

INFORMATYKA KLASA 8

Wymagania edukacyjne niezbędne do uzyskania
poszczególnych śródrocznych i rocznych ocen
klasyfikacyjnych z informatyki w Szkole Podstawowej z
Oddziałami Integracyjnymi i Oddziałami Dwujęzycznymi nr
211 im J. Korczaka
oparte na programie nauczania informatyki w szkole
podstawowej „Lubię to!”

OKRES I						
Tytuł	Numer i temat lekcji	(ocena dopuszczająca) Uczeń:	(ocena dostateczna) Uczeń:	(ocena dobra) Uczeń:	(ocena bardzo dobra) Uczeń:	(ocena celująca) Uczeń:
DZIAŁ 1. GRAFIKA KOMPUTEROWA						
1.1. Tworzenie i modyfikowanie obrazów	1. - 3. Tworzenie i modyfikowanie obrazów	• tworzy rysunek za pomocą podstawowych narzędzi programu GIMP i zapisuje go w pliku • zaznacza fragmenty obrazu • wykorzystuje schowek do kopiowania i wklejania fragmentów obrazu	• omawia znaczenie warstw obrazu w programie GIMP • tworzy i usuwa warstwy w programie GIMP • umieszcza napisy na obrazie w programie GIMP zapisuje rysunki w różnych formatach graficznych	• używa narzędzi zaznaczania dostępnych w programie GIMP • zmienia kolejność warstw obrazu w programie GIMP • opisuje podstawowe formaty graficzne • wykorzystuje warstwy, tworząc rysunki w programie GIMP • rysuje figury geometryczne, wykorzystując narzędzia zaznaczania w programie GIMP	• łączy warstwy w obrazach tworzonych w programie GIMP • wykorzystuje filtry programu GIMP do poprawiania jakości zdjęć tworzy fotomontaże, kolaże w programie GIMP	• tworząc rysunki w programie GIMP, wykorzystuje narzędzia nieomówione na lekcji
1.2. Animacje w programie GIMP	4. - 7. Tworzenie animacji	• wyjaśnia, czym jest animacja	• dodaje gotowe animacje do obrazów wykorzystując filtry programu GIMP	• dodaje gotowe animacje dla kilku fragmentów obrazu: odtwarzane jednocześnie oraz odtwarzane po kolei	• tworzy animację poklatkową, wykorzystując warstwy w programie GIMP	• przedstawia proste historie poprzez animacje utworzone w programie GIMP
1.3. Tworzenie plakatu – zadanie projektowe	8. - 9 Tworzenie plakatu – zadanie projektowe	współpracuje w grupie, przygotowując plakat	• planuje pracę w grupie poprzez przydzielanie zadań poszczególnym jej członkom	• wyszukuje, zbiera i samodzielnie tworzy materiały niezbędne do wykonania plakatu • przestrzega praw autorskich podczas zbierania materiałów do projektu	• wykorzystuje chmurę obliczeniową do zbierania materiałów niezbędnych do wykonania plakatu	• planuje pracę w grupie i współpracuje z jej członkami, przygotowując dowolny projekt

DZIAŁ 2. PROJEKTY	
--------------------------	--

2.1. Informatyka w naszym przyszłym życiu	10. - 12. Informatyka w naszym przyszłym życiu	• bierze udział w przygotowaniu dokumentacji wykonując powierzone mu zadania o niewielkim stopniu trudności	• bierze udział w przygotowaniu • wprowadza dane do zaprojektowanych tabel • wyszukuje informacje o technologiach	• przygotowuje dokumentację, wykonuje ewentualne obliczenia , projektuje tabele oraz wykresy • współpracuje w grupie podczas pracy nad projektem	• bierze udział w przygotowaniu dokumentacji, przygotowuje zestawienia, drukuje wyniki • współpracuje w grupie podczas pracy nad projektem • nawiązuje do historii informatyki • projekt przedstawia rozwój informatyki i technologii	• bierze udział w przygotowaniu dokumentacji, tworzy zestawienia zawierające zaawansowane formuły, wykresy oraz elementy graficzne • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera
2.2. Sterowanie obiektem na ekranie	13.-16. Sterowanie obiektem na ekranie	• aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności • testuje grę na różnych etapach • współpracuje w grupie podczas pracy nad projektem	• bierze udział w pracach nad wypracowaniem koncepcji gry • współpracuje w grupie podczas pracy nad projektem	• programuje wybrane funkcje i elementy gry • opracowuje opis gry	• implementuje i optymalizuje kod źródłowy gry, korzystając z wypracowanych założeń	• rozbudowuje grę o nowe elementy • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera

