

Plan wynikowy z informatyki dla klasy 8 szkoły podstawowej zgodny z programem nauczania „Lubię to!”

program Python

Wydawnictwo Nowa Era

Nauczyciel prowadzący: Tomasz Szewc

Rok szkolny: 2022/2023

Tytuł w podręczniku	Numer i temat lekcji	Ocena dopuszczająca Uczeń:	Ocena dostateczna Uczeń:	Ocena dobra Uczeń:	Ocena bardzo dobra Uczeń:	Ocena celująca Uczeń:
	1. Organizacja pracy w nowym roku szkolnym	<ul style="list-style-type: none"> wymienia zasady bezpieczeństwa obowiązujące w pracowni komputerowej stosuje poznane zasady bezpieczeństwa w pracowni oraz podczas pracy na komputerze określa, za co może uzyskać daną ocenę; wymienia możliwości poprawy oceny niedostatecznej oraz zasady pracy na zajęciach komputerowych 				
	2. Platforma Office- jej możliwości i ograniczenia.	<ul style="list-style-type: none"> potrafi zalogować się na platformie Office.com zna możliwości oferowane przez platformę Office.com obsługuje aplikację Teams (komunikuje się na chacie- pisze i odbiera wiadomości; odbiera i wysyła pliki; nawiązuje połączenia wideo) potrafi wylogować się z aplikacji Teams oraz platformy Office.com 				
DZIAŁ 1. Arkusz kalkulacyjny						
1.1. Formuły i adresowanie względne w arkuszu kalkulacyjnym	3. i 4. 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, wykorzystując adresowanie względne 	<ul style="list-style-type: none"> samodzielnie tworzy i kopiuje skomplikowane formuły obliczeniowe
1.2. Funkcje oraz adresowanie bezwzględne i mieszane w arkuszu	5. i 6. Funkcje oraz adresowanie bezwzględne i mieszane w arkuszu	<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 	<ul style="list-style-type: none"> korzysta z biblioteki funkcji, aby wyszukiwać potrzebne funkcje stosuje adresowanie 	<ul style="list-style-type: none"> stosuje zaawansowane funkcje arkusza w tabelach stworzonych na własne potrzeby

kalkulacyjnym	kalkulacyjnym			kalkulacyjnym <ul style="list-style-type: none"> ustawia format danych komórki odpowiadający jej zawartości w formułach stosuje adresowanie względne, bezwzględne i mieszane 	względne, bezwzględne lub mieszane w zaawansowanych formułach obliczeniowych	
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
DZIAŁ 2. Programowanie w języku Python						
2.1. Wprowadzenie do programowania w języku Python	11., 12. i 13. Wprowadzenie do programowania w języku Python	<ul style="list-style-type: none"> definiuje pojęcia: algorytm, program, programowanie podaje kilka sposobów przedstawienia algorytmu 	<ul style="list-style-type: none"> wymienia różne sposoby przedstawienia algorytmu: opis słowny, schemat blokowy, lista kroków poprawnie formułuje problem do rozwiązania wyjaśnia różnice między interaktywnym, a skryptowym trybem 	<ul style="list-style-type: none"> wymienia przykładowe środowiska programistyczne wyjaśnia, czym jest specyfikacja problemu opisuje etapy rozwiązywania 	<ul style="list-style-type: none"> pisze proste programy w trybie skryptowym języka Python 	<ul style="list-style-type: none"> zapisuje algorytmy różnymi sposobami oraz pisze programy o większym stopniu trudności

			<p>pracy</p> <ul style="list-style-type: none"> • stosuje odpowiednie polecenie języka Python, aby wyświetlić tekst na ekranie • omawia różnice pomiędzy kodem źródłowym, a kodem wynikowym • tłumaczy, czym jest środowisko programistyczne 	<p>problemów</p> <ul style="list-style-type: none"> • opisuje etapy powstawania programu komputerowego • zapisuje proste polecenia języka Python 		
2.2. Piszemy programy w języku Python	14., 15. i 16. Piszemy programy w języku Python	<ul style="list-style-type: none"> • tłumaczy, do czego używa się zmiennych w programach • pisze proste programy w trybie skryptowym języka Python z wykorzystaniem zmiennych 	<ul style="list-style-type: none"> • wykonuje obliczenia w języku Python • omawia działanie operatorów arytmetycznych • stosuje listy w języku Python oraz operatory logiczne 	<ul style="list-style-type: none"> • wykorzystuje instrukcję warunkową <i>if</i> oraz <i>if else</i> w programach • wykorzystuje iterację w konstruowanych algorytmach • wykorzystuje w programach instrukcję iteracyjną <i>for</i> • definiuje funkcje w języku Python i omawia różnice między funkcjami zwracającymi wartość, a funkcjami niezwracającymi wartości 	<ul style="list-style-type: none"> • 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 	<ul style="list-style-type: none"> • pisze programy w języku Python do rozwiązywanie zadań matematycznych • tworzy program składający się z kilku funkcji wywoływanych w programie głównym
2.3. Algorytmy	17., 18. i 19.	<ul style="list-style-type: none"> • wyjaśnia działanie 	<ul style="list-style-type: none"> • zapisuje w postaci listy 	<ul style="list-style-type: none"> • omawia algorytm 	<ul style="list-style-type: none"> • wyjaśnia różnice 	<ul style="list-style-type: none"> • pisze programy

