

Wymagania na poszczególne oceny z informatyki – klasa 8

Zagadnienie	Numer i temat lekcji	ocena dopuszczająca Uczeń:	ocena dostateczna Uczeń:	ocena dobra Uczeń:	ocena bardzo dobra Uczeń:	ocena celująca Uczeń:
1. Arkusz kalkulacyjny						
1.1. Formuły i adresowanie względne w arkuszu kalkulacyjnym	1. i 2. Formuły i adresowanie względne w arkuszu kalkulacyjnym	<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) 	<ul style="list-style-type: none"> określa zasady wprowadzania danych do komórek arkusza kalkulacyjnego dodaje i usuwa wiersze oraz kolumny w tabeli 	<ul style="list-style-type: none"> tworzy proste formuły obliczeniowe wyjaśnia, czym jest adres względny 	<ul style="list-style-type: none"> kopiuje utworzone formuły obliczeniowe, 	<ul style="list-style-type: none"> samodzielnie tworzy i kopiuje skomplikowane formuły obliczeniowe
1.2. Funkcje oraz adresowanie bezwzględne i mieszane w arkuszu kalkulacyjnym	3. 4. 5. 6. Funkcje oraz adresowanie bezwzględne i mieszane w arkuszu kalkulacyjnym	<ul style="list-style-type: none"> rozumie różnice między adresowaniem względnym, bezwzględnym i mieszanym 	<ul style="list-style-type: none"> stosuje w arkuszu podstawowe funkcje: (SUMA, ŚREDNIA), wpisuje je ręcznie oraz korzysta z kreatora 	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> korzysta z biblioteki funkcji, aby wyszukiwać potrzebne funkcje stosuje adresowanie względne, bezwzględne lub mieszane w formułach obliczeniowych 	<ul style="list-style-type: none"> stosuje zaawansowane funkcje arkusza w tabelach stworzonych na własne potrzeby
1.3. Przedstawianie danych na wykresie	7. i 8. Przedstawianie danych na wykresie	<ul style="list-style-type: none"> wstawia wykres do arkusza kalkulacyjnego 	<ul style="list-style-type: none"> omawia i modyfikuje poszczególne elementy wykresu 	<ul style="list-style-type: none"> dobiera odpowiedni wykres do rodzaju danych 	<ul style="list-style-type: none"> tworzy wykres dla więcej niż jednej serii danych 	<ul style="list-style-type: none"> tworzy rozbudowane wykresy dla wielu serii danych
1.4. Zastosowania arkusza kalkulacyjnego	9. i 10. Zastosowania arkusza kalkulacyjnego	<ul style="list-style-type: none"> korzysta z arkusza kalkulacyjnego w celu stworzenia kalkulacji wydatków 	<ul style="list-style-type: none"> zapisuje w tabeli arkusza kalkulacyjnego dane otrzymane z prostych doświadczeń i przedstawia je na wykresie 	<ul style="list-style-type: none"> sortuje oraz filtruje dane w arkuszu kalkulacyjnym 	<ul style="list-style-type: none"> tworzy prosty model (na przykładzie rzutu sześcienną kostką do gry) w arkuszu kalkulacyjnym stosuje filtry niestandardowe 	<ul style="list-style-type: none"> przygotowuje rozbudowane arkusze kalkulacyjne korzysta z arkusza kalkulacyjnego do analizowania doświadczeń z innych przedmiotów

