

Wymagania edukacyjne niezbędne do otrzymania przez ucznia poszczególnych śródrocznych i rocznych ocen klasyfikacyjnych z informatyki w klasie drugiej .

Wymagania na oceny śródroczne (I półrocze) obejmują wymagania z działu I, zaś na oceny roczne obejmują wszystkie wymagania z działów od I do II włącznie (cały rok szkolny)

W związku z uszczupleniem przez MEN podstawy programowej, w rozkładzie materiału zmniejszyła się liczba godzin na realizację obowiązkowych zagadnień. Uzyskane w ten sposób dodatkowe godziny pozostają do dyspozycji nauczyciela w trakcie roku szkolnego. Zgodnie z założeniami MEN: *Ograniczony zakres treści nauczania – wymagań szczegółowych – da nauczycielom i uczniom więcej czasu na spokojniejszą i bardziej dogłębną realizację programów nauczania.*

Lp.	Temat	Liczba godzin	Zapisy podstawy programowej
Rozdział 1. Arkusz kalkulacyjny i bazy danych			
1	Jak pobierać dane do arkusza kalkulacyjnego	3	II.3c, II.4
2	Wyciągamy wiedzę z danych	4	II.3c, II.4
3	Korespondencja seryjna	2	II.2, II.3b, II.3c, II.3d
4	Relacyjne bazy danych	3	II.2, II.3c, II.3d
P1	Współdziałanie aplikacji – projekt zespołowy	3	II.3b, II.3c, II.3.d, II.4, IV.5
Rozdział 2. Algorytmika i programowanie w języku Python			
5	Od problemu do programu	4	I.1, I.3, II.1, II.2
6	Wyszukujemy i sumujemy	2	I.1, I.3, II.1, II.2
7	Binarny system liczbowy	3	I.2a, I.3, II.1
8	Czy ta liczba jest pierwsza?	3	I.2a, I.3, II.1
9	Algorytm Euklidesa i działania na ułamkach	3	I.2a, I.3, II.1
Suma godzin			30

Plan wynikowy – Informatyka na czasie, część 2 (Python)

Lp.	Temat	Liczba godzin	Osiągnięcia uczniów	
			Wymagania podstawowe. Uczeń:	Wymagania ponadpodstawowe. Uczeń:
1	Jak pobierać dane do arkusza kalkulacyjnego	3	– wyjaśnia, dlaczego warto stosować narzędzia wymiany danych – wymienia podstawowe zastosowania arkusza kalkulacyjnego – wyjaśnia podstawowe pojęcia związane z arkuszem kalkulacyjnym: skoroszyt, arkusz, adres komórki, formuła, funkcja, zakres adresów – pobiera dane z różnych źródeł i przetwarza je – modyfikuje dane w arkuszu – wykorzystuje adresy komórek w formułach obliczeniowych – wyjaśnia różnice między formułami i funkcjami – korzysta z wbudowanych funkcji arkusza kalkulacyjnego – stosuje różne sposoby zaznaczania zakresów komórek – kopiuje dane z komórek i wkleja je na różne sposoby, również między arkuszami – kopiuje formuły – stosuje funkcje: SUMA, ŚREDNIA, MAX, MIN, DŁ, JEŻELI – przedstawia dane w postaci wykresów	– wyszukuje samodzielnie w internecie dane potrzebne do realizacji określonych zadań – importuje do arkusza dane z różnych źródeł, w tym ze stron WWW – buduje złożone formuły pozwalające wykonywać obliczenia, rozwiązujące określone problemy – poprawnie stosuje adresowanie względne, bezwzględne i mieszane – modyfikuje dane podczas ich importowania – wyszukuje w internecie informacje na temat nowych funkcji i stosuje je w zadaniach – dobiera typ wykresu do rodzaju danych – interpretuje otrzymane wyniki zgodnie z ustalonymi założeniami
2	Wyciągamy wiedzę z danych	4	– omawia różnicę między filtrowaniem i sortowaniem danych – filtruje i sortuje dane – tworzy tabele i stosuje w nich sortowanie i filtrowanie danych – stosuje formuły arkusza kalkulacyjnego do losowego generowania zbiorów danych – stosuje funkcję INDEKS do zwracania wartości określonych komórek – opisuje możliwości tabel przestawnych	– generuje zestawy danych za pomocą narzędzi online – modyfikuje style tabel przestawnych – buduje tabele przestawne dla dużych zbiorów danych – tworzy fragmentatory i korzysta z osi czasu – interpretuje wyniki tabel i wykresów przestawnych – stosuje tabele przestawne do rozwiązywania złożonych zadań, w których wykorzystano duże zbiory danych

