

Wymagania na poszczególne oceny z informatyki - klasa 8

Wymagania na ocenę śródroczną

Wymagania na każdy stopień wyższy niż dopuszczający obejmują również wymagania na stopień poprzedni.

Dopuszcza się kontrolowanie wiedzy z tego, co uczniowie opanowali we wcześniejszych latach nauki/możliwe jest odwoływanie się do wiedzy i umiejętności z poprzednich klas.

Arkusze kalkulacyjny				
<p>Wymagania z podstawy programowej - Treści nauczania:</p> <p>Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:</p> <ul style="list-style-type: none"> korzysta z aplikacji komputerowych, przygotowuje dokumenty i prezentacje, także w chmurze, dostosowuje format i wygląd opracowań do ich treści i przeznaczenia, wykazując się przy tym umiejętnościami: <ul style="list-style-type: none"> rozwiązywania zadań rachunkowych z programu nauczania z różnych przedmiotów w zakresie szkoły podstawowej, z codziennego życia: umieszcza dane w tabeli arkusza kalkulacyjnego, posługuje się podstawowymi funkcjami, stosuje adresowanie względne, bezwzględne i mieszane, przedstawia dane w postaci różnego typu wykresów, porządkuje i filtruje dane. zapisuje efekty swojej pracy w różnych formatach i przygotowuje wydruki; <p>Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:</p> <ul style="list-style-type: none"> prezentuje przykłady zastosowań informatyki w innych dziedzinach, w zakresie pojęć, obiektów oraz algorytmów. 				
Wymagania na ocenę dopuszczającą Uczeń:	Wymagania na ocenę dostateczną Uczeń:	Wymagania na ocenę dobrą Uczeń:	Wymagania na ocenę bardzo dobrą Uczeń:	Wymagania na ocenę celującą Uczeń:
<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> 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

<ul style="list-style-type: none"> wstawia wykres do arkusza kalkulacyjnego korzysta z arkusza kalkulacyjnego w celu stworzenia kalkulacji wydatków 	<p>prosty doświadczeń i przedstawia je na wykresie</p>		<p>kostką do gry) w arkuszu kalkulacyjnym</p> <ul style="list-style-type: none"> stosuje filtry niestandardowe 	
---	--	--	---	--

Programowanie w języku C++

Wymagania z podstawy programowej - Treści nauczania:

Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:

- formułuje problem w postaci specyfikacji (czyli 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;
- rozwija znajomość algorytmów i wykonuje eksperymenty z algorytmami, korzystając z pomocy dydaktycznych lub dostępnego oprogramowania do demonstracji działania algorytmów;

Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:

- 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 działu I pkt 2;

Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi. Uczeń:

- poprawnie posługuje się terminologią związaną z informatyką i technologią.

Wymagania na ocenę dopuszczającą Uczeń:	Wymagania na ocenę dostateczną Uczeń:	Wymagania na ocenę dobrą Uczeń:	Wymagania na ocenę bardzo dobrą Uczeń:	Wymagania na ocenę celującą Uczeń:
<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> wymienia różne sposoby przedstawienia algorytmu: opis słowny, schemat blokowy, lista kroków 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 wykonuje obliczenia w języku C++ 	<ul style="list-style-type: none"> wymienia przykładowe środowiska programistyczne opisuje etapy rozwiązywania problemów zapisuje proste polecenia języka C++ wykorzystuje instrukcję warunkową if oraz if else w programach wykorzystuje iterację w konstruowanych algorytmach wykorzystuje w programach instrukcję iteracyjną for 	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> 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

	<ul style="list-style-type: none"> • omawia działanie operatorów arytmetycznych • stosuje tablice w języku C++ oraz operatory logiczne 	<ul style="list-style-type: none"> • definiuje funkcje w języku C++ i omawia różnice między funkcjami zwracającymi wartość a funkcjami niezwracającymi wartości 		
--	--	--	--	--

Wymagania na ocenę roczną (Wymagania na ocenę roczną obejmują również zagadnienia na ocenę śródroczną)

Programowanie w języku C++ (Algotrymy)

Wymagania z podstawy programowej - Treści nauczania:

Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:

- formułuje problem (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;
- stosuje przy rozwiązywaniu problemów podstawowe algorytmy:
 - 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),
 - wyszukiwania i porządkowania: wyszukuje element w zbiorze uporządkowanym i nieuporządkowanym oraz porządkuje elementy w zbiorze metodą przez proste wybieranie i zliczanie;