2. Programowanie w języku C++						
2.1. Wprowadzenie do programowania w języku C++	11. 12. 13. Wprowadzenie do programowania w języku C++	<ul style="list-style-type: none"> definiuje pojęcia: algorytm, program, programowanie podaje kilka sposobów przedstawienia algorytmu wymienia różne sposoby przedstawienia algorytmu: opis słowny, lista kroków 	<ul style="list-style-type: none"> poprawnie formułuje problem do rozwiązania omawia różnice pomiędzy kodem źródłowym a kodem wynikowym tłumaczy, czym jest środowisko programistyczne 	<ul style="list-style-type: none"> wymienia przykładowe środowiska programistyczne wyjaśnia, czym jest specyfikacja problemu opisuje etapy powstawania programu komputerowego stosuje odpowiednie polecenie języka C++, aby wyświetlić tekst na ekranie 	<ul style="list-style-type: none"> opisuje etapy rozwiązywania problemów zapisuje proste polecenia języka C++ 	<ul style="list-style-type: none"> zapisuje algorytmy różnymi sposobami pisze proste programy w języku C++
2.2. Piszemy programy w języku C++	14. 15. 16. 17. Piszemy programy w języku C++	<ul style="list-style-type: none"> tłumaczy, do czego używa się zmiennych w programach 	<ul style="list-style-type: none"> wykonuje obliczenia w języku C++ omawia działanie operatorów arytmetycznych stosuje tablice w języku C++ oraz operatory logiczne 	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> pisze programy zawierające instrukcje warunkowe, pętle oraz funkcje wyjaśnia, jakie błędy zwraca interpreter czyta kod źródłowy i opisuje jego działanie 	<ul style="list-style-type: none"> pisze programy w języku C++ do rozwiązywania zadań matematycznych tworzy program składający się z kilku funkcji wywołanych w programie głównym konstruuje złożone sytuacje warunkowe (wiele warunków) w algorytmach pisze proste programy w języku C++ z wykorzystaniem zmiennych
2.3. Algorytmy na liczbach naturalnych	18. 19. 20. 21. Algorytmy na liczbach naturalnych	<ul style="list-style-type: none"> zapisuje w postaci listy kroków algorytm badania podzielności liczb naturalnych 	<ul style="list-style-type: none"> wyjaśnia działanie operatora modulo wyjaśnia algorytm badania podzielności liczb 	<ul style="list-style-type: none"> omawia algorytm Euklidesa i zapisuje go w wybranej postaci wyjaśnia algorytm wyodrębniania cyfr danej liczby i zapisuje go w wybranej postaci 	<ul style="list-style-type: none"> wyjaśnia różnice między instrukcją iteracyjną <code>while</code> a pętlą <code>for</code> pisze programy obliczające NWD, stosując algorytm Euklidesa wykorzystuje w programach instrukcję iteracyjną <code>while</code> 	<ul style="list-style-type: none"> pisze programy wykorzystujące algorytm Euklidesa (np. obliczający NWW) oraz wyodrębniania cyfr danej liczby

2.4. Algorytmy wyszukiwania	22. 23. 24. Algorytmy wyszukiwania	<ul style="list-style-type: none"> • wyjaśnia potrzebę wyszukiwania informacji w zbiorze 	<ul style="list-style-type: none"> • zapisuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym 	<ul style="list-style-type: none"> • implementuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym 	<ul style="list-style-type: none"> • sprawdza działanie programów wyszukujących element w zbiorze 	<ul style="list-style-type: none"> • samodzielnie zapisuje w wybranej postaci algorytm wyszukiwania elementu w zbiorze
2.5. Algorytmy porządkowania	25. 26. 27. Algorytmy porządkowania	<ul style="list-style-type: none"> • wyjaśnia potrzebę porządkowania danych • sprawdza działanie programu sortującego dla różnych danych 	<ul style="list-style-type: none"> • zapisuje w wybranej formie algorytm porządkowania metodą przez wybieranie • omawia implementację algorytmusortowania przez wybieranie 	<ul style="list-style-type: none"> • implementuje algorytm porządkowania metodą przez wybieranie • stosuje pętle zagnieżdżone 	<ul style="list-style-type: none"> • omawia funkcje zastosowane w kodzie źródłowym algorytmu sortowania przez wybieranie • wyjaśnia, jak działają pętle zagnieżdżone 	<ul style="list-style-type: none"> • wprowadza modyfikacje w implementacji algorytmu porządkowania przez wybieranie
3. Projekty						
3. 1. Dokumentacja szkolnej imprezy sportowej	28. 29. 30. Dokumentacja szkolnej imprezy sportowej	<ul style="list-style-type: none"> • bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, wykonując powierzone mu zadania o niewielkim stopniu trudności 	<ul style="list-style-type: none"> • bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej • wprowadza dane do zaprojektowanych tabel 	<ul style="list-style-type: none"> • przygotowuje dokumentację imprezy, wykonuje obliczenia, projektuje tabele oraz wykresy • współpracuje w grupie podczas pracy nad projektem 	<ul style="list-style-type: none"> • bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, przygotowuje zestawienia, drukuje wyniki • współpracuje w grupie podczas pracy nad projektem 	<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