbowych.
  - Przedstawienie kilku serii danych na jednym wykresie.
  - Posługiwanie się funkcjami: Średnia, Suma.
  - Znajomość zagadnienia i umiejętność wykorzystania sortowania danych.
  - Posługiwanie się adresami względnymi, bezwzględnymi, mieszanymi.
  - Wykonywanie operacji matematycznych na całych zbiorach danych.
  - Zastosowanie arkusza kalkulacyjnego do rozwiązywania problemów z innych przedmiotów.
  - Użycie arkusza kalkulacyjnego do przygotowania projektu na lekcję fizyki.
- 4) Tworzenie prezentacji multimedialnych
  - Wykorzystanie dźwięku przy tworzeniu prezentacji multimedialnej.
  - Wykorzystanie filmu przy tworzeniu prezentacji multimedialnej.
  - Wykorzystanie prezentacji multimedialnej do przedstawienia wyników projektu z arkusza kalkulacyjnego.
- 5) Tworzenie stron internetowych w HTML
  - Poznanie podstawowych funkcji i zastosowań HTML.
  - Poznanie podstawowych elementów projektu w HTML i podgląd strony.
  - Znaczniki: przykłady, wykorzystanie do formatowania czcionki, łączenie znaczników.
  - Poznanie podstawowych zasad tworzenia stron internetowych.
  - Zmiana koloru tła, wstawienie i ustawienie ilustracji.
  - Użycie HTML do przygotowania projektu: własnej strony.

## KLASA 8

## I. Wprowadzenie

- 1) Historia i teoria informatyki
  - Zapoznanie się z głównymi etapami w historycznym rozwoju informatyki i technologii.
  - Zapoznanie się z terminologią związaną z informatyką i technologią oraz jej wykorzystanie.
  - Poznanie zastosowań informatyki w różnych dziedzinach życia.
  - Zapoznanie się z kompetencjami informatycznymi w różnych zawodach.

## II. Kształtowanie pojęć informatycznych i rozwijanie umiejętności posługiwania się nimi

- 1) Wprowadzenie do programowania
  - Rozwijanie intuicji związanej z pojęciami informatycznymi poznanymi w klasach IV-VII.
  - Poznanie zasad tworzenia programu komputerowego w dydaktycznym środowisku programowania (Nianiolang).
  - Pisanie prostego programu przy użyciu tekstowego języka programowania.
  - Rozumienie pojęć wejścia i wyjścia z programu.
  - Posługiwanie się instrukcją wyjścia w celu wyświetlenia na konsoli komunikatów.
  - Posługiwanie się instrukcją wejścia w celu pobrania danych od użytkownika.
  - Kształtowanie pojęcia zmiennej oraz świadomości stanu wartości zmiennych w danej chwili.
  - Rozumienie różnicy między wyjściem stałym i zmiennym.
  - Rozumienie czym są i dlaczego są wymagane trzy różne etapy tworzenia programu: algorytm, implementacja i testowanie.
- 2) Wprowadzenie do algorytmów
  - Rozwijanie intuicji związanej z podstawowymi pojęciami informatyki: kolejność liniowa, instrukcja, algorytm.
  - Analiza problemu, opracowanie rozwiązania, przetestowanie rozwiązania dla przykładowych danych.
  - Implementacja rozwiązania w języku tekstowym i sprawdzanie rozwiązania problemu dla przykładowych danych.
  - Analiza błędów.
  - Objaśnianie przebiegu działania programów.
  - Rysowanie diagramów rozwiązań.
  - Rozumienie różnicy między algorytmami liniowymi i algorytmami z warunkami.
- 3) Operacje arytmetyczne na zmiennych
  - Poznanie sposobów rozwiązywania zadań arytmetycznych z wykorzystaniem komputera.
  - Stosowanie w programach instrukcji służących do wykonywania prostych operacji arytmetycznych na zmiennych. Dodawanie, odejmowanie, mnożenie, dzielenie oraz znajdowanie reszty z dzielenia na liczbach całkowitych.
  - Rozumienie i używanie pojęć związanych z arytmetyką: suma, różnica, iloczyn oraz iloraz.
  - Rozumienie potrzeby użycia zmiennej pomocniczej.
- 4) Instrukcja warunkowa
  - Zapisywanie i ocena prawdziwości wyrażeń logicznych.
  - Wykorzystanie instrukcji warunkowej do podejmowania decyzji.
  - **Implementacja algorytmów znajdowania maksimum i minimum z dwóch liczb.**
  - **Implementacja algorytmu badania podzielności liczby.**
  - Łączenie warunków.
- 5) Funkcje
  - Zapoznanie się koncepcją funkcji w informatyce.
  - Wykorzystywanie funkcji do wykonywania prostych obliczeń.
  - Implementacja algorytmów przy użyciu funkcji.

- 6) Pętla pojedyncza oraz pętla w pętli
- Rozumienie różnicy między pętlą warunkową i iteracyjną.
  - Zapisywanie algorytmów do zadań wymagających użycia pętli.
  - Stosowanie pętli do wypisywania wzorów na konsolę oraz modyfikacji pobranych od użytkownika danych.
  - **Implementacja algorytmu znajdowania minimum ze strumienia.**
  - **Implementacja algorytmu znajdowania maksimum ze strumienia.**
  - **Implementacja algorytmu wyodrębniania cyfr danej liczby.**
  - **Implementacja algorytmu naiwnego testu pierwszości.**
  - **Implementacja algorytmu konwersji liczby dziesiętnej na liczbę binarną.**
  - **Implementacja algorytmu konwersji liczby binarnej na liczbę dziesiętną.**
  - Użycie pętli iteracyjnej do konwersji między systemami binarnym i dziesiętnym.
- 7) Tablice
- Rozumienie czym są struktury danych.
  - Wykorzystanie tablicy jako sposobu reprezentacji złożonych danych.
  - Wczytywanie i wypisywanie elementów tablicy.
  - **Implementacja algorytmu wyszukiwania liniowego.**
  - **Implementacja algorytmu znajdowania największego elementu tablicy.**
  - **Implementacja algorytmu znajdowania najmniejszego elementu tablicy.**
  - **Implementacja algorytmu znajdowania lidera.**
  - **Implementacja algorytmu sortowania przez wybieranie.**
  - **Implementacja algorytmu sortowania przez zliczanie.**
  - **Implementacja algorytmu sortowania bąbelkowego.**
  - **Implementacja algorytmu wyszukiwania binarnego.**
  - Przetwarzanie elementów tablic.
  - **Implementacja algorytmu Sita Eratostenesa.**
- 8) Napisy
- Zapoznanie się z kodem ASCII.
  - Wykonywanie operacji na napisach.
  - **Implementacja algorytmu szyfrowania napisu za pomocą szyfru Cezara.**
  - **Implementacja algorytmu badania czy napis jest palindromem.**
  - **Implementacja algorytmu badania czy dwa napisy są swoimi anagramami.**
  - **Implementacja algorytmu wyszukiwania słów podwójnych w napisie.**
  - **Implementacja algorytmu wyszukiwania wzorca w napisie.**

### III. Poznawanie technologii informacyjnych

- 1) Programowanie w wizualnym języku Scratch
  - Wybór jednej z dostępnych postaci duszka, stworzenie tła.
  - Umiejętność tworzenia programu realizującego grę, wykorzystującą sterowanie duszkiem i interakcję z otoczeniem.
- 2) Tworzenie filmów z wykorzystaniem różnych multimediów
  - Cyfrowa edycja klatek i fragmentów filmów.
  - Użycie edytora filmów do przygotowania projektu na język polski.
- 3) Tworzenie dokumentów za pomocą edytora tekstu
  - Stworzenie własnego szablonu dokumentu tekstowego.
  - Zastosowanie podziału dokumentu na strony i wstawienie numeracji stron.
  - Wykorzystanie następujących elementów: przypis, stopka, nagłówek.
- 4) Obliczenia w arkuszu kalkulacyjnym
  - Posługiwanie się funkcjami: ILOCZYN, JEŻELI, LICZ.JEŻELI.
  - Zastosowanie filtrowania danych, w tym funkcji: WYSZUKAJ.PIONOWO, ZNAJDŹ, INDEKS.
  - Wykorzystanie arkusza kalkulacyjnego do rozwiązywania problemów z życia codziennego.



- Użycie arkusza kalkulacyjnego do przygotowania projektu na lekcję matematyki.
- 5) Tworzenie prezentacji multimedialnych
- Wykorzystanie animacji przy tworzeniu prezentacji multimedialnej.
  - Wykorzystanie hiperłączy przy tworzeniu prezentacji multimedialnej.
  - Użycie prezentacji multimedialnej do podsumowania projektu z arkusza kalkulacyjnego.
- 6) Tworzenie stron internetowych w HTML
- Wstawienie tytułu, opisu, autora strony, zastosowanie komentarzy.
  - Poprawne wykorzystanie formatowania akapitów.
  - Zastosowanie odsyłaczy do innej strony lub własnej podstrony.
  - Wybór wzoru tła strony.
  - Wstawienie ramki, tabeli, formatowanie tabeli.
  - Wstawienie elementów multimedialnych: filmów, muzyki.
  - Użycie HTML do przygotowania projektu: własnej strony.

## MATERIAŁ NAUCZANIA W KLASACH 7-8

Materiał nauczania informatyki na poziomie każdej klasy podzielony jest na 3 uzupełniające się działy:

- I. **wprowadzenie**, w skład którego wchodzi zagadnienia z zakresu przestrzegania prawa i bezpieczeństwa korzystania z technologii informacyjnych i informacji dostępnych w sieci,
- II. **podstawy programowania**, w szczególności proces rozumienia, analizowania i rozwiązywania problemów (myślenia komputacyjnego)
- III. **technologie informacyjne** wspierające naukę oraz ich zastosowanie do realizacji projektów z innych przedmiotów szkolnych.

Program nauczania *InstaKod* został zbudowany w oparciu o zasadę spiralności. Treści, które uczniowie poznają w jednej klasie utrwalane i rozbudowywane są w klasie następnej. W ten sposób uczniowie posiadają głębsze rozumienie omawianych pojęć. Nauczyciele pracujący szybciej niż sugeruje to program, mogą bez problemu omawiać materiał z klasy następnej. Sugerowane jest przenoszenie uczniów najzdolniejszych do kolejnych poziomów klas po zrealizowaniu przez nich treści z poziomu klasy.

### KLASA 7

<b>I. Wprowadzenie</b>	
<b>Prawo i zasady bezpieczeństwa</b>	
Treści	Komentarze
Zagadnienia etyczne związane z wykorzystaniem komputerów	Rozwinięcie takich zagadnień jak: prywatność i cyfrowa tożsamość, własność intelektualna, równy dostęp do informacji i dzielenie się nią, etyczna praca z informacjami.
Wyszukiwanie informacji	Skuteczne wyszukiwanie wiarygodnych informacji w internecie. Inne funkcje wyszukiwarek. Krytyczne podejście do źródeł informacji.
<b>Sieci komputerowe</b>	
Treści	Komentarze
Sieć komputerowa	Budowa i funkcjonowanie różnych sieci komputerowych, w tym: domowej, szkolnej, internet
Dzielenie się plikami	Znane i nowe możliwości dzielenia się plikami między urządzeniami.
Licencje	Różne typy licencji na oprogramowanie i zasoby w sieci.

II. Programowanie	
Wprowadzenie, zmienna liczbowa, wejście, wyjście	
Treści	Komentarze
Edytor języka Nianiolang	Wprowadzanie do programowania w języku tekstowym. Zapoznanie się ze środowiskiem edytora języka Nianiolang.
Zmienna	Pojęcie zmiennej. Deklaracja oraz inicjalizacja zmiennej w języku Nianiolang.
Wejście i wyjście	Wczytywanie liczby do zmiennej. Wypisywanie napisów, liczb, oraz wzorów na konsoli.
Operacje arytmetyczne	
Treści	Komentarze
Dodawanie i odejmowanie stałych od zmiennej	Znajdowanie liczby, która jest od danej liczby o 3 większa, o 2 mniejsza, itd., dla jednej, dwóch i trzech liczb.
Obliczanie sumy i różnicy	Obliczanie wartości prostych wyrażeń arytmetycznych. Rozbudowywanie rozumienia stanu programu w oparciu o zmienną pomocniczą.
Iloczyn, iloraz oraz reszta z dzielenia	Obliczanie wartości prostych wyrażeń arytmetycznych. Rozbudowywanie rozumienia stanu programu w oparciu o zmienną pomocniczą.
Algorytm liniowy	Rozbudowanie rozumienia algorytmów liniowych.
Instrukcje warunkowe oraz warunki zagnieżdżone	
Treści	Komentarze
Instrukcja warunkowa z porównaniem między zmienną a stałą	Uzależnienie wykonania jednej instrukcji bądź bloku instrukcji i innej instrukcji bądź innego bloku instrukcji od spełnienia danego warunku. Rozwiązywanie prostych zadań tekstowych wymagających użycia instrukcji warunkowej.
Instrukcja warunkowa z porównaniem między dwiema zmiennymi	Uzależnienie wykonania instrukcji bądź bloku instrukcji od prawdziwości danego porównania między zmienną a stałą. Porównanie dwóch zmiennych w celu sprawdzenia czy są równe. <b>Algorytm znajdowania minimum z dwóch liczb. Algorytm znajdowania maksimum z dwóch liczb.</b>
Badanie podzielności	<b>Algorytm sprawdzania podzielności jednej liczby przez drugą.</b> Badanie parzystości liczby.
Łączenie warunków	Łączenie ze sobą dwóch instrukcji warunkowych. Sprawdzanie czy dana liczba należy do przedziału.
Zagnieżdżanie warunków	<b>Algorytm znajdowania minimum z trzech liczb. Algorytm znajdowania maksimum z trzech liczb. Algorytm sortowania bąbelkowego trzech liczb. Algorytm znajdowania liczby środkowej z trzech liczb.</b>
Algorytm z warunkami	Rozbudowanie rozumienia algorytmów z warunkami.

<b>Pętla pojedyncza i pętla w pętli</b>	
Treści	Komentarze
Pętla warunkowa While	Wprowadzenie do pętli na przykładzie pętli warunkowej. <b>Algorytm znajdowania minimum ze strumienia. Algorytm znajdowania maksimum ze strumienia. Algorytm wyodrębniania cyfr danej liczby. Algorytm naiwnego testu pierwszości.</b>
Licznik	Użycie zmiennej jako licznika wykonanych kroków iteracji.
Sposoby zakończenia iteracji	Różnice w zakończeniu iteracji, gdy liczba powtórzeń jest z góry określona lub zależna od warunku.
Algorytm Euklidesa, znajdowanie NWD	<b>Algorytm Euklidesa w wersji z odejmowaniem. Algorytm Euklidesa w wersji z resztą z dzielenia.</b>
Algorytmy znajdowania NWW	<b>Algorytm znajdowania NWW przy pomocy algorytmu Euklidesa.</b>
Pętla iteracyjna For	Różnice między pętlą warunkową, a pętlą iteracyjną.
Pętla w pętli	Wyświetlanie na konsoli wzorów (motywów graficznych) przy użyciu pętli zagnieżdżonej, w tym kwadratów, trójkątów i prostokątów.

<b>Wprowadzenie do tablic</b>	
Treści	Komentarze
Wprowadzenie do struktur danych	Wprowadzenie pojęcia struktury danych na przykładzie tablicy.
Operacje na tablicach	Wczytywanie danych do tablicy oraz wypisywanie elementów tablicy.

<b>III. Technologie informacyjne</b>	
<b>Chmura</b>	
Treści	Komentarze
Kolaż	Korzystanie z wielu zdjęć lub ilustracji to stworzenia swojej własnej ilustracji.
Nowe narzędzia	Narzędzia zaznaczania: zaznaczenie odręczne, inteligentne nożyce.
Edycja zdjęć	Obrót, skalowanie, odbicie, kontrast, balans kolorów filtry, warstwy.
<b>Edytor tekstowy</b>	
Treści	Komentarze
Szablony	Korzystanie z gotowych szablonów podczas tworzenia dokumentu
Formatowanie dokumentów	Formatowanie dokumentu w zadany sposób, łączenie dokumentów
Wstawianie elementów	Wstawianie symboli, tabeli, obrazków do dokumentu

<b>Arkusz kalkulacyjny</b>	
Treści	Komentarze
Wykresy	Przykłady nowych rodzajów wykresów, dopasowywanie wykresów do danych.
Serie danych	Tworzenie serii danych, przedstawianie kilku serii na jednym wykresie.
Funkcje	Funkcja Średnia, przypomnienie Suma. Znaczenie i wykorzystanie.
Sortowanie	Znaczenie i wykorzystanie sortowania do serii danych.
Formuły: adresy	Adresy względne, bezwzględne, mieszane (kopiowanie i zastrzeżenie formuł).
Zastosowanie arkusza do rozwiązywania problemów z innych przedmiotów	Zastosowanie narzędzi i zdobytej wiedzy do rozwiązania zadania związanego na przykład z analizą danych z doświadczenia wykonanego na lekcji fizyki.
<b>Program do tworzenia prezentacji</b>	
Treści	Komentarze
Dźwięk	Wstawianie dźwięku do prezentacji.
Film	Wstawianie filmu do prezentacji.
<b>HTML</b>	
Treści	Komentarze
Wstęp	Czym jest i do czego służy HTML. Podstawowe edytory.
Ramy strony	Podstawowe elementy każdej strony w kodzie źródłowym i ich znaczenie.
Tekst i uruchomienie	Wpisywanie tekstu i uruchamianie strony.
Znaczniki	Czym są znaczniki, jak je stosować, dobór koloru tła, rozmiaru i koloru czcionki za pomocą znaczników. Łączenie znaczników.
Cechy dobrej strony WWW	Podstawowe zasady tworzenia stron internetowych, cechy które powinny zawierać.
Obrazki/ilustracje	Wstawienie i ustalenie pozycji obrazka na stronie.

