

**WYMAGANIA EDUKACYJNE
Z INFORMATYKI
DLA UCZNIÓW KLASY VIII**

SPOSODY SPRAWDZANIA OSIĄGNIĘĆ, WARUNKI UZYSKIWANIA WYŻSZYCH STOPNI

1. Wymagania konieczne (na ocenę dopuszczającą) obejmują wiadomości i umiejętności umożliwiające uczniowi dalszą naukę, bez których nie jest on w stanie zrozumieć kolejnych zagadnień omawianych na lekcjach i wykonywać prostych zadań nawiązujących do życia codziennego.
2. Wymagania podstawowe (na ocenę dostateczną) obejmują wiadomości i umiejętności stosunkowo łatwe do opanowania, przydatne w życiu codziennym, bez których nie jest możliwe kontynuowanie nauki.
3. Wymagania rozszerzające (na ocenę dobrą) obejmują wiadomości i umiejętności o średnim stopniu trudności, które są przydatne na kolejnych poziomach kształcenia.
4. Wymagania dopełniające (na ocenę bardzo dobrą) obejmują wiadomości i umiejętności złożone, o wyższym stopniu trudności, wykorzystywane do rozwiązywania zadań problemowych.
5. Wymagania wykraczające (na ocenę celującą) obejmują stosowanie zdobytych wiadomości i umiejętności w sytuacjach trudnych, złożonych i nietypowych.

Wymagania konieczne (ocena dopuszczająca) Uczeń:	Wymagania podstawowe (ocena dostateczna) Uczeń:	Wymagania rozszerzające (ocena dobra) Uczeń:	Wymagania dopełniające (ocena bardzo dobra) Uczeń:	Wymagania wykraczające (ocena celująca) Uczeń:
• Arkusz kalkulacyjny				
• omawia zastosowanie oraz budowę arkusza kalkulacyjnego • określa adres komórki • wprowadza dane różnego rodzaju do komórek arkusza kalkulacyjnego • formatuje zawartość komórek (wyrównanie tekstu oraz wygląd czcionki) • rozumie różnice między adresowaniem względnym, bezwzględnym i mieszanym • wstawia wykres do arkusza kalkulacyjnego • korzysta z arkusza kalkulacyjnego w celu stworzenia kalkulacji wydatków	• określa zasady wprowadzania danych do komórek arkusza kalkulacyjnego • dodaje i usuwa wiersze oraz kolumny w tabeli • stosuje w arkuszu podstawowe funkcje: (SUMA, ŚREDNIA), wpisuje je ręcznie oraz korzysta z kreatora • omawia i modyfikuje poszczególne elementy wykresu • zapisuje w tabeli arkusza kalkulacyjnego dane otrzymane z prostych doświadczeń i przedstawia je na wykresie	• tworzy proste formuły obliczeniowe • wyjaśnia, czym jest adres względny • wykorzystuje funkcję JEŻELI do tworzenia algorytmów z warunkami w arkuszu kalkulacyjnym • ustawia format danych komórki odpowiadający jej zawartości • w formułach stosuje adresowanie względne, bezwzględne i mieszane • dobiera odpowiedni wykres do rodzaju danych • sortuje oraz filtruje dane w arkuszu kalkulacyjnym	• kopiuje utworzone formuły obliczeniowe, wykorzystując adresowanie względne • korzysta z biblioteki funkcji, aby wyszukiwać potrzebne funkcje • stosuje adresowanie względne, bezwzględne lub mieszane w zaawansowanych formułach obliczeniowych • tworzy wykres dla więcej niż jednej serii danych • tworzy prosty model (na przykładzie rzutu sześciennej kostką do gry) w arkuszu kalkulacyjnym • stosuje filtry niestandardowe	• samodzielnie tworzy i kopiuje skomplikowane formuły obliczeniowe • stosuje zaawansowane funkcje arkusza w tabelach tworzonych na własne potrzeby • tworzy rozbudowane wykresy dla wielu serii danych • przygotowuje rozbudowane arkusze kalkulacyjne korzysta z arkusza kalkulacyjnego do analizowania doświadczeń z innych przedmiotów