			– tworzy tabele przestawne – filtruje dane w tabeli przestawnej – aktualizuje tabelę przestawną po modyfikacji danych źródłowych – stosuje gotowe style tabel przestawnych – podsumowuje dane w tabeli przestawnej na różne sposoby – stosuje różne sposoby wyświetlania wartości w tabeli przestawnej – grupuje i rozgrupowuje daty w tabelach przestawnych – tworzy wykresy przestawne	
3	Korespondencja seryjna	2	– tworzy dokument główny korespondencji seryjnej – umieszcza pola korespondencji seryjnej w tworzonych dokumentach – tworzy bazę adresatów – stosuje reguły warunkowe do personalizacji listów seryjnych – poprawnie scala dokumenty seryjne	– zarządza danymi adresatów korespondencji seryjnej w arkuszu kalkulacyjnym – tworzy zestawy dokumentów seryjnych (listy, etykiety, koperty) – drukuje listy seryjne – wysyła korespondencję seryjną za pomocą poczty elektronicznej
4	Relacyjne bazy danych	3	– wyjaśnia podstawowe pojęcia związane z bazami danych: tabela, atrybut, rekord, pole, klucz główny, klucz obcy, relacja – wymienia różne zastosowania baz danych – projektuje nieduże bazy danych – zarządza danymi w bazie danych w programie MS Access – modyfikuje dane zawarte w bazie danych – tworzy tabele i definiuje relacje między nimi	– tworzy kwerendy w programie MS Access
P1	Współdziałanie aplikacji – projekt zespołowy	3	– współpracuje w grupie, korzystając z narzędzi online – korzysta z programów graficznych podczas pracy nad zadaniem projektowym – stosuje funkcje arkusza kalkulacyjnego do przetwarzania danych – testuje rozwiązania wypracowane w grupie – korzysta z zasobów internetowych, wyszukując potrzebne	– prezentuje efekty pracy grupowej na forum klasy – przyjmuje rolę lidera odpowiedzialnego za zespół i projekt

			informacje – stosuje zasady netykiety i korzysta z niej w komunikacji zdalnej	
5	Od problemu do programu	4	– wyjaśnia pojęcie algorytmu – podaje przykłady algorytmów niekomputerowych – wymienia cechy poprawnego algorytmu – wyjaśnia na przykładzie pojęcie specyfikacji problemu – tworzy algorytm wyznaczania pierwiastka kwadratowego – zapisuje algorytm Herona w postaci listy kroków – wyjaśnia pojęcia związane z algorytmiką i programowaniem: schemat blokowy, lista kroków, kod źródłowy, kod wynikowy, kompilator, interpreter, słowa kluczowe, funkcje, plik wykonywalny – zapisuje algorytm w postaci kodu źródłowego – kompiluje zapisany kod źródłowy – znajduje i poprawia błędy w kodzie źródłowym programu – wyjaśnia pojęcie zmiennej i typu zmiennej – wymienia zasady tworzenia kodu źródłowego w wybranym języku programowania – stosuje podstawowe konstrukcje wybranego języka programowania: instrukcje wejścia i wyjścia, operatory arytmetyczne i logiczne oraz instrukcję warunkową – tworzy program sprawdzający warunek trójkąta	– tworzy samodzielnie programy, wykorzystując poznane instrukcje wybranego języka programowania – stosuje w swoich programach zagnieżdżone instrukcje warunkowe – pisze programy rozwiązujące zadania matematyczne i fizyczne oraz problemy z napisami
6	Wyszukujemy i sumujemy	2	– wyjaśnia, na czym polega iteracyjne rozwiązywanie problemu – stosuje w swoich programach podstawowe rodzaje pętli: while i for – opisuje zasady użycia pętli w programach – analizuje w tabeli działanie algorytmu krok po kroku – opisuje, jak komputer porównuje dwie wartości – tworzy program wyszukiwania największej liczby z zadanego ciągu liczb	– stosuje różne rodzaje pętli, dostosowując wybór do rozwiązywanego problemu – stosuje w kodzie operatory inkrementacji i dekrementacji – używa w algorytmach rozwiązania z wartownikiem, czyli specjalną wartością wskazującą koniec ciągu wczytywanych do programu wartości – analizuje kod źródłowy i poprawia ewentualne błędy