## KLASA 8

I. Wprowadzenie	
Komputer i historia informatyki	
Treści	Komentarze
Historia informatyki i technologii	Przedstawienie głównych etapów w historycznym rozwoju informatyki i technologii.
Fachowa terminologia	Wprowadzenie terminologii związanej z informatyką i technologią.
Informatyka w różnych dziedzinach i zawodach	Przedstawienie zastosowania informatyki w różnych dziedzinach życia oraz kompetencji informatycznych w różnych zawodach.

II. Programowanie	
Funkcje	
Treści	Komentarze
Wprowadzenie do funkcji	Pojęcie funkcji. Deklaracja funkcji w języku Nianiolang. Wywoływanie własnych funkcji w funkcji głównej programu.
Typy funkcji	Budowanie funkcji bez parametrów, funkcji przyjmujących parametry przekazane jako wartości oraz funkcji przyjmujących parametry przekazane przez referencję.
Pętla pojedyncza i pętla w pętli	
Treści	Komentarze
Pętla warunkowa While	Rozwinięcie rozumienia działania pętli na przykładzie pętli warunkowej. <b>Algorytm znajdowania minimum ze strumienia. Algorytm znajdowania maksimum ze strumienia. Algorytm wyodrębniania cyfr danej liczby. Algorytm naiwnego testu pierwszości.</b>
Licznik	Użycie zmiennej jako licznika wykonanych kroków iteracji.
Sposoby zakończenia iteracji	Różnice w zakończeniu iteracji, gdy liczba powtórzeń z góry określona lub zależna od warunku.
Konwersja między systemami liczbowymi	<b>Algorytm konwersji liczby dziesiętnej na liczbę binarną. Algorytm konwersji liczby binarnej na liczbę dziesiętną</b>
Pętla iteracyjna	Różnice między pętlą warunkową a pętlą iteracyjną.
Pętla w pętli	Wyświetlanie na konsoli nieskomplikowanych wzorów przy użyciu pętli zagnieżdżonej w tym kwadratów, trójkątów i prostokątów.
Konwersja między systemem binarnym i dziesiętnym	Użycie pętli do konwersji liczby zapisanej w systemie binarnym na system dziesiętny oraz liczby zapisanej w systemie dziesiętnym na system binarny.

<b>Tablice</b>	
Treści	Komentarze
Wprowadzenie do struktur danych	Wprowadzenie pojęcia struktury danych na przykładzie tablicy.
Operacje na tablicach	Wczytywanie danych do tablicy oraz wypisywanie elementów tablicy.
Przeszukiwanie tablicy	<b>Algorytm wyszukiwania liniowego. Algorytm znajdowania największego elementu tablicy. Algorytm znajdowania najmniejszego elementu tablicy. Algorytm znajdowania lidera. Algorytm wyszukiwania binarnego.</b>
Sortowanie elementów tablicy	<b>Algorytm sortowania przez wybieranie. Algorytm sortowania przez zliczanie. Algorytm sortowania bąbelkowego.</b>
Przetwarzanie elementów	Zwiększanie, zmniejszanie, mnożenie oraz dzielenie na konkretnych elementach tablicy. Dodawanie i odejmowanie tablic.
Generacja liczb pierwszych	<b>Algorytm „Sito Eratostenesa”</b>
<b>Napisy</b>	
Treści	Komentarze
Kod ASCII	Wprowadzenie do kodu ASCII. Znajdowanie wartości w tabeli ASCII dla podanego znaku. Znajdowanie znaku w tabeli ASCII dla podanej wartości. <b>Algorytm szyfrowania napisu za pomocą szyfru Cezara.</b>
Operacje na napisach	Wczytywanie i wypisywanie napisów. Wypisywanie konkretnych znaków z danego napisu. Sprawdzanie długości napisu.
Palindromy i anagramy	<b>Algorytm badania czy napis jest palindromem. Algorytm badania czy dwa napisy są swoimi anagramami.</b>
Złożone operacje	Odwracanie napisu. Zamiana małych liter na wielkie litery i na odwrót. <b>Algorytm wyszukiwania słów podwójnych w napisie. Algorytm wyszukiwania wzorca w napisie.</b>

<b>III. Technologie informacyjne</b>	
<b>Scratch</b>	
Treści	Komentarze
Tło i Duszek	Wybór tła i Duszka spośród dostępnych.
Nowi bohaterowie	Dodawanie dodatkowych ruchomych elementów poza duszkiem.
Gra	Stworzenie gry opierającej się na ruchu duszkiem na przykład gry, w której należy unikać spadających jabłek.
<b>Program do tworzenia filmów</b>	
Treści	Komentarze
Wstęp	Interfejs i podstawowe funkcje programu.
Edycja klatek filmu	Cyfrowa obróbka podstawowych parametrów zdjęć – klatek i dodanych fragmentów filmów.
Praca w grupach	Praca w grupach nad zadaniem projektem, filmem. Poznanie zalet pracy w grupie.

<b>Edytor tekstowy</b>	
Treści	Komentarze
Szablony	Tworzenie własnych szablonów
Formatowanie dokumentu	Podział dokumentu na strony, wstawianie: numeracji, przypisów, stopki i nagłówka
<b>Arkusz kalkulacyjny</b>	
Treści	Komentarze
Funkcje	Funkcja Iloczyn
Filtrowanie danych	Funkcje: WYSZUKAJ.PIONOWO, ZNAJDŹ.INDEKS
Funkcje warunkowe	Funkcje: JEŻELI, LICZ.JEŻELI
Przykłady zastosowania arkusza kalkulacyjnego do rozwiązywania problemów z życia codziennego	W postaci realizacji zadanego projektu.
<b>Program do tworzenia prezentacji</b>	
Treści	Komentarze
Animacje	Wstawianie animacji do prezentacji
Hiperłącza	Wstawianie hiperłączy do prezentacji
<b>HTML</b>	
Treści	Komentarze
Podstawowe elementy strony	Tytuł, opis i autor strony. Akapity w tekście. Komentarze.
Odsyłacze	Czym są i do czego służą. Odsyłacz do strony internetowej, odsyłacz do własnej podstrony.
Tło	Wybór własnego tła strony.
Tabele	Wstawienie tabel i ich formatowanie.
Filmy i muzyka	Wstawianie filmów i muzyki na stronę.
Ramki	Wstawianie ramek.



## ROZKŁAD MATERIAŁU W KLASACH 7-8

Poniżej zostały zamieszczone tabele przedstawiające bloki tematyczne realizowane w danej klasie wraz z liczbą godzin potrzebną na ich zrealizowanie. Dla każdej klasy przygotowano również szczegółowy rozkład materiału.

Do obliczeń przyjęto, że rok szkolny liczy około 190 dni lekcyjnych. Przewidując 1 godzinę zajęć informatyki tygodniowo otrzymujemy 38 lekcji informatyki rocznie. Biorąc pod uwagę, że w niektórych tygodniach lekcje nie będą mogły się odbyć ze względu na okoliczności losowe, wycieczki szkolne itp., na zrealizowanie materiału w każdej klasie zostały zarezerwowane **32** godziny.

### RAMOWY ROZKŁAD MATERIAŁU KLASA 7

Blok tematyczny	Liczba godzin
<b>I. Wprowadzenie</b>	
Prawo i zasady bezpieczeństwa	1
Sieć komputerowa, licencje na zasoby i oprogramowanie	1
<b>II. Programowanie</b>	
Wejście i wyjście	1
Operacje arytmetyczne	4
Instrukcja warunkowa i warunki zagnieżdżone	7
Pętla warunkowa While	6
Pętla iteracyjna For	2
Wprowadzenie do tablic	1
<b>III. Technologie informacyjne</b>	
Program graficzny GIMP	1
Edytor tekstowy	1
Arkusze kalkulacyjny	2
Prezentacje	1
HTML	3

## RAMOWY ROZKŁAD MATERIAŁU KLASA 8

<b>Blok tematyczny</b>	<b>Liczba godzin</b>
<b>IV. Wprowadzenie</b>	
Elementy teorii i historii informatyki	1
<b>V. Programowanie</b>	
Funkcje	1
Pętla warunkowa While	3
Pętla iteracyjna For	2
Tablice	8
Napisy	6
*Przeszukiwanie binarne	
<b>VI. Technologie informacyjne</b>	
Scratch	1
Edytor filmów	1
Edytor tekstowy	1
Arkusze kalkulacyjny	2
Prezentacje multimedialne	2
HTML	3

\*Temat nieobowiązkowy (wykraczający poza podstawę programową).

## SZCZEGÓŁOWY ROZKŁAD MATERIAŁU KLASA 7

Dział / Rozdział	Temat i opis	Liczba godzin
Dział I	Wprowadzenie	
Rozdział 1	Prawo i zasady bezpieczeństwa Własność intelektualna Równy dostęp do informacji i dzielenie się nią Etyczna praca z informacjami Wyszukiwanie informacji w internecie w tym krytyczne podejście do źródeł i samych informacji	1
Rozdział 2	Sieć komputerowa, licencje na zasoby i oprogramowanie Budowa i funkcjonowanie sieci komputerowej, domowej, szkolnej oraz internetu Dzielenie się plikami z różnych urządzeń (np. Chmura w telefonie i na komputerze) Różne typy licencji na oprogramowanie i zasoby w sieci	1
Dział II	Podstawy programowania	
Rozdział 1	Wejście i wyjście z programu, wprowadzenie	1
Rozdział 2	Operacje arytmetyczne, dodawanie, odejmowanie, mnożenie, dzielenie oraz reszta z dzielenia na 2-3 zmiennych	2
Rozdział 3	Operacje arytmetyczne, zadania tekstowe	2
Rozdział 4	Instrukcja warunkowa, wprowadzenie	1
Rozdział 5	Instrukcja warunkowa, badanie podzielności i łączenie warunków	2
Rozdział 6	Instrukcja warunkowa, zadania tekstowe	2
Rozdział 7	Pętla warunkowa While, wprowadzenie do pętli, pętle nieskończone	1
Rozdział 8	Pętla warunkowa While, wyodrębnianie cyfr danej liczby, sprawdzanie czy liczba jest pierwsza	2
Rozdział 9	Pętla warunkowa While, algorytm Euklidesa w obu wersjach iteracyjnych (z odejmowaniem i resztą z dzielenia)	1
Rozdział 10	Pętla iteracyjna For, rysowanie wzorków	2
Rozdział 11	Warunki zagnieżdżone, znajdowanie maksimum i minimum z trzech liczb	1
Rozdział 12	Warunki zagnieżdżone, znajdowanie liczby środkowej, sortowanie bąbelkowe	1
Rozdział 13	Pętla warunkowa While, zadania trudne, NWD i NWW	2
Rozdział 14	Wprowadzenie do tablic, wczytywanie i wypisywanie elementów	1
Dział III	Technologie informacyjne	
Rozdział 1	Program graficzny GIMP Przypomnienie z klas V i VI Bardziej zaawansowana obróbka własnych zdjęć Nowe, bardziej zaawansowane funkcje graficzne programu i ich wykorzystanie Projekt grupowy na lekcję plastyki, kolaż „Pory roku” projekt VII.1	1

ROZKŁAD MATERIAŁU W KLASACH 7-8

Rozdział 2	Edytor tekstowy Korzystanie z gotowych szablonów Formatowanie istniejącego dokumentu w zadany sposób Łączenie kilku tekstów w jeden Wstawianie symboli, tabeli, obrazów	1
Rozdział 3	Arkusze kalkulacyjne Przypomnienie podstawowych wykresów i ich zastosowanie, przykłady innych wykresów Dopasowanie wykresów do danych Przedstawienie kilku serii danych na jednym wykresie Funkcje: Średnia + przypomnienie Suma	1
	Arkusze kalkulacyjne Sortowanie danych w kolumnie lub wierszu Adresy względne, bezwzględne i mieszane Wykonywanie operacji na całych zbiorach danych Przykłady zastosowania arkusza do rozwiązywania problemów z innych przedmiotów, np. matematyka, fizyka Projekt między - przedmiotowy na lekcję fizyki „Analiza danych z doświadczenia” projekt VII.2	1
Rozdział 4	Prezentacje Wstawianie dźwięku i filmu do prezentacji Prezentacja podsumowująca projekt z arkusza kalkulacyjnego (dodać film z doświadczenia)	1
Rozdział 5	HTML Podstawowe informacje, zastosowanie i edytory Ramy dokumentu i znaczenie poszczególnych elementów Wpisywanie tekstu Zapis i uruchomienie strony	1
	HTML Znaczniki: przykłady i zastosowanie Rozmiar czcionki Kolor czcionki i tła Łączenie znaczników	1
	HTML Podstawowe zasady tworzenia dobrych stron internetowych Wstawienie obrazka Ustawienie obrazka na stronie Projekt „Strona o mnie” projekt VII.3.	1
	Podsumowanie i ocenianie	1
	<b>RAZEM</b>	<b>32</b>

## SZCZEGÓŁOWY ROZKŁAD MATERIAŁU KLASA 8

Dział / Rozdział	Temat i opis	Liczba godzin
Dział I	Wprowadzenie	
Rozdział 1	Elementy teorii i historii informatyki Główne etapy w historycznym rozwoju informatyki i technologii Terminologia związana z informatyką i technologią Zastosowania informatyki w innych dziedzinach Kompetencje informatyczne w różnych zawodach	1
Dział II	Podstawy programowania	
Rozdział 1	Rozdział 1. Proste funkcje	1
Rozdział 2	Rozdział 2. Pętla warunkowa While , powtórzenie	2
Rozdział 3	Rozdział 3. Pętla iteracyjna For, powtórzenie	2
Rozdział 4	Rozdział 5. Tablice,, wczytywanie i wypisywanie elementów	2
Rozdział 5	Rozdział 4. Pętla warunkowa While, konwersja między systemem binarnym a dziesiętnym	1
Rozdział 6	Rozdział 6. Tablice, przeszukiwanie elementów, sortowanie przez wybieranie i zliczanie	2
Rozdział 7	Rozdział 7. Tablice, przetwarzanie elementów	2
Rozdział 8	Rozdział 8. Tablica, zadania trudne, sito Eratostenesa	2
Rozdział 9	Rozdział 9. Napisy, wprowadzenie, kody ASCII	2
Rozdział 10	Rozdział 10. Napisy, Palindromy	2
Rozdział 11	Rozdział 11. Napisy, złożone operacje	2
	*Przeszukiwanie binarne	
Dział III	Technologie informacyjne	
Rozdział 1	Scratch Tworzenie bardziej zaawansowanych programów z interakcją z otoczeniem Gra/sterowanie duszkiem	1
Rozdział 2	Edytor filmów Interfejs programu, jego zastosowanie i możliwości Cyfrowa obróbka klatek i fragmentów filmów Projekt grupowy na lekcję języka polskiego „Scena z Małego Księcia” projekt VIII.1	1
Rozdział 3	Edytor tekstowy Tworzenie własnego szablonu Wstawienie numeracji stron, podział dokumentów na strony Przypisy, stopka, nagłówki	1
Rozdział 4	Arkusz kalkulacyjny Funkcja: ILOCZYN Filtrowanie danych, funkcje: WYSZUKAJ.PIONOWO, ZNAJDŹ.INDEKS	1
	Arkusz kalkulacyjny Funkcje: JEŻELI, LICZ.JEŻELI Przykłady zastosowania arkusza kalkulacyjnego do rozwiązywania problemów z życia codziennego Zaawansowany projekt „Prosta analiza finansów firmy A” projekt VIII.2.	1

ROZKŁAD MATERIAŁU W KLASACH 7-8

Rozdział 5	Prezentacje multimedialne Wstawianie animacji i hiperłączy do prezentacji	1
	Prezentacja podsumowująca projekt z arkusza kalkulacyjnego	1
Rozdział 6	HTML Tytuł, opis i autor strony Akapity i komentarze Czym są i do czego stosujemy odsyłacze	1
	HTML Odsyłacz do strony internetowej Odsyłacz do własnej podstrony Wybór własnego tła strony	
	HTML Wstawianie tabeli i jej formatowanie Wstawianie filmów i muzyki Wstawianie ramek	1
	Projekt strony „Moje hobby” projekt VIII.3.	1
	Podsumowanie i ocenianie	1
	<b>RAZEM</b>	<b>32</b>
	*Przeszukiwanie binarne	

\*Temat nieobowiązkowy (wykraczający poza podstawę programową).

## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA

Zadaniem nauczyciela jest realizowanie takiego zakresu programu, na jaki pozwalają indywidualne możliwości danego ucznia. Podczas lekcji informatyki należy przekazywać wiedzę w zrozumiały dla uczniów sposób, w szczególności podając przykłady praktycznego zastosowania omawianych tematów:

- pokazując konkretne zastosowania poznanych umiejętności do praktycznych zagadnień, wskazując korzyści z tych zastosowań w życiu codziennym, (np. projekty międzyprzedmiotowe z zastosowaniem technologii informatycznych),
- pokazując zależność pojmowania podstaw programowania z koniecznością rozumienia jak działają aplikacje komputerowe w ich podstawach, stanowiących narzędzia kontroli i skutecznego ich wykorzystania w przyszłej pracy,
- prezentując alfabetyzację informatyczną jako współczesny odpowiednik podstawowych umiejętności czytania ze zrozumieniem i liczenia.

Zrozumienie przez uczniów danego tematu osiągnięte jest poprzez ćwiczenia praktyczne. W zakresie podstaw programowania uczniowie rozwiązują problemy i zadania informatyczne w środowisku programistycznym, które dostępne jest na koncie ucznia na platformie instakod.pl. Ćwiczenia te nie tylko budują lepsze rozumienie zagadnień programistycznych oraz sprawność w posługiwaniu się językiem, ale również wspomagają umiejętność czytania ze zrozumieniem, koncentrację i wyobraźnię. Zadaniem nauczyciela jest omówienie na tablicy zadań przykładowych oraz zadań, które sprawiają uczniom szczególne trudności. Rozwiązania tych zadań powinny być budowane wspólnie z uczniami, w szczególności analiza i dyskusja nad błędami jest dobrym środkiem dydaktycznym.

Rozwiązywanie przykładów wspólnie z uczniami poprzez zadawanie prostych pytań nakierowujących pozwala im lepiej zrozumieć dane zagadnienie. W szczególności bardzo korzystne jest angażowanie uczniów najstarszych poprzez poprowadzenie ich do rozwiązania ścieżką dedukcji (dostosowując pytania do ich poziomu).

Nauka podstaw programowania powinna odbywać się dwutorowo: na kartce papieru (karcie pracy), bez możliwości interakcji z komputerem oraz przy wsparciu komputera i środowiska programistycznego.

Nauczyciel, w pierwszej kolejności, zapisuje na tablicy wspólnie z uczniami wymyślone rozwiązania, prezentując algorytmy i rozwiązania za pomocą różnorodnych diagramów i schematów. Pozwala to na wdrażanie uczniów do intencyjności działań, w odróżnieniu od pracy metodą prób i błędów (interakcji ze środowiskiem programistycznym). Równolegle, głębsze rozumienie wprowadzanych zagadnień ćwiczone może być przy pomocy papierowych kart pracy, gdzie uczniowie znajdują błędy w kodzie programu, analizują stany wartości początkowych i końcowych dla danego programu oraz rysują diagramy rozwiązań.

Uczniowie rozwiązują zestawy zadań informatycznych dostępnych na platformie InstaKod w wizualnym środowisku informatycznym. Przechodzą przez ścieżkę pełnego rozwiązania: analizę i zrozumienie treści zadania, poprzez przemyślenie prawidłowego wyjścia dla konkretnych danych wejściowych, zapisanie rozwiązania w formie diagramu, sprawdzenie jego skuteczności, implementację, automatyczne testowanie, analizę wyników testów oraz korektę błędów.

Praca z podręcznikiem jest również bardzo korzystna. Kształtuje umiejętność czytania ze zrozumieniem oraz wyszukiwania najistotniejszych informacji. Umiejętność przeanalizowania w podręczniku prezentowanych przez nauczyciela na lekcji rozwiązań przykładów, pozwala uczniom na pracę we własnym tempie, uczniom słabszym na powtórny, samodzielny już analizę przykładów, a uczniom zdolnym na samodzielną pracę nad kolejnymi tematami.

Praca w grupach pozwala uczniom budować umiejętności społeczne. W szczególności dotyczy to grupowych projektów międzyprzedmiotowych.

## KLASA 7

<b>I. Wprowadzenie</b>	
<b>Prawo i zasady bezpieczeństwa</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• Omawia zagadnienia etyczne związane z wykorzystaniem komputerów</li> <li>• pokazuje jak wyszukiwać informacje w sieci</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z etycznymi zagadnieniami związanymi z wykorzystaniem komputerów</li> <li>• wyszukują informacje w internecie, krytycznie podchodząc do źródeł</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

<b>Sieci komputerowe</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia czym jest sieć komputerowa</li> <li>• omawia budowę i funkcjonowanie różnych sieci komputerowych</li> <li>• pokazuje jak dzielić się plikami między urządzeniami w danej sieci</li> <li>• omawia różne typy licencji na oprogramowania</li> <li>• pokazuje jak korzystać z pola tekstowego</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z pojęciem sieci komputerowej</li> <li>• zapoznają się z budową i funkcjonowaniem różnych sieci komputerowych</li> <li>• dzielą się plikami korzystając z lokalnej sieci</li> <li>• zapoznają się z różnymi typami licencji na oprogramowania</li> <li>• samodzielnie wykonują projekt na lekcję plastyki</li> </ul>

<b>II. Programowanie</b>	
<b>Wprowadzenie, zmienna liczbowa, wejście, wyjście</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje uczniom środowisko programistyczne języka Nianiolang</li> <li>• wyjaśnia różnice między językiem wizualnym i tekstowym</li> <li>• pokazuje na przykładzie strukturę programu w języku Nianiolang</li> <li>• przypomina pojęcia wejścia i wyjścia</li> <li>• pokazuje na przykładzie instrukcje interakcji z konsolą w języku Nianiolang</li> <li>• zleca uczniom rozwiązanie zadań z wypisywaniem napisów na konsolę</li> <li>• przypomina pojęcie zmiennej</li> <li>• pokazuje na przykładzie sposoby deklaracji i inicjalizacji zmiennych w języku Nianiolang</li> <li>• zleca uczniom rozwiązanie zadań z wczytywaniem i wypisywaniem liczb oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• poznają środowisko programistyczne języka Nianiolang</li> <li>• przypominają sobie pojęcia wejścia i wyjścia</li> <li>• objaśniają działanie poznanych instrukcji</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują na papierze karty pracy mające utrwalić poznane koncepcje oraz instrukcje</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• przypominają sobie pojęcie zmiennej</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów z lekcji</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>



Operacje arytmetyczne	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje 5 operatorów arytmetycznych w języku Nianiolang</li> <li>• rozwiązuje na tablicy przykładowe zadania z użyciem każdego z operatorów arytmetycznych</li> <li>• wyjaśnia czym jest algorytm liniowy</li> <li>• zleca uczniom rozwiązanie zadań z użyciem operatorów arytmetycznych oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• przypomina potrzebę korzystania ze zmiennej pomocniczej przy bardziej złożonych obliczeniach</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji dodawania i odejmowania oraz zmiennej pomocniczej</li> <li>• zleca uczniom rozwiązanie zadań z użyciem operatorów arytmetycznych i zmiennej pomocniczej oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• poznają 5 operatorów arytmetycznych w języku Nianiolang</li> <li>• objaśniają działanie poznanych instrukcji</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do przykładów, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• poznają koncepcje zmiennej pomocniczej</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie w zeszytach rysują liniowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów z lekcji</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

Instrukcje warunkowe oraz warunki zagnieżdżone	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje instrukcję warunkową w języku Nianiolang</li> <li>• przypomina czym jest Algorytm z warunkami</li> <li>• przypomina różnicę między algorytmem liniowym i drzewkowym</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem instrukcji warunkowej dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z użyciem instrukcji warunkowej dla dwóch warunków oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• poznają instrukcję warunkową w języku Nianiolang</li> <li>• objaśniają działanie instrukcji warunkowej</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się ze sposobem rozwiązywania zadań z zagnieżdżonymi warunkami</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> </ul>

## SPOSOBY OSIĄGANIA CELÓW KSZTAŁCENIA I WYCHOWANIA W KLASACH 7-8

<ul style="list-style-type: none"> <li>• pokazuje jak rysować diagram drzewkowy dla warunków zagnieżdżonych</li> <li>• rozwiązuje na tablicy przykładowe zadanie z warunkami zagnieżdżonymi dla warunku między dwiema zmiennymi</li> <li>• zleca uczniom rozwiązanie zadań z warunkami zagnieżdżonymi oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>
--	--

<b>Pętla pojedyncza i pętla w pętli</b>	
<b>Czynności nauczyciela</b>	<b>Czynności uczniów</b>
<ul style="list-style-type: none"> <li>• pokazuje zasadę działania pętli warunkowej While w języku Nianiolang</li> <li>• objaśnia jak wygląda Algorytm z warunkami dla pętli pojedynczej</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą warunkową While</li> <li>• zleca uczniom rozwiązanie zadań z pętli warunkowej While oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia na tablicy implementację algorytmu Euklidesa przy pomocy pętli warunkowej While</li> <li>• zleca uczniom rozwiązanie zadań wymagających wykorzystanie algorytmu Euklidesa oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• pokazuje zasadę działania pętli iteracyjnej For w języku Nianiolang</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą iteracyjną For</li> <li>• zleca uczniom rozwiązanie zadań z pętli iteracyjnej For oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• wyjaśnia zasadę konstrukcji pętli w pętli w języku Nianiolang</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą w pętli</li> <li>• zleca uczniom rozwiązanie zadań z pętlą w pętli oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia na tablicy implementację algorytmów znajdowania NWD i NWW przy pomocy pętli warunkowej While</li> <li>• zleca uczniom rozwiązanie zadań wymagających wykorzystanie algorytmów znajdowania NWD i NWW oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• poznają zasadę działania pętli warunkowej While w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli warunkowej While</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z realizacją algorytmu Euklidesa przy pomocy pętli</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznany algorytm</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• poznają zasadę działania pętli iteracyjnej For w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli iteracyjnej For</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> </ul>

	<ul style="list-style-type: none"> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się ze sposobem konstrukcji pętli w pętli w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli w pętli</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z realizacją algorytmów znajdowania NWD i NWW przy pomocy pętli</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane algorytmy</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>
--	---

<b>Wprowadzenie do tablic</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia na przykładzie tablic czym są struktury danych</li> <li>• pokazuje w jaki sposób wczytywać dane do tablicy i jak wpisywać elementy tablicy na konsolę</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem tablic</li> <li>• zleca uczniom rozwiązanie zadań z użyciem tablic oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• zapoznają się z pojęciami: struktury danych, tablice</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję tablicy</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

**III. Technologie informacyjne****Edytor graficzny**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina użyteczność edytora graficznego</li> <li>• pokazuje jak korzystać z narzędzi zaznaczania</li> <li>• pokazuje jak edytować zdjęcia</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję plastyki</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą rysunki, korzystając z narzędzi zaznaczania</li> <li>• edytują zdjęcia</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję plastyki</li> </ul>

**Edytor tekstowy**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina użyteczność edytora tekstowego</li> <li>• pokazuje jak korzystać z gotowego szablonu</li> <li>• pokazuje jak formatować dokumenty</li> <li>• pokazuje jak wstawiać elementy (symbole, tabele, obrazki)</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• korzystają z gotowego szablonu</li> <li>• formatują dokumenty</li> <li>• wstawiają elementy (symbole, tabele, obrazki) do dokumentu tekstowego</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

**Arkusz kalkulacyjny**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina użyteczność arkusza kalkulacyjnego</li> <li>• pokazuje jak tworzyć różne rodzaje wykresów</li> <li>• pokazuje jak tworzyć serie danych</li> <li>• pokazuje działanie funkcji ŚREDNIA oraz SUMA</li> <li>• wyjaśnia znaczenie i pokazuje jak wykorzystywać sortowanie serii danych</li> <li>• wyjaśnia różnicę między adresami względnymi, bezwzględnymi i mieszanymi</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> <li>• zleca wykonanie projektu na lekcję fizyki</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą różne rodzaje wykresów</li> <li>• tworzą serie danych</li> <li>• korzystają z funkcji ŚREDNIA oraz SUMA</li> <li>• sortują dane</li> <li>• poprawnie stosują adresy względne, bezwzględne i mieszane</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję fizyki</li> </ul>

**Program do tworzenia prezentacji**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina użyteczność programu do tworzenia prezentacji</li> <li>• pokazuje jak dodawać dźwięk do prezentacji</li> <li>• pokazuje jak dodawać film do prezentacji</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• dodają dźwięk do prezentacji</li> <li>• dodają film do prezentacji</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

<b>HTML</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia czym jest HTML</li> <li>• pokazuje w kodzie źródłowym podstawowe elementy każdej strony</li> <li>• pokazuje jak dodawać tekst do strony</li> <li>• pokazuje jak stosować znaczniki</li> <li>• omawia cechy dobrej strony WWW</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą prostą stronę WWW</li> <li>• dodają tekst do strony</li> <li>• dodają znaczniki do strony</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

## KLASA 8

<b>I. Wprowadzenie</b>	
<b>Komputer i historia informatyki</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• omawia główne etapy w historii informatyki</li> <li>• wprowadza terminologię związaną z informatyką</li> <li>• omawia zastosowania informatyki w różnych dziedzinach życia oraz kompetencji informatycznych w różnych zawodach</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• samodzielnie wykonują projekt na lekcję plastyki</li> </ul>

<b>II. Programowanie</b>	
<b>Funkcje</b>	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wprowadza pojęcie funkcji</li> <li>• wyjaśnia użyteczność dzielenia złożonych zadań na mniejsze funkcje</li> <li>• pokazuje w jaki sposób zadeklarować i wywołać funkcję w języku Nianiolang</li> <li>• rozwiązuje na tablicy przykład zadania z użyciem funkcji</li> <li>• wyjaśnia różnice między różnymi typami funkcji</li> <li>• zleca uczniom rozwiązanie zadań z użyciem funkcji oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• poznają pojęcie funkcji</li> <li>• objaśniają działanie poznanych instrukcji</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane koncepcje oraz instrukcje</li> <li>• samodzielnie rozwiązują na komputerach zadania analogiczne do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

Pętla pojedyncza i pętla w pętli	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina zasadę działania pętli warunkowej While w języku Nianiolang</li> <li>• objaśnia jak wygląda Algorytm z warunkami dla pętli pojedynczej</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą warunkową While</li> <li>• zleca uczniom rozwiązanie zadań z pętli warunkowej While oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• przypomina zasadę działania pętli iteracyjnej For w języku Nianiolang</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą iteracyjną For</li> <li>• zleca uczniom rozwiązanie zadań z pętli iteracyjnej For oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• przypomina zasadę konstrukcji pętli w pętli w języku Nianiolang</li> <li>• rozwiązuje na tablicy przykładowe zadanie z pętlą w pętli</li> <li>• zleca uczniom rozwiązanie zadań z pętlą w pętli oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia na tablicy sposób konwersji liczb między systemem binarnym i dziesiętnym w języku Nianiolang</li> <li>• zleca uczniom rozwiązanie zadań z konwersją między systemami binarnym i dziesiętnym oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• przypominają sobie zasadę działania pętli warunkowej While w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli warunkowej While</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• przypominają sobie zasadę działania pętli iteracyjnej For w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli iteracyjnej For</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• przypominają sobie sposoby konstrukcji pętli w pętli w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję pętli w pętli</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się ze sposobem konwersji liczb między systemem binarnym i dziesiętnym w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane algorytmy</li> <li>• samodzielnie w zeszytach rysują drzewkowe algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>



Tablice	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• przypomina na przykładzie tablic czym są struktury danych</li> <li>• przypomina w jaki sposób wczytywać dane do tablicy i jak wpisywać elementy tablicy na konsolę</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem tablic</li> <li>• zleca uczniom rozwiązanie zadań z użyciem tablic oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia na tablicy implementacje algorytmów sortowania przez wybieranie i sortowania przez zliczanie</li> <li>• zleca uczniom rozwiązanie zadań wymagających wykorzystanie algorytmów sortowania przez zliczanie i sortowania przez wybieranie oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• pokazuje w jaki sposób przetwarzać elementy tablicy</li> <li>• rozwiązuje na tablicy przykładowe zadanie z przetwarzaniem elementów tablicy</li> <li>• zleca uczniom rozwiązanie zadań wymagające przetworzenia elementów tablicy oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia na tablicy implementacje algorytmu „Sito Eratostenesa”</li> <li>• zleca uczniom rozwiązanie zadań wymagających wykorzystanie algorytmu sita Eratostenesa oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• przypominają sobie pojęcia: struktury danych, tablice</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję tablicy</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z implementacją algorytmów sortowania przez wybieranie w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane algorytmy</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• poznają sposoby przetwarzania elementów tablicy w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające sposoby przetwarzania elementów tablicy w języku Nianiolang</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z algorytmem „Sito Eratostenesa”</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznany algorytm</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>