• Programowanie w języku C++				
- definiuje pojęcia: algorytm, program, programowanie - podaje kilka sposobów przedstawienia algorytmu - tłumaczy, do czego używa się zmiennych w programach - pisze proste programy w trybie skryptowym języka C++ z wykorzystaniem zmiennych - wyjaśnia działanie operatora modulo - wyjaśnia algorytm badania podzielności liczb - wyjaśnia potrzebę wyszukiwania informacji w zbiorze - określa różnice między wyszukiwaniem w zbiorach uporządkowanym i nieuporządkowanym - sprawdza działanie programów wyszukujących element w zbiorze - wyjaśnia potrzebę porządkowania danych - sprawdza działanie programu sortującego dla różnych danych	- wymienia różne sposoby przedstawienia algorytmu: opis słowny, schemat blokowy, lista kroków - poprawnie formułuje problem do rozwiązania - wyjaśnia różnice między interaktywnym a skryptowym trybem pracy - stosuje odpowiednie polecenie języka C++, aby wyświetlić tekst na ekranie - omawia różnice pomiędzy kodem źródłowym a kodem wynikowym - tłumaczy, czym jest środowisko programistyczne - wykonuje obliczenia w języku C++ - omawia działanie operatorów arytmetycznych - stosuje tablice w języku C++ oraz operatory logiczne - zapisuje w postaci listy kroków algorytm badania podzielności liczb naturalnych - wykorzystuje w programach instrukcję iteracyjną <code>while</code> - zapisuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym, w tym elementu największego i najmniejszego - zapisuje algorytm wyszukiwania elementu w zbiorze uporządkowanym metodą połowienia - implementuje grę w zgadywanie liczby - zapisuje w wybranej formie algorytm porządkowania metodami przez wybieranie oraz przez zliczanie - omawia implementację algorytmu sortowania przez wybieranie - stosuje pętle zagnieżdżone i wyjaśnia, jak działają	- wymienia przykładowe środowiska programistyczne - wyjaśnia, czym jest specyfikacja problemu - opisuje etapy rozwiązywania problemów - opisuje etapy powstawania programu komputerowego - zapisuje proste polecenia języka C++ - wykorzystuje instrukcję warunkową <code>if</code> oraz <code>if else</code> w programach - wykorzystuje iterację w konstruowanych algorytmach - wykorzystuje w programach instrukcję iteracyjną <code>for</code> - definiuje funkcje w języku C++ i omawia różnice między funkcjami zwracającymi wartość a funkcjami niezwracającymi wartości - omawia algorytm Euklidesa w wersji z odejmowaniem i z dzieleniem – zapisuje go w wybranej postaci - wyjaśnia algorytm wyodrębniania cyfr danej liczby i zapisuje go w wybranej postaci - implementuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym - omawia funkcje zastosowane w realizacji algorytmu wyszukiwania metodą połowienia - implementuje algorytm wyszukiwania największej wartości w zbiorze - omawia implementację algorytmu sortowania przez zliczanie - omawia funkcje zastosowane w kodzie źródłowym algorytmów sortowania przez wybieranie oraz przez zliczanie	- pisze proste programy w trybie skryptowym języka C++ - buduje złożone schematy blokowe służące do przedstawiania skomplikowanych algorytmów - konstruuje złożone sytuacje warunkowe (wiele warunków) w algorytmach - pisze programy zawierające instrukcje warunkowe, pętle oraz funkcje - wyjaśnia, jakie błędy zwraca interpreter - czyta kod źródłowy i opisuje jego działanie - wyjaśnia różnice między instrukcją iteracyjną <code>while</code> a pętlą <code>for</code> - pisze programy obliczające NWD, stosując algorytm Euklidesa, oraz wypisujące cyfry danej liczby - wyjaśnia różnice między algorytmem Euklidesa w wersjach z odejmowaniem i z dzieleniem - samodzielnie zapisuje w wybranej postaci algorytm wyszukiwania elementu w zbiorze metodą połowienia, w tym elementu największego i najmniejszego - implementuje algorytm wyszukiwania elementu w zbiorze metodą połowienia - implementuje algorytmy porządkowania metodami przez wybieranie oraz przez zliczanie - wprowadza modyfikacje w implementacji algorytmów porządkowania przez wybieranie oraz przez zliczanie	- zapisuje algorytmy różnymi sposobami oraz pisze programy o większym stopniu trudności - pisze programy w języku C++ do rozwiązywania zadań matematycznych - tworzy program składający się z kilku funkcji wywoływanych w programie głównym - pisze programy wykorzystujące algorytmy Euklidesa (np. obliczający NWW) oraz wyodrębniania cyfr danej liczby - samodzielnie modyfikuje i optymalizuje algorytmy wyszukiwania - samodzielnie modyfikuje i optymalizuje programy sortujące metodą przez wybieranie, metodą przez zliczanie