Wymagania na ocenę dopuszczającą Uczeń:	Wymagania na ocenę dostateczną Uczeń:	Wymagania na ocenę dobrą Uczeń:	Wymagania na ocenę bardzo dobrą Uczeń:	Wymagania na ocenę celującą Uczeń:
<ul style="list-style-type: none"> •definiuje pojęcia: algorytm •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 	<ul style="list-style-type: none"> • zapisuje w postaci listy kroków algorytm badania podzielności liczb naturalnych • wykorzystuje w programach instrukcję iteracyjną while • zapisuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym, w tym elementu największego i najmniejszego • zapisuje algorytm wyszukiwania elementu w zbiorze 	<ul style="list-style-type: none"> • 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 	<ul style="list-style-type: none"> • wyjaśnia różnice między instrukcją iteracyjną while a pętlą for • 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 	<ul style="list-style-type: none"> • 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

<ul style="list-style-type: none"> • sprawdza działanie programu sortującego dla różnych danych 	<p>uporządkowanym metodą połowienia</p> <ul style="list-style-type: none"> • implementuje grę w zgadywanie liczby • zapisuje w wybranej formie algorytm porządkowania metodami przez wybieranie oraz przez zliczanie • omawia algorytm sortowania przez wybieranie • stosuje pętle zagnieżdżone i wyjaśnia, jak działają 	<ul style="list-style-type: none"> • 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 	<p>tym elementu największego i najmniejszego</p> <ul style="list-style-type: none"> • 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 	
--	--	---	---	--

Projekty

Wymagania z podstawy programowej - Treści nauczania:

Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:

- prezentuje przykłady zastosowań informatyki w innych dziedzinach, w zakresie pojęć, obiektów oraz algorytmów.

Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:

- projektuje, tworzy i testuje oprogramowanie sterujące robotem lub innym obiektem na ekranie lub w rzeczywistości
- korzysta z aplikacji komputerowych, przygotowuje dokumenty i prezentacje, także w chmurze, dostosowuje format i wygląd opracowań do ich treści i przeznaczenia, wykazując się przy tym umiejętnościami:

- rozwiązywania zadań rachunkowych z programu nauczania z różnych przedmiotów w zakresie szkoły podstawowej, z codziennego życia: umieszcza dane w tabeli arkusza kalkulacyjnego, posługuje się podstawowymi funkcjami, stosuje adresowanie względne, bezwzględne i mieszane, przedstawia dane w postaci różnego typu wykresów, porządkuje i filtruje dane,

- zapisuje efekty swojej pracy w różnych formatach i przygotowuje wydruki;

Rozwijanie kompetencji społecznych. Uczeń:

- Bierze udział w różnych formach współpracy: w programie w parach lub w zespole, realizuje projekty, projektuje, tworzy i prezentuje efekty wspólnej pracy,
- 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;
- przedstawia główne etapy w historycznym rozwoju informatyki i technologii

Przestrzeganie prawa i zasad bezpieczeństwa. Uczeń:

- 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;
- postępuje etycznie w pracy z informacjami;

Wymagania na ocenę dopuszczającą Uczeń:	Wymagania na ocenę dostateczną Uczeń:	Wymagania na ocenę dobrą Uczeń:	Wymagania na ocenę bardzo dobrą Uczeń:	Wymagania na ocenę celującą Uczeń:
<ul style="list-style-type: none"> • bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, wykonując powierzone mu zadania o niewielkim stopniu trudności • aktywnie uczestniczy w pracach zespołu, znajduje informacje w internecie, umieszcza je w chmurze • testuje grę na różnych etapach • współpracuje w grupie podczas pracy nad projektem • bierze aktywny udział w dyskusji nad wyborem atrakcyjnego zawodu wymagającego kompetencji informatycznych 	<ul style="list-style-type: none"> • bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej • wprowadza dane do zaprojektowanych tabel • bierze udział w pracach nad wypracowaniem koncepcji gry • analizuje zebrane dane • tworzy projekt prezentacji multimedialnej • gromadzi informacje dotyczące wybranych zawodów, umieszcza je w zaprojektowanych tabelach i dokumentach tekstowych 	<ul style="list-style-type: none"> • przygotowuje dokumentację imprezy, wykonuje obliczenia, projektuje tabele oraz wykresy • programuje wybrane funkcje i elementy gry • opracowuje opis gry • tworzy prezentację wg projektu zaakceptowanego przez zespół • projektuje tabele do zapisywania informacji o zawodach 	<ul style="list-style-type: none"> • bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, przygotowuje zestawienia, drukuje wyniki • implementuje i optymalizuje kod źródłowy gry, korzystając z wypracowanych założeń • analizuje i weryfikuje pod względem merytorycznym i technicznym przygotowaną prezentację • aktywnie uczestniczy w pracach zespołu, weryfikuje opracowane treści i łączy wszystkie dokumenty w całość 	<ul style="list-style-type: none"> • 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 • wzbogaca prezentację o elementy podnoszące jej walory estetyczne i merytoryczne • podczas dyskusji przyjmuje funkcję moderatora