Napisy	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia czym jest kod ASCII</li> <li>• rozwiązuje na tablicy przykładowe zadanie, w którym znajduje wartość ASCII odpowiadającą danemu znakowi</li> <li>• zleca uczniom rozwiązanie zadań, które wymagają znalezienia wartości ASCII odpowiadającej danemu znakowi oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• pokazuje podstawowe operacje na napisach</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem napisów</li> <li>• zleca uczniom rozwiązanie zadań z użyciem napisów oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• wprowadza pojęcie palindromu</li> <li>• rozwiązuje na tablicy przykładowe zadanie, w którym sprawdza czy dany napis jest palindromem</li> <li>• zleca uczniom rozwiązanie zadań wymagających sprawdzenia czy dany napis jest palindromem oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> <li>• omawia na tablicy złożone operacje na napisach</li> <li>• rozwiązuje na tablicy przykładowe zadanie z użyciem złożonych operacji na napisach</li> <li>• zleca uczniom rozwiązanie zadań z użyciem złożonych operacji na napisach oraz przetestowanie swoich rozwiązań dla przykładowych danych</li> </ul>	<ul style="list-style-type: none"> <li>• poznają kod ASCII</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające koncepcję kodu ASCII</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się z podstawowymi operacjami na Napisach w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane instrukcje</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• poznają termin palindrom</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznaną koncepcję</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> <li>• zapoznają się ze złożonymi operacjami na Napisach w języku Nianiolang</li> <li>• analizują sposób rozwiązania przykładu wybranego przez nauczyciela</li> <li>• samodzielnie rozwiązują w karcie pracy zadania utrwalające poznane instrukcje</li> <li>• samodzielnie w zeszytach rysują algorytmy rozwiązania zadań analogicznych do przykładów z lekcji, a następnie implementują swoje rozwiązania na komputerach</li> <li>• testują swoje rozwiązania dla przykładowych danych</li> <li>• objaśniają krok po kroku działanie swoich programów</li> </ul>



**III. Technologie informacyjne****Scratch**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak wybrać tło i duszka</li> <li>• pokazuje jak dodać innych ruchomych bohaterów poza duszkiem</li> <li>• pokazuje jak stworzyć prostą grę</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• wybierają tło i duszka</li> <li>• dodają innych ruchomych bohaterów</li> <li>• tworzą prostą grę</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

**Program do tworzenia filmów**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• wyjaśnia użyteczność programu do tworzenia filmów</li> <li>• omawia podstawowe funkcje programu</li> <li>• pokazuje jak edytować klatki filmu</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą prosty film</li> <li>• edytują klatki filmu</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

**Edytor tekstowy**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak dodawać ilustracje</li> <li>• pokazuje jak tworzyć listy punktowe oraz wielopoziomowe listy numerowane</li> <li>• pokazuje jak dodawać tabele</li> <li>• zleca wykonanie projektu na lekcję języka polskiego</li> </ul>	<ul style="list-style-type: none"> <li>• dodają ilustracje</li> <li>• tworzą listy punktowe i wielopoziomowe</li> <li>• dodają tabele</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> <li>• samodzielnie wykonują projekt na lekcję języka polskiego</li> </ul>

**Arkusz kalkulacyjny**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak korzystać z funkcji ILOCZYN</li> <li>• pokazuje jak filtrować dane z użyciem funkcji WYSZUKAJ.PIONOWO oraz ZNAJDŹ.INDEKS</li> <li>• pokazuje jak działają funkcje warunkowe na przykładzie funkcji JEŻELI oraz LICZ.JEŻELI</li> <li>• omawia zastosowania arkusza kalkulacyjnego do rozwiązywania problemów z życia codziennego</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• korzystają z funkcji ILOCZYN</li> <li>• filtrują dane przy użyciu funkcji WYSZUKAJ.PIONOWO oraz ZNAJDŹ.INDEKS</li> <li>• korzystają z funkcji JEŻELI oraz LICZ.JEŻELI</li> <li>• samodzielnie wykonują projekt na lekcję geografii</li> </ul>

**Program do tworzenia prezentacji**

Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje jak dodać animacje do prezentacji</li> <li>• pokazuje jak wstawić hiperłącza do prezentacji</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• dodają animacje do prezentacji</li> <li>• wstawiają hiperłącza do prezentacji</li> <li>• wykonują ćwiczenia z podręcznika</li> </ul>

HTML	
Czynności nauczyciela	Czynności uczniów
<ul style="list-style-type: none"> <li>• pokazuje w kodzie źródłowym podstawowe elementy każdej strony (tytuł, opis oraz autora strony, akapity w tekście, komentarze)</li> <li>• pokazuje jak dodawać odsyłacze</li> <li>• pokazuje jak zmieniać tło</li> <li>• pokazuje jak wstawiać tabele</li> <li>• pokazuje jak dodawać filmy i muzykę</li> <li>• pokazuje jak wstawiać ramki</li> <li>• zleca wykonywanie ćwiczeń opisanych w podręczniku</li> </ul>	<ul style="list-style-type: none"> <li>• tworzą prostą stronę WWW wraz z podstawowymi elementami (tytuł, opis oraz autora strony, akapity w tekście, komentarze)</li> <li>• dodają odsyłacze</li> <li>• zmieniają tło</li> <li>• wstawiają tabele</li> <li>• dodają film i muzykę</li> <li>• wstawiają ramki</li> <li>• samodzielnie wykonują ćwiczenia z podręcznika</li> </ul>

## OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

Uwagi ogólne dotyczące oceniania, wynikające bezpośrednio z rozporządzenia MEN, a przedstawione na str. 46 stosują się również do klas 7-8, dla przypomnienia przedstawiono poniżej zasady wyznaczania proponowanej oceny z informatyki dla uczniów realizujących program *InstaKod*. Tabele, które umieszczono poniżej tekstu, uwzględniające te zasady mają charakter orientacyjny, mogą być pomocne przy formułowaniu oceny ucznia.

Na ocenę dopuszczającą zasługuje uczeń, który:

- wykonał minimum wysiłku w zrozumieniu najprostszych zagadnień omawianych w danym temacie.
- w zakresie podstaw programowania:
  - rozwiązał na swoim koncie zadania przykładowe, mając dostęp do wzorcowych rozwiązań.
  - potrzebuje pomocy nauczyciela w rozwiązywaniu prostych zadań, nie mając dostępu do wzorcowych rozwiązań.

Na ocenę dostateczną zasługuje uczeń, który:

- wykonał wysiłek w zrozumieniu najprostszych zagadnień omawianych w danym temacie.
- w zakresie podstaw programowania
  - rozwiązał na swoim koncie zadania przykładowe, mając dostęp do wzorcowych rozwiązań.
  - samodzielnie rozwiązał na swoim koncie najprostsze zadania (zadania z poziomu 1).

Na ocenę dobrą zasługuje uczeń, który:

- rozumie i potrafi własnymi słowami wyjaśnić omawiane zagadnienia.
- systematycznie odrabia pracę domową.
- w zakresie podstaw programowania samodzielnie rozwiązał na swoim koncie zadania najprostsze oraz średniej trudności (zadania z poziomu 1 i 2).

Na ocenę bardzo dobrą zasługuje uczeń, który:

- rozumie i potrafi własnymi słowami wyjaśnić omawiane zagadnienia.
- systematycznie odrabia pracę domową.
- jest zaangażowany w lekcję i aktywnie w niej uczestniczy.
- w zakresie podstaw programowania samodzielnie rozwiązał na swoim koncie zadania ze wszystkich trzech poziomów trudności.

Na ocenę celującą zasługuje uczeń, który:

- rozumie i potrafi własnymi słowami wyjaśnić omawiane zagadnienia.
- systematycznie odrabia pracę domową.
- w zakresie podstaw programowania samodzielnie rozwiązał na swoim koncie zadania ze wszystkich trzech poziomów trudności oraz zadania z gwiazdką
- wykazał się wysokim poziomem opanowania wiedzy oraz aktywnością na lekcjach
- w szczególności na ocenę celującą zasługują uczniowie, którzy przeszli do wyższych poziomów klas w InstaKod po zrealizowaniu na ocenę bardzo dobrą poziomu swojej klasy oraz uczniowie, którzy otrzymali wyróżnienia w konkursach informatycznych.

KLASA 7

<b>I. Wprowadzenie</b>				
<b>Prawo i zasady bezpieczeństwa</b>				
Na ocenę <b>2</b> uczniów:	Na ocenę <b>3</b> uczniów:	Na ocenę <b>4</b> uczniów:	Na ocenę <b>5</b> uczniów:	Na ocenę <b>6</b> uczniów:
z pomocą nauczyciela wyszukuje informacje w internecie	samodzielnie wyszukuje informacje w internecie	omawia zagadnienia etyczne związane z wykorzystaniem komputerów; samodzielnie wyszukuje informacje w internecie	omawia zagadnienia etyczne związane z wykorzystaniem komputerów; samodzielnie wyszukuje informacje w internecie, krytycznie oceniając źródła	rozszerza zagadnienia etyczne związane z wykorzystaniem komputerów wspomniane w podręczniku; samodzielnie wyszukuje informacje w internecie, krytycznie oceniając źródła
<b>Sieci komputerowe</b>				
Na ocenę <b>2</b> uczniów:	Na ocenę <b>3</b> uczniów:	Na ocenę <b>4</b> uczniów:	Na ocenę <b>5</b> uczniów:	Na ocenę <b>6</b> uczniów:
z pomocą nauczyciela udostępnia swoje pliki w obrębie sieci lokalnej	samodzielnie udostępnia swoje pliki w obrębie sieci lokalnej	omawia różne typy licencji na oprogramowanie; samodzielnie udostępnia swoje pliki w obrębie sieci lokalnej	omawia budowę i funkcjonowanie różnych sieci komputerowych; omawia różne typy licencji na oprogramowanie; samodzielnie udostępnia swoje pliki w obrębie sieci lokalnej	rozszerza wspomniane w podręczniku zagadnienia związane z siecią komputerową; omawia różne typy licencji na oprogramowanie; samodzielnie udostępnia swoje pliki w obrębie sieci lokalnej

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

II. Programowanie				
Wprowadzenie, zmienna liczbowa, wejście, wyjście				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>rozdziela instrukcje wejścia i wyjścia z programu;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji</p>	<p>rozdziela instrukcje wejścia i wyjścia z programu;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji;</p> <p>z pomocą nauczyciela rozwiązuje trudniejsze zadania</p>	<p>zna i potrafi objaśnić działanie instrukcji wejścia i wyjścia;</p> <p>rozumie różnice między wyjściem stałym i zmiennym;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji;</p> <p>z pomocą nauczyciela rozwiązuje trudniejsze zadania</p>	<p>zna i potrafi objaśnić działanie instrukcji wejścia i wyjścia;</p> <p>rozumie różnice między wyjściem stałym i zmiennym;</p> <p> płynnie rozwiązuje zadania proste oraz o średniej trudności;</p> <p>rozumie pojęcie stanu programu;</p> <p>objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna i potrafi objaśnić działanie instrukcji wejścia i wyjścia;</p> <p>rozumie różnice między wyjściem stałym i zmiennym;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>rozumie pojęcie stanu programu;</p> <p>objaśnia krok po kroku działanie swoich programów</p>
Operacje arytmetyczne				
<p>rozdziela operatory dodawania, odejmowania, mnożenia, dzielenia oraz reszty z dzielenia;</p> <p>z pomocą nauczyciela tworzy prosty program, który wykonuje jedną z pięciu operacji (dodawanie, odejmowanie, mnożenie, dzielenie lub reszta z dzielenia)</p>	<p>rozdziela operatory dodawania, odejmowania, mnożenia, dzielenia oraz reszty z dzielenia;</p> <p>samodzielnie tworzy prosty program, który wykonuje jedną z pięciu operacji (dodawanie, odejmowanie, mnożenie, dzielenie lub reszta z dzielenia);</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności</p>	<p>zna i potrafi objaśnić działanie operatorów dodawania, odejmowania, mnożenia, dzielenia oraz reszty z dzielenia;</p> <p>rozumie czym jest zmienna pomocnicza i kiedy należy ją stosować;</p> <p>samodzielnie tworzy prosty program, który wykonuje jedną z pięciu operacji (dodawanie, odejmowanie, mnożenie, dzielenie lub reszta z dzielenia);</p> <p>z pomocą nauczyciela rozwiązuje zadania trudne</p>	<p>zna i potrafi objaśnić działanie operatorów dodawania, odejmowania, mnożenia, dzielenia oraz reszty z dzielenia;</p> <p>rozumie czym jest zmienna pomocnicza i kiedy należy ją stosować;</p> <p> płynnie rozwiązuje zadania proste oraz o średniej trudności;</p> <p>objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna i potrafi objaśnić działanie operatorów dodawania, odejmowania, mnożenia, dzielenia oraz reszty z dzielenia;</p> <p>rozumie czym jest zmienna pomocnicza i kiedy należy ją stosować;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

Instrukcje warunkowe oraz warunki zagnieżdżone				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>rozdzieli sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>z pomocą nauczyciela tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p>	<p>rozdzieli sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>potrafi wskazać, która instrukcja służy do sprawdzania warunków;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające porównania między dwoma zmiennymi;</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p>samodzielnie tworzy prosty program składający się z kilku instrukcji, w tym instrukcji warunkowej, dla porównania między zmienną i stałą;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudne, w tym zadania wymagające łączenia warunków;</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p> płynnie rozwiązuje zadania proste oraz o średniej trudności, w tym zadania wymagające łączenia warunków;</p> <p>objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>zna sześć operatorów porównania;</p> <p>potrafi ocenić prawdziwość porównania między dwoma liczbami;</p> <p>zna i potrafi objaśnić działanie instrukcji warunkowej;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

Pętla pojedyncza i pętla w pętli				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>potrafi wskazać różnice między pętlą warunkową While i pętlą iteracyjną For;</p> <p>z pomocą nauczyciela tworzy prosty program z pętlą warunkową While lub pętlą iteracyjną For;</p> <p>z pomocą nauczyciela implementuje algorytm Euklidesa</p>	<p>potrafi wskazać różnice między pętlą warunkową While i pętlą iteracyjną For;</p> <p>potrafi wskazać, która zmienna służy jako licznik;</p> <p>samodzielnie tworzy prosty program realizujący z pętlą warunkową While lub pętlą iteracyjną For;</p> <p>z pomocą nauczyciela implementuje algorytm Euklidesa;</p> <p>z pomocą nauczyciela implementuje algorytmy znajdowania NWD i NWW;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności</p>	<p>rozumie i potrafi wyjaśnić zasadę działania pętli warunkowej While i pętli iteracyjnej For;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>samodzielnie tworzy prosty i średniozaawansowany program składający, wymagający użycia pętli warunkowej While lub pętli iteracyjnej For;</p> <p>samodzielnie implementuje algorytm Euklidesa;</p> <p>z pomocą nauczyciela implementuje algorytmy znajdowania NWD i NWW;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudniejsze, w tym zadania wymagające użycia pętli w pętli</p>	<p>rozumie i potrafi wyjaśnić zasadę działania pętli warunkowej While i pętli iteracyjnej For;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>wyjaśnia czym jest pętla w pętli i kiedy należy ją stosować;</p> <p>samodzielnie implementuje algorytm Euklidesa;</p> <p>samodzielnie implementuje algorytmy znajdowania NWD i NWW;</p> <p> płynnie rozwiązuje zadania proste oraz o średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)</p>	<p>rozumie i potrafi wyjaśnić zasadę działania pętli warunkowej While i pętli iteracyjnej For;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>wyjaśnia czym jest pętla w pętli i kiedy należy ją stosować;</p> <p>samodzielnie implementuje algorytm Euklidesa;</p> <p>samodzielnie implementuje algorytmy znajdowania NWD i NWW;</p> <p> płynnie rozwiązuje zadania proste oraz o średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p> rozwiązuje zadania trudne (z *)</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

<b>Wprowadzenie do tablic</b>				
Na ocenę <b>2</b> uczeń:	Na ocenę <b>3</b> uczeń:	Na ocenę <b>4</b> uczeń:	Na ocenę <b>5</b> uczeń:	Na ocenę <b>6</b> uczeń:
rozdziela instrukcje inicjalizacji tablicy oraz dodania elementu do tablicy; z pomocą nauczyciela tworzy prosty program z użyciem tablic	rozdziela instrukcje inicjalizacji tablicy oraz dodania elementu do tablicy; samodzielnie tworzy prosty program z użyciem tablic; z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności	wyjaśnia czym są struktury danych i dlaczego są potrzebne zna i potrafi objaśnić działanie instrukcji związanych z tablicami; samodzielnie tworzy prosty i średniozaawansowany program z użyciem tablic; z pomocą nauczyciela rozwiązuje zadania trudniejsze	wyjaśnia czym są struktury danych i dlaczego są potrzebne zna i potrafi objaśnić działanie instrukcji związanych z tablicami; płynnie rozwiązuje zadania proste oraz o średniej trudności; objaśnia krok po kroku działanie swoich programów; potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)	wyjaśnia czym są struktury danych i dlaczego są potrzebne zna i potrafi objaśnić działanie instrukcji związanych z tablicami; płynnie rozwiązuje zadania o dowolnym poziomie trudności; objaśnia krok po kroku działanie swoich programów rozwiązuje zadania trudne (*)



OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

III. Technologie informacyjne				
Edytor graficzny				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela otwiera edytor graficzny; z pomocą nauczyciela korzysta z narzędzi zaznaczania (zaznaczanie odręczne, inteligentne norzyce)	samodzielnie otwiera edytor graficzny; samodzielnie korzysta z narzędzi zaznaczania (zaznaczanie odręczne, inteligentne norzyce) z pomocą nauczyciela edytuje zdjęcia (obrót, skalowanie, odbicie, kontrast, balans kolorów, filtry, warstwy)	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego samodzielnie korzysta z narzędzi zaznaczania (zaznaczanie odręczne, inteligentne norzyce) samodzielnie edytuje zdjęcia (obrót, skalowanie, odbicie, kontrast, balans kolorów, filtry, warstwy) rozwiązuje zadania wspomagające się przykładowym rozwiązaniem	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego samodzielnie korzysta z narzędzi zaznaczania (zaznaczanie odręczne, inteligentne norzyce) samodzielnie edytuje zdjęcia (obrót, skalowanie, odbicie, kontrast, balans kolorów, filtry, warstwy) samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera edytor graficzny; potrafi podać przykłady zastosowania edytora graficznego samodzielnie korzysta z narzędzi zaznaczania (zaznaczanie odręczne, inteligentne norzyce) samodzielnie edytuje zdjęcia (obrót, skalowanie, odbicie, kontrast, balans kolorów, filtry, warstwy) samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; przygotowuje rysunki na konkursy informatyczne
Edytor tekstowy				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela otwiera edytor tekstowy; z pomocą nauczyciela wstawia elementy do dokumentu tekstowego (symbole, tabele, obrazki);	samodzielnie otwiera edytor tekstowy; samodzielnie wstawia elementy do dokumentu tekstowego (symbole, tabele, obrazki);	samodzielnie otwiera edytor tekstowy; samodzielnie wstawia elementy do dokumentu tekstowego (symbole, tabele, obrazki); korzysta z gotowych szablonów; rozwiązuje zadania wspomagające się przykładowym rozwiązaniem	samodzielnie otwiera edytor tekstowy; samodzielnie wstawia elementy do dokumentu tekstowego (symbole, tabele, obrazki); korzysta z gotowych szablonów; formatuje dokument w zadany sposób; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera edytor tekstowy; samodzielnie wstawia elementy do dokumentu tekstowego (symbole, tabele, obrazki); korzysta z gotowych szablonów; formatuje dokument w zadany sposób; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; samodzielnie pisze teksty, prawidłowo stosując poznane dotychczas narzędzia

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

Arkusz kalkulacyjny				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
<p>z pomocą nauczyciela otwiera arkusz kalkulacyjny;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>z pomocą nauczyciela wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>z pomocą nauczyciela formatuje tło komórki</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>rozdziela pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>rozumie, czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania arkusza kalkulacyjnego;</p> <p>samodzielnie wyjaśnia pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>samodzielnie wyjaśnia czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p> <p>rozwiązuje zadania wspomagając się przykładowym rozwiązaniem</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania arkusza kalkulacyjnego;</p> <p>samodzielnie wyjaśnia pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>samodzielnie wyjaśnia czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności</p>	<p>samodzielnie otwiera arkusz kalkulacyjny;</p> <p>potrafi podać przykłady zastosowania arkusza kalkulacyjnego;</p> <p>samodzielnie wyjaśnia pojęcia: wiersz, kolumna, komórka, zakres komórek, adres komórki;</p> <p>samodzielnie zaznacza wskazany zakres komórek;</p> <p>samodzielnie wyjaśnia czym jest zakres komórek;</p> <p>samodzielnie wypełnia danymi tabelę arkusza kalkulacyjnego;</p> <p>samodzielnie formatuje tło komórki;</p> <p>z pomocą nauczyciela dokonuje prostych operacji arytmetycznych na dwóch komórkach</p> <p>samodzielnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p>korzysta z opcji nie wspomnianych w podręczniku</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

<b>Program do tworzenia prezentacji</b>				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela otwiera program do tworzenia prezentacji; z pomocą nauczyciela dodaje dźwięk do prezentacji	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje dźwięk do prezentacji; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje dźwięk do prezentacji; z pomocą nauczyciela dodaje film do prezentacji; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje dźwięk do prezentacji; samodzielnie dodaje film do prezentacji; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje dźwięk do prezentacji; samodzielnie dodaje film do prezentacji; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności korzysta z opcji nie wspomnianych w podręczniku
<b>HTML</b>				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela tworzy prostą stronę mającą nagłówek i jeden akapit tekstu;	samodzielnie tworzy prostą stronę mającą nagłówek i jeden akapit tekstu;	omawia cechy dobrej strony WWW; samodzielnie tworzy prostą stronę mającą nagłówek i jeden akapit tekstu; z pomocą nauczyciela dodaje znaczniki do swojej strony; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem	omawia cechy dobrej strony WWW; wyjaśnia czym jest i do czego służy HTML; samodzielnie tworzy prostą stronę składającą się z kilku nagłówków oraz kilku akapitów tekstu; samodzielnie dodaje znaczniki do swojej strony; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności	omawia cechy dobrej strony WWW; wyjaśnia czym jest i do czego służy HTML; samodzielnie tworzy estetyczną, prostą stronę składającą się z kilku nagłówków oraz kilku akapitów tekstu; samodzielnie dodaje znaczniki do swojej strony; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; korzysta z opcji nie wspomnianych w podręczniku

KLASA 8

I. Wprowadzenie				
Komputer i historia informatyki				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
z pomocą nauczyciela wymienia kilka terminów związanych z informatyką i technologią informatyczną.	samodzielnie wymienia kilka terminów związanych z informatyką i technologią informatyczną.	samodzielnie wymienia kilka terminów związanych z informatyką i technologią informatyczną i objaśnia ich znaczenie; omawia zastosowania informatyki w różnych dziedzinach życia oraz kompetencji informatycznych w różnych zawodach.	samodzielnie wymienia kilka terminów związanych z informatyką i technologią informatyczną i objaśnia ich znaczenie; omawia zastosowania informatyki w różnych dziedzinach życia oraz kompetencji informatycznych w różnych zawodach; przedstawia główne etapy w historycznym rozwoju informatyki i technologii informacyjnej.	wymienia terminy związane z informatyką i technologią informatyczną, nie wspomniane w podręczniku i objaśnia ich znaczenie; omawia zastosowania informatyki w różnych dziedzinach życia oraz kompetencji informatycznych w różnych zawodach; przedstawia główne etapy w historycznym rozwoju informatyki i technologii informacyjnej.

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

II. Programowanie				
Funkcje				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela buduje prosty program z użyciem funkcji.	samodzielnie tworzy prosty program z użyciem funkcji; z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności.	zna i potrafi objaśnić strukturę funkcji w języku Nianiolang; rozumie czym jest funkcja i kiedy należy ją stosować; samodzielnie tworzy prosty oraz średniozaawansowany program z użyciem funkcji; z pomocą nauczyciela rozwiązuje zadania trudne.	zna i potrafi objaśnić strukturę funkcji w języku Nianiolang; rozumie czym jest funkcja i kiedy należy ją stosować; płynnie rozwiązuje zadania proste oraz średniej trudności; objaśnia krok po kroku działanie swoich programów; potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *)	zna i potrafi objaśnić strukturę funkcji w języku Nianiolang; rozumie czym jest funkcja i kiedy należy ją stosować; płynnie rozwiązuje zadania o dowolnym poziomie trudności; objaśnia krok po kroku działanie swoich programów.

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

<b>Pętla pojedyncza i pętla w pętli</b>				
Na ocenę <b>2</b> uczeń:	Na ocenę <b>3</b> uczeń:	Na ocenę <b>4</b> uczeń:	Na ocenę <b>5</b> uczeń:	Na ocenę <b>6</b> uczeń:
<p>potrafi rozróżnić pętlę warunkową While od pętli iteracyjnej For;</p> <p>z pomocą nauczyciela tworzy prosty program z pętlą warunkową While lub pętlą iteracyjną For;</p> <p>z pomocą nauczyciela implementuje algorytm Euklidesa.</p>	<p>potrafi rozróżnić pętlę warunkową While i pętlę iteracyjną For;</p> <p>potrafi wskazać, która zmienna służy jako licznik;</p> <p>samodzielnie tworzy prosty program realizujący z pętlą warunkową While lub pętlą iteracyjną For;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające konwersji liczby między systemem binarnym i dziesiętnym.</p>	<p>rozumie i potrafi wyjaśnić zasadę działania pętli warunkowej While i pętli iteracyjnej For;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>samodzielnie rozwiązuje proste i o średniej trudności zadania, w tym zadania wymagające konwersji liczby między systemem binarnym i dziesiętnym;</p> <p>samodzielnie rozwiązuje zadania średniej trudności; z pomocą nauczyciela rozwiązuje zadania trudniejsze.</p>	<p>rozumie i potrafi wyjaśnić zasadę działania pętli warunkowej While i pętli iteracyjnej For;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>wyjaśnia czym jest pętla w pętli i kiedy należy ją stosować;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *).</p>	<p>rozumie i potrafi wyjaśnić zasadę działania pętli warunkowej While i pętli iteracyjnej For;</p> <p>wyjaśnia różnicę między pętlą skończoną a nieskończoną;</p> <p>rozumie i potrafi wyjaśnić czym jest licznik;</p> <p>wyjaśnia czym jest pętla w pętli i kiedy należy ją stosować;</p> <p> płynnie rozwiązuje zadania proste oraz średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>samodzielnie rozwiązuje zadania trudne (z *).</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

Tablice				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
<p>rozdziela instrukcje inicjalizacji tablicy oraz dodania elementu do tablicy;</p> <p>z pomocą nauczyciela tworzy prosty program z użyciem tablic.</p>	<p>rozdziela instrukcje inicjalizacji tablicy oraz dodania elementu do tablicy;</p> <p>samodzielnie tworzy prosty program z użyciem tablic;</p> <p>z pomocą nauczyciela implementuje algorytmy sortowania przez wybieranie i sortowania przez zliczanie;</p> <p>z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności, w tym zadania wymagające przetwarzania elementów tablicy.</p>	<p>wyjaśnia czym są struktury danych i dlaczego są potrzebne; zna i potrafi objaśnić działanie instrukcji związanych z tablicami;</p> <p>samodzielnie tworzy prosty i średniozaawansowany program z użyciem tablic;</p> <p>samodzielnie implementuje algorytmy sortowania przez wybieranie i sortowania przez zliczanie;</p> <p>z pomocą nauczyciela implementuje algorytm „sito Eratostenesa”;</p> <p>z pomocą nauczyciela rozwiązuje zadania trudniejsze.</p>	<p>wyjaśnia czym są struktury danych i dlaczego są potrzebne; zna i potrafi objaśnić działanie instrukcji związanych z tablicami;</p> <p>samodzielnie implementuje algorytmy sortowania przez wybieranie i sortowania przez zliczanie;</p> <p>samodzielnie implementuje algorytm „sito Eratostenesa”;</p> <p> płynnie rozwiązuje zadania proste oraz o średniej trudności;</p> <p> objaśnia krok po kroku działanie swoich programów;</p> <p>potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *).</p>	<p>wyjaśnia czym są struktury danych i dlaczego są potrzebne; zna i potrafi objaśnić działanie instrukcji związanych z tablicami;</p> <p>samodzielnie implementuje algorytmy sortowania przez wybieranie i sortowania przez zliczanie;</p> <p>samodzielnie implementuje algorytm „sito Eratostenesa”;</p> <p> płynnie rozwiązuje zadania o dowolnym poziomie trudności;</p> <p> objaśnia krok po kroku działanie swoich programów</p>

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

<b>Napisy</b>				
Na ocenę <b>2</b> uczniów:	Na ocenę <b>3</b> uczniów:	Na ocenę <b>4</b> uczniów:	Na ocenę <b>5</b> uczniów:	Na ocenę <b>6</b> uczniów:
znajduje znak w tabeli ASCII dla zadanej wartości; z pomocą nauczyciela tworzy prosty program z użyciem napisów	znajduje znak w tabeli ASCII dla zadanej wartości; samodzielnie tworzy prosty program z użyciem napisów; z pomocą nauczyciela rozwiązuje zadania o średniozaawansowanym stopniu trudności.	wyjaśnia czym jest tablica ASCII; zna i potrafi objaśnić działanie instrukcji związanych z napisami; samodzielnie tworzy prosty i średniozaawansowany program z użyciem napisów; z pomocą nauczyciela rozwiązuje zadania trudniejsze.	wyjaśnia czym jest tablica ASCII; zna i potrafi objaśnić działanie instrukcji związanych z napisami; wyjaśnia pojęcie palindromu; płynnie rozwiązuje zadania proste oraz średniej trudności; objaśnia krok po kroku działanie swoich programów; potrzebuje pomocy nauczyciela jedynie przy rozwiązywaniu najtrudniejszych zadań (z *).	wyjaśnia czym jest tablica ASCII; zna i potrafi objaśnić działanie instrukcji związanych z napisami; wyjaśnia pojęcie palindromu; płynnie rozwiązuje zadania o dowolnym poziomie trudności; objaśnia krok po kroku działanie swoich programów.



OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

III. Technologie informacyjne				
Scratch				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
z pomocą nauczyciela loguje się na platformę Scratch;  z pomocą nauczyciela wybiera tło oraz duszka i tworzy prostą animację.	samodzielnie loguje się na platformę Scratch;  samodzielnie wybiera tło oraz duszka i tworzy prostą animację;  z pomocą nauczyciela dodaje inne ruchome elementy.	samodzielnie loguje się na platformę Scratch;  samodzielnie wybiera tło oraz duszka i tworzy prostą animację;  samodzielnie dodaje inne ruchome elementy.	samodzielnie loguje się na platformę Scratch;  samodzielnie wybiera tło oraz duszka i tworzy prostą animację;  samodzielnie dodaje inne ruchome elementy;  tworzy własne historyjki i realizuje je w języku Scratch	samodzielnie loguje się na platformę Scratch;  samodzielnie wybiera tło oraz duszka i tworzy prostą animację;  samodzielnie dodaje inne ruchome elementy;  tworzy własne historyjki i realizuje je w języku Scratch;  samodzielnie tworzy skomplikowane animacje.
Program do tworzenia filmów				
Na ocenę 2 uczeń:	Na ocenę 3 uczeń:	Na ocenę 4 uczeń:	Na ocenę 5 uczeń:	Na ocenę 6 uczeń:
z pomocą nauczyciela dokonuje prostej obróbki filmu.	samodzielnie dokonuje prostej obróbki filmu;  rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	omawia zastosowania programu do tworzenia filmów;  samodzielnie dokonuje prostej obróbki filmu;  rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	omawia zastosowania programu do tworzenia filmów;  samodzielnie dokonuje prostej obróbki filmu;  samodzielnie rozwiązuje zadania o dowolnym poziomie trudności.	omawia zastosowania programu do tworzenia filmów;  samodzielnie dokonuje prostej obróbki filmu;  samodzielnie rozwiązuje zadania o dowolnym poziomie trudności;  korzysta z opcji nie wspomnianych w podręczniku.

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

<b>Edytor tekstowy</b>				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela otwiera edytor tekstowy; z pomocą nauczyciela dzieli dokument na strony.	samodzielnie otwiera edytor tekstowy; samodzielnie dzieli dokument na strony; z pomocą nauczyciela wstawia numerację, przypisy, stopkę i nagłówek; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	samodzielnie otwiera edytor tekstowy; samodzielnie dzieli dokument na strony; samodzielnie wstawia numerację, przypisy, stopkę i nagłówek; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	samodzielnie otwiera edytor tekstowy; samodzielnie dzieli dokument na strony; samodzielnie wstawia numerację, przypisy, stopkę i nagłówek; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności.	samodzielnie otwiera edytor tekstowy; samodzielnie dzieli dokument na strony samodzielnie wstawia numerację, przypisy, stopkę i nagłówek; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; samodzielnie pisze teksty prawidłowo stosując poznane dotychczas narzędzia.
<b>Arkusz kalkulacyjny</b>				
Na ocenę 2 uczniów:	Na ocenę 3 uczniów:	Na ocenę 4 uczniów:	Na ocenę 5 uczniów:	Na ocenę 6 uczniów:
z pomocą nauczyciela otwiera arkusz kalkulacyjny; z pomocą nauczyciela korzysta z funkcji ILOCZYN.	samodzielnie otwiera arkusz kalkulacyjny; samodzielnie korzysta z funkcji ILOCZYN; z pomocą nauczyciela korzysta z funkcji warunkowych, JEŻELI oraz LICZ.JEŻELI.	samodzielnie otwiera arkusz kalkulacyjny; potrafi podać przykłady zastosowania arkusza kalkulacyjnego w życiu codziennym; samodzielnie korzysta z funkcji ILOCZYN; samodzielnie korzysta z funkcji warunkowych, JEŻELI oraz LICZ.JEŻELI; z pomocą nauczyciela korzysta z funkcji filtracji danych, WYSZUKAJ.PIONOWO oraz ZNAJDŹ.INDEKS rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	samodzielnie otwiera arkusz kalkulacyjny; potrafi podać przykłady zastosowania arkusza kalkulacyjnego w życiu codziennym; samodzielnie korzysta z: <ul style="list-style-type: none"> <li>funkcji ILOCZYN;</li> <li>z funkcji warunkowych, JEŻELI oraz LICZ.JEŻELI;</li> <li>z funkcji filtracji danych, WYSZUKAJ.PIONOWO oraz ZNAJDŹ.INDEKS;</li> </ul> samodzielnie rozwiązuje zadania o dowolnym poziomie trudności.	samodzielnie otwiera arkusz kalkulacyjny; potrafi podać przykłady zastosowania arkusza kalkulacyjnego w życiu codziennym; samodzielnie korzysta z: <ul style="list-style-type: none"> <li>funkcji ILOCZYN;</li> <li>z funkcji warunkowych, JEŻELI oraz LICZ.JEŻELI;</li> <li>z funkcji filtracji danych, WYSZUKAJ.PIONOWO oraz ZNAJDŹ.INDEKS;</li> </ul> samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; korzysta z opcji nie wspomnianych w podręczniku.