na liczbach naturalnych	Algorytmy na liczbach naturalnych	<p>operatora modulo</p> <ul style="list-style-type: none"> • wyjaśnia algorytm badania podzielności liczb 	<p>kroków algorytm badania podzielności liczb naturalnych</p> <ul style="list-style-type: none"> • wykorzystuje w programach instrukcję iteracyjną <i>while</i> 	<p>Euklidesa w wersji z odejmowaniem i z dzieleniem – zapisuje go w wybranej postaci</p> <ul style="list-style-type: none"> • wyjaśnia algorytm wyodrębniania cyfr danej liczby i zapisuje go w wybranej postaci 	<p>między instrukcją iteracyjną <i>while</i>, a pętlą <i>for</i></p> <ul style="list-style-type: none"> • pisze programy obliczające NWD, stosując algorytm Euklidesa, oraz wypisujące cyfry danej liczby • wyjaśnia różnice między algorytmem Euklidesa w wersjach z odejmowaniem i z dzieleniem 	<p>wykorzystujące algorytmy Euklidesa (np. obliczający NWW) oraz wyodrębniania cyfr danej liczby</p>
2.4. Algorytmy wyszukiwania	20. i 21. Algorytmy wyszukiwania	<ul style="list-style-type: none"> • wyjaśnia potrzebę wyszukiwania informacji w zbiorze • określa różnice między wyszukiwaniem w zbiorach uporządkowanym i nieuporządkowanym • sprawdza działanie programów wyszukujących element w zbiorze 	<ul style="list-style-type: none"> • 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 	<ul style="list-style-type: none"> • implementuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym • omawia funkcje zastosowane w realizacji algorytmu wyszukiwania metodą połowienia • implementuje algorytm wyszukiwania największej 	<ul style="list-style-type: none"> • 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 	<ul style="list-style-type: none"> • samodzielnie modyfikuje i optymalizuje algorytmy wyszukiwania

				wartości w zbiorze		
2.5. Algorytmy porządkowania	22. i 23. 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 metodami przez wybieranie oraz przez zliczanie • omawia implementację algorytmu sortowania przez wybieranie • stosuje pętle zagnieżdżone i wyjaśnia, jak działają 	<ul style="list-style-type: none"> • omawia implementację algorytmu sortowania przez zliczanie • omawia funkcje zastosowane w kodzie źródłowym algorytmów sortowania przez wybieranie oraz przez zliczanie 	<ul style="list-style-type: none"> • implementuje algorytmy porządkowania metodami przez wybieranie oraz przez zliczanie • wprowadza modyfikacje w implementacji algorytmów porządkowania przez wybieranie oraz przez zliczanie 	<ul style="list-style-type: none"> • samodzielnie modyfikuje i optymalizuje programy sortujące metodą przez wybieranie, metodą przez zliczanie
• DZIAŁ 4. Projekty						
4.1. Dokumentacja szkolnej imprezy sportowej	24. i 25. 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
4.2. Sterowanie	26., 27. i 28. Sterowanie	<ul style="list-style-type: none"> • aktywnie uczestniczy 	<ul style="list-style-type: none"> • bierze udział w pracach nad wypracowaniem 	<ul style="list-style-type: none"> • programuje wybrane funkcje i 	<ul style="list-style-type: none"> • implementuje i optymalizuje kod 	<ul style="list-style-type: none"> • rozbudowuje grę o nowe elementy

obiektem na ekranie	obiektem na ekranie	<p>w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności</p> <ul style="list-style-type: none"> • testuje grę na różnych etapach • współpracuje w grupie podczas pracy nad projektem 	<p>koncepcji gry</p> <ul style="list-style-type: none"> • współpracuje w grupie podczas pracy nad projektem 	<p>elementy gry</p> <ul style="list-style-type: none"> • opracowuje opis gry 	<p>źródłowy gry, korzystając z wypracowanych założeń</p>	<ul style="list-style-type: none"> • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera
4.3. Historia i rozwój informatyki	29., 30. i 31. Historia i rozwój informatyki	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności – znalezienie informacji w internecie, umieszczenie ich w chmurze 	<ul style="list-style-type: none"> • współpracuje z innymi podczas pracy nad projektem • analizuje zebrane dane • tworzy projekt prezentacji multimedialnej 	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania • tworzy prezentację wg projektu zaakceptowanego przez zespół 	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu • analizuje i weryfikuje pod względem merytorycznym i technicznym przygotowaną prezentację 	<ul style="list-style-type: none"> • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera • wzbogaca prezentację o elementy podnoszące jej walory estetyczne i merytoryczne
4.4. Informatyka w moim przyszłym życiu	32. Informatyka w moim przyszłym życiu	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności • bierze aktywny udział w dyskusji nad wyborem atrakcyjnego zawodu 	<ul style="list-style-type: none"> • gromadzi informacje dotyczące wybranych zawodów, umieszcza je w zaprojektowanych tabelach i dokumentach tekstowych 	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu • projektuje tabele do zapisywania informacji o zawodach • weryfikuje i formatuje przygotowane dokumenty 	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu, weryfikuje opracowane treści i łączy wszystkie dokumenty w całość 	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu, przyjmuje rolę lidera • podczas dyskusji przyjmuje funkcję moderatora

		wymagającego kompetencji informatycznych		tekstowe		
--	--	--	--	----------	--	--

Plan wynikowy jest zgodny z podstawą programową dla klasy VIII szkoły podstawowej w przedmiocie informatyka.
Realizacja treści planu wynikowego może ulegać zmianom z powodu:

- wydarzeń i sytuacji wynikających z bieżącego funkcjonowania szkoły,
- okoliczności nieplanowanych.

Nauczyciel realizujący