• Projekty				
• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, wykonując powierzone mu zadania o niewielkim stopniu trudności • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności • testuje grę na różnych etapach • współpracuje w grupie podczas pracy nad projektem • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności – znalezienie informacji w internecie, umieszczenie ich w chmurze • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności • bierze aktywny udział w dyskusji nad wyborem atrakcyjnego zawodu wymagającego kompetencji informatycznych	• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej • wprowadza dane do zaprojektowanych tabel • bierze udział w pracach nad wypracowaniem koncepcji gry • współpracuje w grupie podczas pracy nad projektem • współpracuje z innymi podczas pracy nad projektem • analizuje zebrane dane • tworzy projekt prezentacji multimedialnej • gromadzi informacje dotyczące wybranych zawodów, umieszcza je w zaprojektowanych tabelach i dokumentach tekstowych	• przygotowuje dokumentację imprezy, wykonuje obliczenia, projektuje tabele oraz wykresy • współpracuje w grupie podczas pracy nad projektem • programuje wybrane funkcje i elementy gry • opracowuje opis gry • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania • tworzy prezentację wg projektu zaakceptowanego przez zespół • aktywnie uczestniczy w pracach zespołu • projektuje tabele do zapisywania informacji o zawodach • weryfikuje i formatuje przygotowane dokumenty tekstowe	• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, przygotowuje zestawienia, drukuje wyniki • współpracuje w grupie podczas pracy nad projektem • implementuje i optymalizuje kod źródłowy gry, korzystając z wypracowanych założeń • aktywnie uczestniczy w pracach zespołu • analizuje i weryfikuje pod względem merytorycznym i technicznym przygotowaną prezentację • aktywnie uczestniczy w pracach zespołu, weryfikuje opracowane treści i łączy wszystkie dokumenty w całość	• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, tworzy zestawienia zawierające zaawansowane formuły, wykresy oraz elementy graficzne • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera • rozbudowuje grę o nowe elementy • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera • wzbogaca prezentację o elementy podnoszące jej walory estetyczne i merytoryczne • aktywnie uczestniczy w pracach zespołu, przyjmuje rolę lidera • podczas dyskusji przyjmuje funkcję moderatora

SPOSOBY SPRAWDZANIA OSIĄGNIĘĆ, WARUNKI UZYSKIWANIA WYŻSZYCH STOPNI z informatyki są zgodne z zasadami oceniania wewnątrzszkolnego i znajdują się w rozdziale 7 Statutu Szkoły.

mgr Mariusz Adaszek

na podstawie Przedmiotowego systemu oceniania. Wymagania na poszczególne oceny – Nowa Era