OPIS ZAŁOŻONYCH OSIĄGNIĘĆ UCZNIĄ W KLASACH 7-8 I PROPOZYCJE METOD OCENIANIA

<b>Program do tworzenia prezentacji</b>				
Na ocenę <b>2</b> uczeń:	Na ocenę <b>3</b> uczeń:	Na ocenę <b>4</b> uczeń:	Na ocenę <b>5</b> uczeń:	Na ocenę <b>6</b> uczeń:
z pomocą nauczyciela otwiera program do tworzenia prezentacji; z pomocą nauczyciela dodaje animacje do prezentacji	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje animacje do prezentacji; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje animacje do prezentacji; z pomocą nauczyciela dodaje hiperłącza do prezentacji; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje animacje do prezentacji; samodzielnie dodaje hiperłącza do prezentacji; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności.	samodzielnie otwiera program do tworzenia prezentacji; samodzielnie dodaje animacje do prezentacji; samodzielnie dodaje hiperłącza do prezentacji; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; korzysta z opcji nie wspomnianych w podręczniku.
<b>HTML</b>				
Na ocenę <b>2</b> uczeń:	Na ocenę <b>3</b> uczeń:	Na ocenę <b>4</b> uczeń:	Na ocenę <b>5</b> uczeń:	Na ocenę <b>6</b> uczeń:
z pomocą nauczyciela tworzy prostą stronę mającą nagłówek i jeden akapit tekstu; z pomocą nauczyciela zmienia tło swojej strony.	samodzielnie tworzy prostą stronę mającą nagłówek i jeden akapit tekstu; samodzielnie zmienia tło swojej strony; z pomocą nauczyciela wstawia tytuł, opis i autora strony.	samodzielnie tworzy prostą stronę mającą nagłówek i jeden akapit tekstu; samodzielnie zmienia tło swojej strony; samodzielnie wstawia tytuł, opis i autora strony; samodzielnie wstawia tabele i ramki do strony; z pomocą nauczyciela dodaje filmy i muzykę do strony; rozwiązuje zadania wspomagając się przykładowym rozwiązaniem.	samodzielnie tworzy prostą stronę mającą nagłówek i jeden akapit tekstu; samodzielnie zmienia tło swojej strony; samodzielnie wstawia tytuł, opis i autora strony; samodzielnie wstawia tabele i ramki do strony; samodzielnie dodaje filmy i muzykę do strony; samodzielnie dodaje odsyłacze; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności.	samodzielnie tworzy prostą stronę mającą nagłówek i jeden akapit tekstu; samodzielnie zmienia tło swojej strony; samodzielnie wstawia tytuł, opis i autora strony; samodzielnie wstawia tabele i ramki do strony; samodzielnie dodaje filmy i muzykę do strony; samodzielnie dodaje odsyłacze; samodzielnie rozwiązuje zadania o dowolnym poziomie trudności; korzysta z opcji nie wspomnianych w podręczniku

# METODY PRACY NA LEKCJACH

## WPROWADZENIE

### Jacy są uczniowie dziś?

- Żyją w bardzo szybko zmieniającym się multimedialnym świecie; prawie każdy uczeń nosi przy sobie smartphone i z niego korzysta.
- Chętnie operują obrazem i dźwiękiem, w kontraście do czytania tekstu (część ma problemy z czytaniem ze zrozumieniem).
- W konsekwencji częstego sięgania po multimedia, przyzwyczajeni są do podążania za najsilniejszym bodźcem przez to też spora część uczniów ma problemy z koncentracją.
- Mają ogromny potencjał do innowacyjności.
- Nie boją się, że nie będą mieli racji, popełnianie błędów traktują jako drogę do celu.
- Są świetnie zaznajomieni z przeszukiwaniem, chatowaniem, tekstowaniem, graniem.
- Wymieniają się snapami, zdjęciami, plikami.
- Są „na bieżąco” z aplikacjami modnymi w ich środowisku (polecany przez grupę rówieśniczą), szybko uczą się z nich korzystać; potrafią wchodzić w interakcję z interesującą ich technologią informacyjną bez pośrednictwa szkoły.
- Mają mnóstwo doświadczenia w interakcji z nowymi technologiami, ale znacznie mniej z tworzeniem w nowych technologiach i niewiele w rozumieniu jak działają.

### Czego chcemy uczyć?

Podstawa programowa informatyki dla szkoły podstawowej definiuje kilka grup uzupełniających się zagadnień:

1. prawo i bezpieczeństwo korzystania z zasobów cyfrowych,
2. podstawy programowania,
3. technologie informacyjne wspierające naukę i pracę.

### Jak uczyć?

Sugerowaną metodą nauczania jest stopniowe wprowadzanie pojęć oraz budowanie umiejętności technicznych i ich stałe ćwiczenie praktyczne poprzez rozwiązywanie zadań i realizację projektów.

W zakresie technologii informacyjnych, po zapoznaniu się z konkretnymi funkcjami aplikacji, uczniowie mają za zadanie praktycznie zastosować swoje umiejętności do realizacji projektów międzyprzedmiotowych zdefiniowanych w programie *InstaKod* i zadanych na innych przedmiotach szkolnych. Podręcznik sugeruje możliwe tematy do realizacji, niemniej najlepszym podejściem jest współpraca nauczycieli w ramach szkoły i podjęcie wspólnej decyzji o tematach projektów. Efekt końcowy pracy ucznia oceniany powinien być, z technicznego punktu widzenia, na zajęciach informatyki, zgodnie z wytycznymi technicznymi, a z merytorycznego punktu widzenia na przedmiocie, na którym projekt był realizowany.

W zakresie nauczania podstaw programowania, ścieżka praktycznego zastosowania nabytych umiejętności to implementacja rozwiązań zadań programistycznych, w postaci fragmentów kodu, które przechodzą automatyczne testy systemu. Warto podkreślać, że nauka podstaw programowania ma bardzo duże zastosowanie praktyczne, umożliwi w przyszłości rozumienie podstaw działania systemów informatycznych i aplikacji wspierających nasze życie na co dzień, a w przypadku uczniów, którzy pójdą dalej ścieżką nauki informatyki - uczestnictwo w ich tworzeniu.

*InstaKod* jest przestrzenią edukacyjną, będącą przeciwieństwem multimedialnej przestrzeni biernego odbiorcy. Jest uporządkowana, skupia uwagę ucznia na istocie wprowadzanych pojęć. Dzieje się tak z wielu powodów, z których najważniejsze wypunktowano poniżej.

- Środowisko programistyczne InstaKod jest tak proste jak to tylko możliwe, by skutecznie wspierać rozumienie zdefiniowanych w podstawie programowej pojęć informatycznych. Język blokowy Assembly dla klas 4-6 składa się z 10 instrukcji. Dostępny dla klas 7-8 język tekstowy Nianiolang ograniczony jest do 50 instrukcji.
- Obszerna baza zadań o zróżnicowanym stopniu trudności i automatyczne testowanie rozwiązań, umożliwia każdemu uczniowi systematyczną naukę we własnym tempie. Stopień trudności zadań został starannie dobrany do poziomu kształcenia wg klas (4-8). Każde zadanie ma określone wyjście dla zadanego wejścia. Dzięki automatycznym testom uczniowie dostają jasny komunikat o poprawności rozwiązania lub o błędzie. Testy wypisują dodatkowo dla testowanego wejścia wzorcowe wyjście oraz wyjście generowane przez testowany program ucznia. Dzięki temu uczeń może przeanalizować i poprawić swoje błędy.
- Zbiory zadań uzupełnione są materiałami wprowadzającymi w formie podręcznika. Zadania w papierowych kartach pracy wykonują uczniowie bez użycia komputera. Taka forma pracy podczas lekcji pozwala lepiej zrozumieć omawiane zagadnienia.
- Nauczyciel posiada dostęp do informacji o wynikach pracy swoich uczniów.
- Zestawy quizów i sprawdzianów pozwalają na obiektywną ocenę efektów pracy uczniów.

### Konstrukcja lekcji

Na każdym poziomie klasy (4-8) znajduje się 10 rozdziałów tematycznych w skład, których wchodzi:

- I. Poradnik dla nauczyciela
- II. Podręcznik dla ucznia
- III. Przykłady zadań programistycznych on-line z wzorcowymi rozwiązaniami omówionymi w podręczniku
- IV. Karty pracy
- V. Zadania programistyczne on-line w podziale na 3 poziomy trudności
- VI. Quiz on-line
- VII. Sprawdziany on-line

#### I. Poradnik dla nauczyciela

Poradnik nauczyciela zawiera sugestie metodyczne do każdego rozdziału, wzorcowe rozwiązania zadań, quizów, kart pracy i sprawdzianów.

#### II. Podręcznik dla ucznia

Podręcznik zawiera wprowadzenie do tematu i wzorcowe rozwiązania przykładów. Uczniowie zapoznają się z pojęciami omawianymi w danym temacie oraz poznają ich praktyczne zastosowanie. Znaleźć tu można również omówienie instrukcji języka, koniecznych do rozwiązania zadań z danego tematu. Sugeruje się by nauczyciel zadawał uczniom pytania o działanie każdej z instrukcji.

#### III. Przykłady

Przykłady składają się z kilku zadań, na bazie których ćwiczony są wprowadzane pojęcia.

Nauczyciel pokazuje uczniom ścieżkę myślenia. Sugeruje się by nauczyciel, omawiając każdy przykład przeszedł przez cały proces informatycznego podejścia do problemu składający się z:

- A. przeczytania treści zadania z jego zrozumieniem,
- B. ustalenia konkretnego wejścia i oczekiwanego wyjścia i wymyślenia wspólnie z uczniami rozwiązania,
- C. zapisania rozwiązania w formie diagramu na tablicy,
- D. implementacji,
- E. uruchomienia programu dla założonych danych i porównanie wyjścia z oczekiwanym,
- F. automatycznego przetestowania i analizy testów i błędów,
- G. korekty,

H. dalszego testowania i korekty aż do skutku.

### **A. Zrozumienie treści zadania**

Czytanie ze zrozumieniem stanowi dla wielu uczniów spore wyzwanie. Sugeruje się by pokazać uczniom jak rozkładać treść zadania na krótkie fragmenty i interpretować je kawałek po kawałku. Gdy uczeń zgłasza trudność w zrozumieniu polecenia nauczyciel nie powinien nigdy dawać „gotowej” interpretacji zadania. Powinien serią pytań motywować ucznia do samodzielnego przeczytania, zrozumienia i wytłumaczenia własnymi słowami treści zadania.

### **B. Wymyślenie rozwiązania wraz z przykładowymi danymi wejściowymi i oczekiwanym wynikiem**

Budowanie rozwiązania można rozpocząć od wymyślenia przykładowego wejścia i oczekiwanego wyjścia. Uczniowie zgłaszają swoje propozycje rozwiązania, a nauczyciel je akceptuje bądź odrzuca. Czasem gdy uczniowie zgłoszą błędne rozwiązanie nauczyciel może je początkowo przyjąć, by dać uczniom szansę samodzielnego poprawienia błędu. Jeśli uczniowie nie wyłapią błędów od razu, można ich nakierować zadając dodatkowe pytania.

### **C. Zapisanie diagramu rozwiązania na tablicy**

Kolejnym krokiem jest zapisanie wymyślonego rozwiązania w formie diagramu. Diagram rozwiązania może być zapisany w dowolny, zrozumiały dla ucznia sposób, z zachowaniem kolejności działań do wykonania. W diagramie zapisujemy, na których pudełkach wykonujemy operacje, w szczególności, gdzie chcemy zapisać dane wejściowe i gdzie będą znajdować się dane wyjściowe. Uczniowie nie muszą zapisywać w zeszytach rozwiązań zadań przykładowych, gdyż wzorcowe rozwiązania dostępne są w podręczniku.

### **D. Implementacja**

W trakcie pierwszych kilku lekcji zaleca się by przed przystąpieniem do implementacji rozwiązania na komputerze, przypomnieć uczniom poznane instrukcje. Ich definicje uczniowie powinni znać na pamięć i być w stanie własnymi słowami wytłumaczyć co dzieje się, gdy uruchomiona zostanie konkretna instrukcja.

### **E. Uruchomienie programu dla założonych danych i porównanie wyjścia z oczekiwanym**

Przed uruchomieniem programu przypominamy przykładowe dane wejściowe i oczekiwany wynik. Program uruchamiamy instrukcją po instrukcji, przy każdym kroku pytając uczniów o to, czego należy się spodziewać po wykonaniu danej instrukcji i skonfrontować to z pojawiającymi się wartościami w pudełkach oraz z zapisami na konsoli. Po zakończeniu programu porównujemy otrzymany wynik z oczekiwanym. Jeżeli się różnią, analizujemy różnice i pytaniami nakierowujemy uczniów na poprawną odpowiedź.

### **F. Automatyczne testy i analiza błędów**

Opcja automatycznego testowania i zaliczenia zadania dostępna jest po ręcznym uruchomieniu zadania. Każda wprowadzona zmiana do kodu wymaga ponownego uruchomienia przed automatycznym testowaniem. Testy zwrócą jeden z trzech możliwych wyników:

- „Dobrze”, co oznacza, że nasze rozwiązanie jest poprawne;
- „Brak danych”, co oznacza, że napisany program próbuje pobrać więcej lub mniej danych wejściowych niż jest podane w zadaniu;
- „Zła odpowiedź”. Przy tym komunikacie należy porównać zapisy w kolumnach „Twoja odpowiedź” i „Prawidłowa odpowiedź”. Wyjście generowane przez program musi, co do znaku, zgadzać się z wyjściem wzorcowym. Warto w szczególności zwrócić uwagę na duże i małe litery, spacje oraz przejścia między wierszami.

### **G. Wykorzystanie błędów do ćwiczenia ich analizy**

Sugeruje się wykorzystywać błędne sugestie uczniów w rozwiązaniach przykładów, by pokazać uczniom ścieżkę analizy błędów i ich poprawiania. Dodatkowo analiza błędów ćwiczona jest poprzez rozwiązywanie kart pracy.

Przechodząc przez całą ścieżkę rozwiązania problemu informatycznego, zachęca się by angażować uczniów w wymyślanie i implementowanie rozwiązań oraz krytyczną analizę sugestii kolegów.

#### IV. Karty pracy

Karty pracy budują głębsze zrozumienie omawianych zagadnienie. Nie mając do dyspozycji komputera, uczniowie nie mają możliwości rozwiązywania zadań metodą prób i błędów. Muszą samodzielnie przeanalizować dany fragment kodu. Dostępne są następujące typy kart pracy:

1. Wartości w pudełkach i zapis na konsoli
2. Wartości początkowe i końcowe
3. Analiza poprawności rozwiązania
4. Diagram rozwiązania

#### V. Zadania

##### A. Zeszyt do informatyki

Rozwiązując zadania, uczniowie powinni powielać pokazaną przez nauczyciela ścieżkę rozwiązania zadania przykładowego.

Uczniowie najchętniej siadają do komputera i implementują rozwiązania, które często nie są całkowicie przemyślane. W przypadku bardziej złożonych zadań, zdarza im się zapomnieć co mieli zrobić, gdzie zapisali wczytane dane, które pudełka należy zmodyfikować i gdzie zostały wykonane obliczenia. Swoją kod testują metodą prób i błędów.

Zachęca się by uczyć od początku intencji działań, wymyślenia rozwiązania i zapisania go na kartce w formie prostego diagramu, z ustaleniem gdzie (w którym pudełku) i jakie działania mają być podjęte. Istotna jest rola nauczyciela, który, pokazując wzorcową ścieżkę rozwiązania przykładów, za każdym razem zapisuje diagram rozwiązania na tablicy.