7	Binarny system liczbowy	3	– zapisuje liczby w binarnym systemie liczbowym oraz w systemie szesnastkowym – wyjaśnia pojęcia związane z systemami liczbowymi: system pozycyjny, podstawa systemu liczbowego – omawia algorytm konwersji liczb między systemami dziesiętnym i binarnym – zapisuje algorytm konwersji między systemami liczbowymi w postaci programu komputerowego	– stosuje operację dzielenia całkowitego w rozwiązywaniu problemów – tworzy algorytmy konwersji między różnymi systemami liczbowymi – programuje algorytmy konwersji między różnymi systemami liczbowymi, stosując strukturę tablicy lub listy
8	Czy ta liczba jest pierwsza?	3	– definiuje liczby złożone i liczby pierwsze – podaje przykłady użycia liczb pierwszych – omawia algorytmy sprawdzające podzielność liczb – bada podzielność wybranych liczb, programując poznane algorytmy w wybranym języku – grupuje instrukcje w funkcje i wyjaśnia cel stosowania funkcji	– tworzy samodzielnie programy dla poznanych algorytmów – wyjaśnia pojęcia liczb doskonałych, bliźniaczych, zaprzyjaźnionych – pisze programy wykorzystujące poznane rodzaje liczb pierwszych – wyjaśnia praktyczne znaczenie liczb pierwszych w informatyce
9	Algorytm Euklidesa i działania na ułamkach	1	– wyjaśnia pojęcia: NWD, NWW – podaje przykłady zastosowania algorytmu Euklidesa – zapisuje algorytm Euklidesa w postaci listy kroków – tworzy program pozwalający na dodawanie ułamków – stosuje odpowiednie konstrukcje wybranego języka programowania do implementacji omawianych zagadnień (w tym: funkcję, która nie zwraca wartości)	– tworzy programy realizujące działania na ułamkach – opisuje algorytm Euklidesa i tworzy realizujący go program w wybranym języku programowania – opisuje różnicę w sprawności dwóch wersji algorytmu Euklidesa: z odejmowaniem i z dzieleniem – poznaje inne zastosowania algorytmu Euklidesa, wykorzystując informacje zawarte w internecie lub innych źródłach

PRZEDMIOTOWE ZASADY OCENIANIA Z INFORMATYKI

I. Postanowienia ogólne

Przedmiotowe Zasady Oceniania zostały opracowane na podstawie:

1. Rozporządzenia Ministra Edukacji Narodowej w sprawie warunków i sposobu oceniania, klasyfikowania i promowania uczniów w szkołach oraz przeprowadzania sprawdzianów i egzaminów w szkołach publicznych;
2. Programu nauczania informatyki,
3. Podstawy programowej kształcenia ogólnego z informatyki
4. Wewnątrzszkolnego Systemu Oceniania;
5. Statutu szkoły.

II. Przedmiotem oceny są

- wiedza i umiejętności wynikające z podstawy programowej nauczania informatyki oraz wymagań programu nauczania;
- aktywność i systematyczność
- wysiłek wkładany przez ucznia

III. Ocenie podlegają

- Sprawdziany z zakresu materiału programowego obejmującego więcej niż trzy ostatnie lekcje, zapowiedziane z tygodniowym wyprzedzeniem. Uczeń ma prawo do poprawy niezaliczonego sprawdzianu
- Kartkówki, które nie są zapowiadane i obejmują trzy ostatnie lekcje
- Prace domowe
- Odpowiedzi ustne
- Aktywność na lekcji
- Praca w grupie

IV. Kryteria i sposoby oceniania

1. W pisemnych formach sprawdzania wiadomości i umiejętności ustala się następujące kryteria ocen wyrażone w procentach:

Stopień celujący	96% i powyżej
Stopień bardzo dobry	Między 86-95%
Stopień dobry	Między 71-85%
Stopień dostateczny	Między 51-70%
Stopień dopuszczający	Między 40-50%
Stopień niedostateczny	Poniżej 40%

2. Ogólne kryteria ocen z informatyki:

Ocenę celującą otrzymuje uczeń, który:

- spełnia kryteria oceny co najmniej bardzo dobrej
- z powodzeniem bierze udział w konkursach oraz olimpiadach informatycznych na szczeblu co najmniej regionalnym
- wykazuje inicjatywę rozwiązywania konkretnych problemów w czasie lekcji i pracy pozalekcyjnej
- umie formułować oryginalne przemyślane wnioski, hierarchizować i selekcjonować nabywaną wiedzę
- w bardzo wysokim stopniu opanował treści programowe, rozszerzając swą wiedzę o wiadomości w znacznym stopniu wykraczające poza program klasy

Ocenę bardzo dobrą otrzymuje uczeń, który:

- spełnia kryteria oceny co najmniej dobrej;
- w bardzo wysokim stopniu opanował treści programowe;
- wnosi twórczy wkład w realizowanie zagadnień;
- biegle i poprawnie posługuje się terminologią informatyczną;
- biegle i bezpiecznie obsługuje komputer;
- samodzielnie rozwiązuje problemy wynikające w trakcie wykonywania zadań programowych;
- stosuje poznane metody do rozwiązywania nowych, własnych problemów;
- potrafi zdefiniować złożony problem, określić plan działania dzieląc problem na podproblemy