OKRES II									
----------	--	--	--	--	--	--	--	--	--

DZIAŁ 4. PROGRAMOWANIE W JĘZYKU PYTHON

4.1. Wprowadzenie do programowania w języku Python	17.- 19. Wprowadzenie do programowania w języku Python	• definiuje pojęcia: algorytm, program, programowanie • podaje kilka sposobów przedstawienia algorytmu	• 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 Python, aby wyświetlić tekst na ekranie • omawia różnice pomiędzy kodem źródłowym a kodem wynikowym • tłumaczy, czym jest środowisko programistyczne	• wymienia przykładowe środowiska programistyczne • wyjaśnia, czym jest specyfikacja problemu • opisuje etapy rozwiązywania problemów • opisuje etapy powstawania programu komputerowego • zapisuje proste polecenia języka Python	• pisze proste programy w trybie skryptowym języka Python	• zapisuje algorytmy różnymi sposobami oraz pisze programy o większym stopniu trudności
4.2. Piszemy programy w języku Python	20.-23. Piszemy programy w języku Python	• tłumaczy, do czego używa się zmiennych w programach • pisze proste programy w trybie skryptowym języka Python z wykorzystaniem zmiennych	• wykonuje obliczenia w języku Python • omawia działanie operatorów arytmetycznych • stosuje listy w języku Python oraz operatory logiczne	• wykorzystuje instrukcję warunkową if oraz if else w programach • wykorzystuje iterację w konstruowanych algorytmach	• buduje złożone schematy blokowe służące do przedstawiania skomplikowanych algorytmów • konstruuje złożone sytuacje warunkowe	• pisze programy w języku Python do rozwiązywania zadań matematycznych • tworzy program składający się z

				- wykorzystuje w programach instrukcję iteracyjną for - definiuje funkcje w języku Python i omawia różnice między funkcjami zwracającymi wartość a funkcjami niezwracającymi wartości	(wiele warunków) w algorytmach - pisze programy zawierające instrukcje warunkowe, pętle oraz funkcje - wyjaśnia, jakie błędy zwraca interpreter - czyta kod źródłowy i opisuje jego działanie	kilku funkcji wywoływanych w programie głównym
4.3. Algorytmy na liczbach naturalnych	24. i 25. Algorytmy na liczbach naturalnych	- wyjaśnia działanie operatora modulo - wyjaśnia algorytm badania podzielności liczb	- zapisuje w postaci listy kroków algorytm badania podzielności liczb naturalnych - wykorzystuje w programach instrukcję iteracyjną while	- 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	- 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	pisze programy wykorzystujące algorytmy Euklidesa (np. obliczający NWW) oraz wyodrębniania cyfr danej liczby
4.4. Algorytmy wyszukiwania	26. i 27. Algorytmy wyszukiwania	- wyjaśnia potrzebę wyszukiwania informacji w zbiorze - określa różnice między wyszukiwaniem w zbiorach uporządkowanym i nieuporządkowanym - sprawdza działanie programów	- zapisuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym, w tym elementu największego i najmniejszego - zapisuje algorytm wyszukiwania elementu w zbiorze uporządkowanym metodą połowienia	- implementuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym - omawia funkcje zastosowane w realizacji algorytmu wyszukiwania metodą połowienia	- samodzielnie zapisuje w wybranej postaci algorytm wyszukiwania elementu w zbiorze metodą połowienia, w tym elementu największego i najmniejszego - implementuje algorytm	samodzielnie modyfikuje i optymalizuje algorytmy wyszukiwania

		wyszukujących element w zbiorze	• implementuje grę w zgadywanie liczby		wyszukiwania elementu w zbiorze metodą połowienia • implementuje algorytm wyszukiwania największej wartości w zbiorze	
4.5. Algorytmy porządkowania	28. i 29 Algorytmy porządkowania	• wyjaśnia potrzebę porządkowania danych • sprawdza działanie programu sortującego dla różnych danych	• 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	• omawia implementację algorytmu sortowania przez zliczanie • omawia funkcje zastosowane w kodzie źródłowym algorytmów sortowania przez wybieranie oraz przez zliczanie	• implementuje algorytmy porządkowania metodami przez wybieranie oraz przez zliczanie • wprowadza modyfikacje w implementacji algorytmów porządkowania przez wybieranie oraz przez zliczanie	• samodzielnie modyfikuje i optymalizuje programy sortujące metodą przez wybieranie, metodą przez zliczanie

DZIAŁ 5. GRAFIKA 3D - dodatkowy		
5.1. Tworzenie projektów	30. Tworzenie projektów	• zna podstawowe narzędzia w poznanych programach i aplikacjach graficznych • tworzy projekt za pomocą podstawowych narzędzi programu i zapisuje go w pliku (na własnym koncie) • wykorzystuje chmurę, pocztę elektroniczną oraz różne urządzenia do pracy nad projektem
5.2. Przygotowanie do druku	31. Przygotowywanie projektu do druku	• gotowy projekt i zapisuje w pliku i przygotowuje do druku • wykorzystuje chmurę, pocztę elektroniczną do przesłania/ udostępnienia projektu