Zaleca się by uczniowie mieli zeszyty do informatyki i zapisywali w nich diagramy wszystkich rozwiązań, a dopiero w kolejnym kroku je implementowali, testowali i poprawiali.

##### B. Poziomy trudności zadań, termin rozwiązania zadań i oceny

Zadania podzielone są na 3 poziomy trudności.

Poziom 1 składa się z trzech zadań najprostszyc. Oczekuje się zaliczenia wszystkich zadań tego działu by uznać, że uczeń zbudował oczekiwane umiejętności na poziomie dostatecznym. Sugerowane przez system oceny przypisują za 2 zaliczone zadania ocenę dopuszczającą (2), za 3 zaliczone zadania ocenę dostateczną (3). Zadania zostały tak dobrane by nawet najsłabsi uczniowie mieli szansę je rozwiązać samodzielnie.

Poziom 2 składa się z trzech zadań średniej trudności. Za zaliczenie łącznie (wraz z poziomem 1) od 4 do 6 zadań system sugeruje ocenę dobrą (4). Zakłada się, że około 80% klasy zaliczy minimum 4 zadania.

Poziom 3 składa się z trzech zadań trudnych. Za zaliczenie łącznie (wraz z poziomem 1 i 2) od 7 do 9 zadań system sugeruje ocenę bardzo dobrą (5). W części rozdziałów w poziomie 3 znajdują się dodatkowe zadania z „\*”. Za zaliczenie 10 lub więcej zadań system sugeruje ocenę 5+.

Należy zachęcać uczniów do częstego zaglądania do swoich ocen dostępnych na koncie ucznia. Pozwala to uczniowi szczerze ocenić swoją systematyczną pracę, czerpać satysfakcję z wyników i planować dalszą pracę.

Nauczyciel ustala termin oddania pracy domowej. Jeśli uczeń poprawnie rozwiąże zadanie po upływie tego terminu zostanie ono zaakceptowane przez system, natomiast proponowana ocena nie zostanie zaktualizowana. Oznacza to, że jeśli uczeń rozwiązał 4 zadania do końca terminu podanego przez nauczyciela

i kolejne 5 zadań po terminie, to w ocenach pojawi się 9 rozwiązanych zadań z 9 (9/9) oraz sugerowana ocena 4. Oceny sugerowane przez system są tylko wsparciem dla nauczyciela. Nauczyciel może skorzystać z sugerowanych przez system ocen lub ustalić zupełnie inną, własną skalę oceniania.

### VI. Kartkówki (quizy)

Do każdego rozdziału przygotowane zostały quizy. Quiz składa się z jednego zadania zbliżonego trudnością do zadań z poziomu 1. Rozwiązanie zadania quizowego nie powinno zająć więcej niż 5 minut. Jego celem jest zmotywowanie uczniów do samodzielnej, systematycznej pracy. Jeżeli uczeń rozwiązał samodzielnie zadania z pierwszego poziomu, powinien bez problemu zaliczyć quiz. W każdym quizie dostępne jest zadanie w trzech różnych wersjach (A, B i C). Dzięki temu minimalizowane jest prawdopodobieństwo ściągania.

### VII. Sprawdziany

Na każdym poziomie klasy przewidziane są 2 sprawdziany. W skład sprawdzianu wchodzi 4 zadania oraz kilka dodatkowych zadań z „\*”. W raporcie z wynikami sprawdzianów nauczyciel otrzymuje informację o liczbie zaliczonych zadań. Skalę oceniania nauczyciel ustala sam, system nie sugeruje ocen za sprawdziany. Najprostsza skala oceniania to

- 1 zadanie – ocena 2
- 2 zadania – ocena 3
- 3 zadania – ocena 4
- 4 zadania - ocena 5
- Rozwiązanie 5 lub więcej zadań – ocena 5+

Sprawdziany dostępne są w 3 wersjach by zminimalizować problem ściągania rozwiązań od innych uczniów. Zadania z „\*” dodane zostały by najlepsi uczniowie nie nudzili się po rozwiązaniu zadań 1-4 i nie przeszkadzali pozostałym uczniom.

### Ważne wskazówki do organizacji pracowni informatycznej

#### Atmosfera cichej, skupionej pracy w pracowni

Proces precyzyjnego, uporządkowanego myślenia stanowi podstawę nauki programowania. Atmosfera cichej, skupionej pracy w pracowni komputerowej i szacunku dla potrzeby skupienia się na swoich myślach kolegów i koleżanek, wspiera ten proces. Ustalenie na samym początku zajęć zasad funkcjonowania w pracowni i egzekwowanie tych zasad jest konieczne.

Skutecznymi technikami wspierającymi tworzenie atmosfery cichej skupionej pracy są:

- Wyciszenie uczniów. Przed wejściem do pracowni uczniowie powinni zakończyć wszelkie dyskusje i zostawić je za drzwiami. Przyzwyczajenie uczniów do wchodzenia do pracowni w ciszy bardzo pomaga w organizacji czasu pracy w pracowni poza lekcjami, np. w czasie przerwy lub po lekcjach, kiedy nauczyciel udostępnia możliwość skorzystania z pracowni do pracy indywidualnej.
- Rozdzielenie lekcji na dwie niezależne części, wspólne omówienie tematu oraz indywidualna praca nad zadaniami.
- Ustalenie sposobu sygnalizowania potrzeby pomocy w czasie skupionej pracy bez odzywania się, np. przez podniesienie ręki. Sugeruje się by pomagać uczniom indywidualnie, prowadząc z nimi rozmowę ściszym głosem.

#### Nakierowujące pytania

Bardzo skuteczną metodą usamodzielniania uczniów w rozwiązywaniu problemów informatycznych jest metoda przybliżających pytań. Polega ona na zadawaniu uczniom tak sformułowanych pytań by prowadziły ich do poprawnego rozwiązania.



### Czytanie ze zrozumieniem

Najczęściej słyszaną prośbą o pomoc ze strony uczniów jest prośba o wytłumaczenie treści polecenia (zadania). Skuteczną metodą doskonalenia czytania ze zrozumieniem jest poproszenie ucznia by przeczytał jeszcze raz polecenie. Jeśli nie przyniesie to skutku, należy poprosić go by przeczytał polecenie na głos. Jeśli nie przyniesie to skutku, należy poprosić go by przeczytał na głos fragment polecenia i powiedział własnymi słowami co przeczytał, równocześnie zadając pytania zmuszające go do zastanowienia się nad treścią.

### Zaangażowanie najlepszych uczniów do pomocy

Po pierwszych kilku zajęciach, gdy zasady pracy w pracowni są już uczniom dobrze znane, warto zaangażować uczniów najzdolniejszych do pomagania uczniom najslabszym. Należy podkreślić, że ich pomoc ma polegać na tłumaczeniu, a nie rozwiązywaniu zadań za kolegów.

### Ścieżka dla uczniów zdolnych

Lekcje informatyki w szkole są dla wszystkich uczniów, zarówno dla tych najmocniejszych, jak i tych najslabszych. InstaKod pozwala na prowadzenie uczniów zdolnych szybszym tempem. Uczniowie mogą samodzielnie, za zgodą nauczyciela przejść do następnych lekcji. Jeśli napotkają na zadania lub pojęcia, których nie są w stanie samodzielnie zrozumieć, mogą skorzystać z czasu zajęć dodatkowych. Nauczyciel nie powinien rezerwować czasu indywidualnej pracy z takimi uczniami w trakcie lekcji z całą klasą. Po zaliczeniu wszystkich rozdziałów z danego poziomu, nauczyciel powinien dopisać ucznia do kolejnego poziomu (dając mu nowy login i hasło).

Uczniowie zaczynający od poziomu **Klasa 4 →Klasa 5 kontynuacja→Klasa 6 kontynuacja→Klasa 7-8**

Uczniowie zaczynający od poziomu **Klasa 5 od podstaw (+rozszerzenia) →Klasa 6 kontynuacja→Klasa 7-8**

Uczniowie zaczynający od poziomu **Klasa 6 od podstaw (+rozszerzenia) → Klasa 7-8**

## **DODATEK A.**

### **LISTA ALGORYTMÓW REALIZOWANYCH W KLASACH 4-6**

1. operacje arytmetyczne: dodawanie i odejmowanie ( $2a - b + 3$ ) \*
2. znajdowanie maksimum z dwóch liczb
3. znajdowanie minimum z dwóch liczb
4. wielokrotność z zapisu binarnego
5. mnożenie przy pomocy pętli
6. kwadrat liczby przy pomocy pętli
7. dzielenie przy pomocy pętli
8. reszta z dzielenia przy pomocy pętli
9. badanie parzystości przy pomocy pętli \*
10. średnia z trzech liczb przy pomocy pętli
11. znajdowanie maksimum z trzech liczb
12. znajdowanie minimum z trzech liczb
13. sortowanie bąbelkowe trzech liczb
14. znajdowanie liczby środkowej z trzech liczb \*

\* Realizowany w zadaniach z poziomu 3.

## **DODATEK B.**

### **LISTA ALGORYTMÓW REALIZOWANYCH W KLASACH 7-8**

1. znajdowanie maksimum z dwóch liczb
2. znajdowanie minimum z dwóch liczb
3. badanie parzystości liczby
4. znajdowanie maksimum z trzech liczb
5. znajdowanie minimum z trzech liczb
6. znajdowanie liczby środkowej
7. sortowanie bąbelkowe trzech liczb
8. znajdowanie maksimum ze strumienia
9. znajdowanie minimum ze strumienia
10. wyodrębnianie cyfr danej liczby
11. naiwny test pierwszości
12. znajdowanie NWD za pomocą algorytmu Euklidesa w wersji z odejmowaniem
13. znajdowanie NWD za pomocą algorytmu Euklidesa w wersji z resztą z dzielenia
14. znajdowanie NWW za pomocą algorytmu Euklidesa
15. konwersja liczby dziesiętnej na liczbę binarną
16. konwersja liczby binarnej na liczbę dziesiętną
17. wyszukiwanie liniowe elementów tablicy
18. wyszukiwanie największego elementu tablicy
19. wyszukiwanie najmniejszego elementu tablicy
20. wyszukiwanie lidera w tablicy \*
21. sortowanie przez wybieranie tablicy
22. sortowanie przez zliczanie tablicy
23. sortowanie bąbelkowe tablicy \*
24. wyszukiwanie binarne
25. generacja liczb pierwszych sitem Eratostenesa
26. szyfrowanie napisu przy pomocy szyfru Cezara
27. sprawdzanie czy napis jest palindromem
28. sprawdzanie czy napisy są swoimi anagramami
29. wyszukiwanie słów podwójnych \*
30. wyszukiwanie wzorca w tekście \*

\* Realizowany w zadaniach z poziomu 3.

## DODATEK C.

### LISTA INSTRUKCJI W JĘZYKU ASSEMBLY

#### Zmienna

Pudełko, czyli miejsce w pamięci komputera, w którym przechowujemy liczbę. Pudełko ma swoją nazwę i rozmiar (liczbę bitów). W Assembly pudełka nazywają się A, B, C i D. Każde ma 16 bitów oraz przypisaną wartość początkową 0.

#### Wejście do programu

Wejście to dane podane do programu.

**Wczytaj do B** jest **wejściem** danych do programu. Przy każdym uruchomieniu programu instrukcja prosi użytkownika o podanie liczby. Wczytana liczba zostanie zapisana w pudełku B. Liczba, która wcześniej znajdowała się w B, zostanie usunięta.

#### Wyjście z programu, stałe i zmienne

Wyjście to dane wypisane przez program w celu komunikacji z użytkownikiem.

W Assembly mamy trzy instrukcje wyjścia. Są to:

**Wypisz pudełko C** wypisuje na konsoli liczbę znajdującą się w pudełku C. Wypisana wartość jest **zmienna** i zależy od tego, jaka liczba aktualnie znajduje się w pudełku C.

**Wypisz napis 2345** wypisuje na konsoli tekst z ramki. Służy programiście do wyświetlania komunikatów dla użytkownika programu. Wypisywany komunikat jest **stały** i nie zależy od wczytanych danych.

**Przejdź do nowej linii** sprawia, że kolejny zapis na konsoli pojawi się w nowym wierszu.

#### Operacje arytmetyczne

**Zwiększ D o 5** oblicza sumę liczby znajdującej się w pudełku D oraz liczby 5 i zapisuje wynik w pudełku D.

**Zmniejsz A o B** oblicza różnicę liczby znajdującą się w pudełku A oraz liczby znajdującą się w pudełku B i zapisuje wynik w pudełku A.

#### Przypisanie wartości do pudełka

Instrukcja **Ustaw** przypisuje (wstawia) do wybranego pudełka konkretną liczbę lub wartość innego pudełka.

**Ustaw A na 5** przypisze do pudełka A liczbę 5. W pudełku A będzie znajdować się liczba 5.

**Ustaw A na B** przypisze do pudełka A liczbę, która znajduje się w pudełku B. W pudełku A i B będzie znajdować się ta sama liczba.

## Instrukcja warunkowa

Instrukcja warunkowa umożliwia warunkowe wykonanie części programu.

**Jeżeli** `A > 9` **skocz do** następnej **inaczej skocz do** końca

Jeżeli warunek `A > 9` **jest** spełniony, to wykonana zostanie instrukcja z następnej linii kodu.

Jeżeli warunek **nie jest** spełniony, (to znaczy że `A < 9` lub `A = 9`) program zostanie zakończony.

## Instrukcja skocz do

**Skocz do** `5` przechodzi do podanego numeru linii kodu, w tym przypadku do linii 5. Pozwala przeskoczyć blok instrukcji lub wrócić do bloku instrukcji już wykonanych.

**Skocz do** końca kończy program.

## Pętla

Pętla służy do powtarzania bloku instrukcji bez konieczności ponownego pisania ich w kodzie.

Poniższy program wypisze na konsoli linię poziomą składającą się z 5 gwiazdek:

**Ustaw** `C` **na** `0` licznik to specjalne pudełko liczące ile razy została wykonana pętla. Na początku licznik ustawiamy na 0.

**Jeżeli** `C = 5` **skocz do** końca **inaczej skocz do** następnej ponieważ chcemy, by pętla została wykonana 5 razy, zakończymy pętlę gdy licznik będzie równy 5.

**Wypisz napis** `*` wypisuje na konsoli `*`.

**Zwiększ** `C` **o** `1` po wykonaniu pętli zwiększamy licznik 1. W ten sposób w liczniku znajduje się aktualna liczba przebiegów pętli.

**Skocz do** `2` wracamy do wiersza numer 2, w którym znajduje się warunek kończący pętlę. Program ponownie sprawdzi czy `C = 5`. Jeśli tak, to znaczy, że wykonaliśmy pętlę 5 razy i możemy ją zakończyć. W przeciwnym razie wykonane zostaną te same instrukcje.

## DODATEK D.

### LISTA INSTRUKCJI W JĘZYKU NIANIOLANG

1. Struktura programu	
<code>use console;</code>	Deklaracja korzystania z biblioteki <b>conosole</b> . W tej bibliotece znajdują się funkcje służące do interakcji z użytkownikiem na konsoli.
<code>def test::main() {     ... }</code>	Funkcja <b>main</b> musi znaleźć się w każdym programie. Po uruchomieniu programu ta funkcja wywoływana jest jako pierwsza. Między klamrami w miejscu wielokropka umieszczane będą instrukcje programu.
2. Konsola	
<code>console::print();</code>	Wypisuje na konsoli komunikat umieszczony między nawiasami ().
<code>console::print(5);</code>	Liczby wpisujemy bez cudzysłowów.
<code>console::print('Witaj');</code>	Napisy umieszczamy między znakami cudzysłowu.
<code>console::println();</code>	Działa tak jak Enter. Przechodzi do następnego wiersza na konsoli.
<code>console::read_int();</code>	Wczytuje liczbę podaną przez użytkownika.
<code>console::read_string();</code>	Wczytuje napis podany przez użytkownika.
3. Zmienne	
<code>var a;</code>	Deklaracja zmiennej o nazwie „a”. Inicjalizacja traktowana jest jak instrukcja, dlatego należy zakończyć ją średnikiem.
<code>a = 10;</code>	Przypisanie zmiennej o nazwie „a” wartości 10. Od teraz zmienna „a” będzie przechowywać wartość 10.
<code>a = console::read_int();</code>	Przypisanie zmiennej wartości, która zostanie pobrana od użytkownika w trakcie wykonywania programu. Wartość ta nie jest znana przed uruchomieniem programu i przy każdym wykonaniu może być inna.
<code>var b = 5;</code>	Możemy przypisać zmiennej wartość w jednym kroku razem z deklaracją. Operacja nadania zmiennej wartości początkowej nazywana jest inicjalizacją. Instrukcja ta deklaruje zmienną o nazwie „b” i przypisuje jej wartość 5.