Ocenę dobrą otrzymuje uczeń, który:

- spełnia kryteria oceny co najmniej dostatecznej;
- opanował treści konieczne, podstawowe i dopełniające;
- umie samodzielnie pracować z podręcznikiem, materiałem źródłowym;
- ustnie i pisemnie stosuje terminologię typową dla danego przedmiotu;
- rozwiązuje typowe problemy z wykorzystaniem poznanych metod oraz różnorodnych źródeł informacji;
- dokonuje trafnej analizy typowych problemów, potrafi sformułować logiczne wnioski;
- bierze aktywny udział w zajęciach, sprawnie pracuje w grupie;
- posługuje się terminologią informatyczną;
- poprawnie i bezpiecznie obsługuje komputer;
- z pomocą nauczyciela rozwiązuje problemy wynikające w trakcie wykonywania zadań programowych;

Ocenę dostateczną otrzymuje uczeń, który:

- spełnia kryteria oceny co najmniej dopuszczającej;
- opanował treści konieczne i podstawowe;
- rozumie treści określone programem nauczania;
- z minimalną pomocą nauczyciela rozwiązuje typowe problemy;
- analizuje podstawowe zależności;
- próbuje porównywać, wnioskować, zajmować stanowisko;
- zna terminologię informatyczną, ale ma trudności z jej zastosowaniem;
- nie potrafi rozwiązać problemów wynikających w trakcie wykonywania zadań programowych, nawet z pomocą nauczyciela;

Ocenę dopuszczającą otrzymuje uczeń, który:

- opanował treści konieczne, przewidziane w podstawie programowej;
- ma braki w podstawowych wiadomościach, lecz potrafi je nadrobić według wskazówek nauczyciela;
- podporządkowuje się instrukcjom nauczyciela i współpracuje z nim w celu nadrobienia zaległości;
- przejawia gotowość do przyswajania nowych wiadomości;
- częściowo zna terminologię informatyczną, ale nie potrafi jej zastosować;
- zadaną pracę wykonuje z pomocą nauczyciela;

Ocenę niedostateczną otrzymuje uczeń, który:

- nie opanował wiadomości programowych w zakresie koniecznym, umożliwiającym mu przejście do wyższego poziomu kształcenia;
- braki wiedzy są na tyle duże, że nie roszą nadziei na ich usunięcie nawet przy pomocy nauczyciela;
- nie potrafi wykonać prostych poleceń wymagających zastosowania podstawowych umiejętności;
- nie potrafi wykonać zadań teoretycznych i praktycznych o niskim stopniu trudności;
- nie potrafi wypowiadać się w sposób spójny i logiczny;
- nie zna terminologii informatycznej;
- nie stosuje bezpiecznej obsługi komputera;
- wykonuje ćwiczenia niespójne, chaotycznie z licznymi błędami;
- nie pracuje systematycznie, nie włącza się aktywnie w przebieg lekcji, nie odpowiada na pytania nauczyciela.

VI. Szczegółowe zasady okresowego podsumowania osiągnięć edukacyjnych

- Na początku roku szkolnego uczniowie zostają poinformowani przez nauczyciela o zakresie wymagań z informatyki, obowiązującym w danym roku (zakres wiadomości i umiejętności, które trzeba mieć opanowane na

koniec roku szkolnego) oraz o sposobie i zasadach oceniania z danego przedmiotu.

- Nieobecność ucznia na lekcji zobowiązuje go do uzupełniania materiału we własnym zakresie
- Na lekcji uczeń może być oceniony za pracę na lekcji: odpowiedź, aktywność, wykonywane ćwiczenia lub brak pracy
- Korzystanie przez ucznia w czasie sprawdzianów, kartkówek i innych form sprawdzania wiedzy z niedozwolonych przez nauczyciela pomocy stanowi podstawę do wystawienia oceny niedostatecznej
- Poprawa sprawdzianu może być dokonana w terminie dwóch tygodni od daty podania oceny i brane są pod uwagę obie oceny
- Uczeń, który był nieobecny na sprawdzianie, zalicza go w terminie ustalonym z nauczycielem
- Kartkówki nie podlegają poprawie
- Ocenę semestralną i roczną nauczyciel wystawia na podstawie ocen częściowych uzyskanych przez ucznia, liczone poprzez średnią ważoną
- Uczeń jest zobowiązany przygotować się do lekcji z 3 ostatnich tematów
- Uczeń może dwa razy na semestr skorzystać z „nieprzygotowania” i nie być brany pod uwagę przy wyborze do odpowiedzi ustnej.
- Uczeń może uzyskiwać za aktywność na lekcji „plusy”. Po zebraniu odpowiedniej liczby plusów są one kasowane a uczeń otrzymuje ocenę bardzo dobrą z aktywności.