## LISTA INSTRUKCJI W JĘZYKU NIANIOLANG KLASY 7-8

<b>var b = 8;</b> <b>var a = b;</b>	Deklaracja zmiennej „b” i inicjalizacja wartością 8. Deklaracja zmiennej „a” i inicjalizacja wartością przechowywaną przez zmienną „b”. W tym przypadku jest to liczba 8.
<b>var a = 10;</b> <b>var b = console::read_int();</b>  <b>a = b;</b>  <b>b = 5;</b>	Deklaracja zmiennej „a” i inicjalizacja wartością 10. Deklaracja zmiennej b i inicjalizacja wartością, która zostanie pobrana od użytkownika w trakcie wykonywania programu. Nadpisanie liczby przechowywanej przez „a” przez liczbę znajdującą się w „b”. Od teraz w zmiennej „a” przechowywana jest liczba znajdującą się zmiennej „b”. Nadpisanie liczby przechowywanej przez „b” przez liczbę 5. Od teraz w zmiennej „b” przechowywana jest liczba 5.
<b>console::print(a);</b>	Wypisanie na konsolę liczby znajdującą się w zmiennej „a”.

**4. Operacje arytmetyczne**

<b>a = 3 + 2;</b>	Dodawanie ("+") W zmiennej „a” zostanie zapisana suma 3 + 2.
<b>b = 4 - 1;</b>	Odejmowanie ("-") W zmiennej „b” zostanie zapisana różnica 4 - 1.
<b>c = 5 * 2;</b>	Mnożenie ("*") W zmiennej „c” zostanie zapisany iloczyn 5 * 2.
<b>d = 15 / 3;</b>	Dzielenie ("/") W zmiennej „d” zostanie zapisany iloraz 15 / 3.
<b>e = 8 % 4;</b>	Reszta z dzielenia ("%") W zmiennej „e” zostanie zapisana reszta z dzielenia 8 przez 4.
<b>a = a + 2;</b> <b>a += 2;</b>	Dodawanie liczby znajdującą się w zmiennej „a” do 2. Wynik operacji zostanie zapisany w zmiennej „a”.
<b>b = b - 2;</b> <b>b -= 2;</b>	Odejmowanie liczby znajdującą się w zmiennej „b” od 2. Wynik operacji zostanie zapisany w zmiennej „b”.
<b>c = c * 2;</b> <b>c *= 2;</b>	Mnożenie liczby znajdującą się w zmiennej „c” przez 2. Wynik operacji zostanie zapisany w zmiennej „c”.
<b>d = d / 2;</b> <b>d /= 2;</b>	Dzielenie liczby znajdującą się w zmiennej „d” przez 2. Wynik operacji zostanie zapisany w zmiennej „d”.
<b>e = e % 2;</b> <b>e %= 2;</b>	Reszta z dzielenia liczby w zmiennej „e” przez 2. Wynik operacji zostanie zapisany w zmiennej „e”.
<b>a = a + b;</b>	Dodawanie liczby w zmiennej „a” do liczby w zmiennej „b”. Wynik operacji zostanie zapisany w zmiennej „a”.
<b>a = (a - b)/3;</b>	Odejmowanie liczby w zmiennej „a” od liczby w zmiennej „b”. Dzielenie różnicy przez 3. Wynik operacji zostanie zapisany w zmiennej „a”.
<b>x++;</b>	Zwiększenie liczby w zmiennej „x” o 1.
<b>x--;</b>	Zmniejszenie liczby w zmiennej „x” o 1.

**5. Instrukcja warunkowa, operatory relacyjne i operatory logiczne**

Instrukcja warunkowa umożliwia wykonanie bloku instrukcji w zależności od prawdziwości danego warunku. Jeśli warunek jest spełniony, dany blok instrukcji zostaje wykonany. W przeciwnym razie zostaje całkowicie pominięty.

<pre>if (a &gt; 2) { ... }</pre>	<p>Jeśli liczba w zmiennej „a” jest większa od 2, zostaną wykonane instrukcje umieszczone między klamrami <b>if</b>. Jeśli liczba w zmiennej „a” nie jest większa od 2, cały blok instrukcji znajdujących się między klamrami <b>if</b> zostanie pominięty.</p>
<pre>if (a % 2 == 0) { ... } else { ... }</pre>	<p>Jeśli reszta z dzielenia liczby w zmiennej „a” przez 2 jest równa 0 (tzn. jeśli w „a” znajduje się liczba parzysta), zostaną wykonane instrukcje umieszczone między klamrami <b>if</b>. Jeśli reszta z dzielenia liczby w zmiennej „a” przez 2 nie jest równa 0 (tzn. jeśli w „a” nie znajduje się liczba parzysta), zostanie wykonany blok instrukcji znajdujący się między klamrami <b>else</b>.</p>

**Operatory relacyjne w języku Nianiolang**

Operator	Przykład użycia	Znaczenie
<	if(a < b)	mniejszy
<=	if(a <= b)	mniejszy bądź równy
>	if(a > b)	większy
>=	if(a >= b)	większy bądź równy
==	if(a == b)	równy
!=	if(a != b)	różny

**Operatory logiczne. Łączenie warunków**

Operator	Przykład użycia	Znaczenie
!	if(!(a == b))	negacja
&&	if(a > b && a < c)	koniunkcja
	if(a == b    a == c)	alternatywa



6. Pętle	
<p><b>Pętla rep ma następującą postać:</b></p> <pre> rep var i (ilość_powtórzeń) {     ... } </pre>	<p>Pętla służy do powtarzania bloku instrukcji bez konieczności ponownego pisania ich w kodzie. Pętla <b>rep</b> jest najprostszą pętlą dostępną w języku Nianiolang. Pętla <b>rep</b> ma następującą postać:</p> <p>Program wykona instrukcje umieszczone w miejscu wielokropka tyle razy, ile wynosi liczba wpisana w nawiasie. Możemy zarówno wpisać liczbę na „sztywno” np. 5, lub przekazać zmienną.</p>
<pre> rep var i (5) {     ... } </pre>	<p>Program wykona 5 razy instrukcje umieszczone w miejscu wielokropka.</p>
<p><b>Pętla while ma następującą postać:</b></p> <pre> while (warunek) {     ... } </pre>	<p>Program ocenia wartość logiczną <b>warunku</b>, wyrażenia umieszczonego między nawiasami pętli while, tzn. sprawdza czy warunek jest spełniony czy też nie. Jeśli nie, to program pominie instrukcje umieszczone w miejscu wielokropka. Jeśli tak, to wykona te instrukcje i wróci do sprawdzenia warunku.</p>
<pre> while ( x &lt; 11 ){     ... } </pre>	<p>Program będzie wykonywał instrukcje umieszczone w miejscu wielokropka dopóki warunek pętli <b>while</b> jest prawdziwy. Instrukcje będą powtarzane dopóki x jest mniejsze od 11.</p>

<p><b>Pętla for ma następującą postać:</b></p> <pre> <b>for ( I; II; III ) {</b>     ... <b>}</b> </pre>	<p>Pętla <b>for</b> działa w podobny sposób jak pętla <b>while</b>. Jest stosowana jedynie w celu zwiększenia czytelności kodu, gdyż umożliwia wykonanie trzech operacji w jednym wierszu. Operacje te to ustawienie licznika, sprawdzenie warunku kończącego pętlę i zmiana wartości licznika. Pętla <b>for</b> ma następującą postać:</p> <p>W miejscu <b>I</b> umieszczamy instrukcję rozpoczynającą. Ta instrukcja wykonana zostanie tylko raz. Najczęściej jest to deklaracja i inicjalizacja licznika.</p> <p>W miejscu <b>II</b> umieszczamy warunek wykonywania pętli. Przed każdym wykonaniem pętli program sprawdza czy relacja jest prawdziwa i wykona pętlę tylko wtedy, jeśli warunek jest spełniony.</p> <p>W miejscu <b>III</b> umieszczamy instrukcję, która wykona się na końcu każdego wykonania pętli. Najczęściej będzie to zmiana wartości licznika.</p>
<pre> <b>for (var i = 0; i &lt; 10; ++i) {</b>     ... <b>}</b> </pre>	<p>Program będzie wykonywał instrukcje umieszczone w miejscu wielokropka, dopóki liczba w zmiennej „i” jest mniejsza od 10. Przy każdym wykonaniu pętli liczba w zmiennej „i” jest zwiększana o 1. Pętla wykona się dokładnie 10 razy.</p>
<p><b>Continue i break</b></p>	<p>Instrukcje <b>continue</b> i <b>break</b> dają nam więcej możliwości w korzystaniu z pętli.</p> <p>Instrukcja <b>continue</b> przerywa wykonywanie bieżącego przebiegu pętli (nie wykonując dalszych instrukcji tego przebiegu przez pętlę) i przechodzi do następnego wykonania pętli.</p> <p>Instrukcja <b>break</b> kończy pętlę.</p>

<b>7. Tablice</b>	
Aby korzystać ze specjalnych instrukcji dla tablic na górze programu umieszczamy: <b>use array;</b>	
<b>var a = [];</b>	Deklaracja tablicy o nazwie „a”. Tablica jest pusta, tzn. nie zawiera żadnych elementów.
<b>var b = array::init(5, 1);</b>	Deklarujemy tablicę o nazwie „b” składającą się z 5 elementów. Wszystkie elementy tablicy zostały zainicjalizowane wartością 1.
<b>array::push(ref a, 3);</b>	Powiększenie tablicy a o jeden element. Do tablicy a dostawiany jest nowy element o wartości 3.
<b>array::push(ref a, console::readln());</b>	Powiększenie tablicy a o jeden element. Do tablicy a dostawiany jest nowy element o wartości równej liczbie pobranej od użytkownika.
<b>array::pop(ref a);</b>	Pomniejszenie tablicy a o jeden element. Ostatni element tablicy a zostanie usunięty.
<b>n = array::len(a);</b>	Do zmiennej n zostanie przypisana długość tablicy a.
<b>console::print( array::len(b));</b>	Na konsoli zostanie wypisana długość tablicy b.
<b>a[3] = 10;</b>	Do elementu o indeksie 3 znajdującym się w tablicy a zostanie przypisana liczba 10.
<b>for var element (b) {     console::print(element); }</b>	Na konsoli zostaną wypisane wszystkie elementy tablicy b.

<b>8. Napisy</b>	
Aby korzystać ze specjalnych instrukcji dla napisów na górze programu umieszczamy: <b>use string;</b>	
<b>var a = 'slowo';</b>	Deklaracja i inicjalizacja zmiennej o nazwie „a” przechowującej napis „slowo”.
<b>var a = 'liczba'; var b = 'tablica'; var c = a . b;</b>	Deklaracja i inicjalizacja zmiennej o nazwie „a” przechowującej napis „liczba”. Deklaracja i inicjalizacja zmiennej o nazwie „b” przechowującej napis „tablica”. Deklaracja i inicjalizacja zmiennej o nazwie „c” przechowującej połączenie zmiennych a i b. Zmienna c będzie przechowywać napis „liczbatablica”.
<b>n = string::length(a);</b>	Do zmiennej n zostanie przypisana długość napisu a.
<b>console::print(string::lenth(b));</b>	Na konsoli zostanie wypisana długość napisu b.
<b>c = string::get_char(a, 3);</b>	Do zmiennej n zostanie przypisany znak o indeksie 3 z napisu a.
<b>console::print(string::get_char(b, 5));</b>	Na konsoli zostanie wypisany znak o indeksie 5 z napisu b.
<b>console::print(array::len(b));</b>	Na konsoli zostanie wypisana długość tablicy b.
<b>x = string::ord('a');</b>	Do zmiennej x zostanie przypisana liczba odpowiadająca numerowi w tablicy ASCII dla litery a.
<b>x = string::chr(68);</b>	Do zmiennej x zostanie przypisany znak odpowiadający numerowi w tablicy ASCII.
<b>console::print(string::substr(a, 3, 5));</b>	Na konsoli zostanie wypisany napis będący 5-znakowym fragmentem napisu a, poczynając od znaku od indeksie 3.
<b>if ('halo' eq 'aloha'){   ... }</b>	Sprawdzenie czy napis 'halo' jest taki sam jak napis 'aloha'. Napisy są różne, więc warunek nie jest spełniony.

**9. Funkcje**

Do tej pory korzystaliśmy z funkcji dostępnych w różnych bibliotekach, takich jak console, array i string. Okazuje się, że możemy sami tworzyć podobne funkcje.

<pre><b>def herbatniki () {     console::print('chce herbatniki'); }</b></pre>	Funkcja o nazwie „herbatniki” wypisująca na konsoli napis „chce herbatniki”.
<pre><b>def podwojona (a) {     console::print(a*2); }</b></pre>	Funkcja o nazwie „podwojona” przyjmuje jeden parametr a. Funkcja wypisuje na konsoli dwukrotność podanego parametru.
<pre><b>def iloczyn(a, b) {     return a*b; }</b></pre>	Funkcja o nazwie „iloczyn” przyjmuje dwa parametry a i b. Funkcja zwraca iloczyn podanych parametrów.

## AUTORZY

Autorem wizji nauczania podstaw programowania i technologii InstaKod, w tym języków Assembly i Nianiolang, jest Andrzej Gąsienica-Samek. Wieloletnie uczestnictwo, doświadczenia i sukcesy w zawodach informatycznych skłoniły go do stworzenia ścieżki nauki programowania, najpierw dla uzdolnionych uczniów szkół średnich, a następnie dla zainteresowanych uczniów II etapu edukacyjnego. Zwieńczeniem tych projektów było napisanie dla najmłodszych uczniów języka wizualnego Assembly oraz stworzenie pierwszego zbioru zadań.

Współautorem programu InstaKod do zajęć informatyki dla klas 4-8 w zakresie podstaw programowania jest profesor Krzysztof Diks z UW, który uczestniczył w jego tworzeniu od pierwszych szkiców. Pasja do uczenia również najmłodszych uczniów, jak i głębokie rozumienie potrzeby popularyzowania informatyki, pozwoliły na stworzenie programu, który umożliwia wszystkim uczniom szkół podstawowych przejście ścieżki podstaw programowania, a uczniom zdolnym wejście na ścieżkę olimpiady informatycznej.

Współautorem platformy InstaKod jest Tomasz Stachowicz. Jego doświadczenie w zarządzaniu projektami informatycznymi, w szczególności platformą InstaLing do nauki słówek, z której od wielu lat korzystają uczniowie w polskich szkołach (aktualnie 140 000), pozwoliło na zbudowanie funkcjonalnego systemu informatycznego i stworzenie jego biznes planu zapewniającego dalsze samofinansowanie.

Program wraz z obudową metodyczną, podręcznikami, zbiorami zadań na platformie InstaKod stworzony został przez zespół pod kierownictwem Ewy Gąsienicy-Samek na bazie wielu lat pracy z uczniami stawiającymi pierwsze kroki w nauce programowania i algorytmiki. W zespole tym uczestniczyli:

Joanna Śmigielska – konsultant programu i podręcznika,

Katarzyna Olędzka – autor pierwszych rozdziałów podręcznika klasy 4 oraz konsultant podręcznika,

Michał Ryczkowski – współautor działu technologii informacyjnych podręcznika klasy 4,

Adam Djellouli – współautor programu i zbioru zadań,

Sebastian Warchoł, Jerzy Kinasz, Kajetan Sokolnicki – administratorzy bazy zadań InstaKod, twórcy rozwiązań wzorcowych przykładów, zadań i kart pracy, testerzy platformy,

Anna Stasiuk – konsultant językowy podręcznika i pomysłodawca części projektów międzyprzedmiotowych w dziale technologii informacyjnych.

Nauka informatyki z *InstaKod* opiera się na pracy na platformie on-line, którą tworzy młody, twórczy zespół techniczny pod kierownictwem Andrzeja Gąsienicy-Samka i Rafała Wyki: Jakub Bujak i Mateusz Woźniak.

Wśród instytucji, które wspierały tworzenie InstaKod należy wymienić:

Uniwersytet Warszawski, Wydział Matematyki, Informatyki i Mechaniki. Studenci MIM UW, w ramach licencjackiego zespołowego projektu realizowanego w Atinea Sp. z o.o. rozbudowali platformę InstaKod oraz zbiór zadań i samouczków w zakresie podstaw programowania dla klas 7-8.

OEIIZK w Warszawie, gdzie odbywają się szkolenia dla nauczycieli z zakresu podstaw programowania w InstaKod; na bazie doświadczeń tych szkoleń powstała platforma szkoleniowa on-line, z której mogą korzystać zdalnie nauczyciele z całej Polski,

SP nr 96 W Warszawie, w której zespół InstaKod poprowadził pilotażowy program z podstaw programowania w klasach 4, 5 i 6 w II semestrze roku szkolnego 2017/2018;

SP Przymierza Rodzin im. Jana Pawła II w Warszawie, w której nauczycielka informatyki, Pani Małgorzata Witczyńska, przeprowadziła samodzielnie pierwszy program pilotażowy z ponad 300 uczniami w pierwszym semestrze 2018/2019 roku.

## AUTORZY

---

Swój wkład w rozwój i formę InstaKod ma 50 nauczycieli, którzy w II semestrze 2018/2019 wzięli udział w I Edycji InstaKod dla szkół pracując łącznie z ponad 6000 uczniów. Ich sugestie stanowiły bazę do zmian metodycznych i technicznych na platformie InstaKod.