

Przedmiotowy system oceniania z informatyki dla klasy 8

Tytuł w podręczniku	Numer i temat lekcji	Wymagania konieczne (ocena dopuszczająca) Uczeń:	Wymagania podstawowe (ocena dostateczna) Uczeń:	Wymagania rozszerzające (ocena dobra) Uczeń:	Wymagania dopełniające (ocena bardzo dobre) Uczeń:	Wymagania wykraczające (ocena celująca) Uczeń:
1. ALGORYTYKA I PROGRAMOWANIE						
1.1 Zapisywanie algorytmów na liczbach naturalnych w języku Scratch	1. Algorytmy sekwencyjne, warunkowe i iteracyjne w języku Scratch	• tworzy zmienne w języku Scratch.	• tworzy skrypty wykonujące działania matematyczne na zmiennych.	• wykorzystuje w budowanych skryptach sytuacje warunkowe • wykorzystuje powtórzenia (iteracje) w budowanych skryptach.	• tworzy skrypty w języku Scratch łączące w sobie sytuacje warunkowe i instrukcje iteracyjne.	• samodzielnie rozwiązuje problemy, wykorzystując zmienne, sytuacje warunkowe oraz instrukcje iteracyjne w języku Scratch.
	2. Realizacja algorytmu Euklidesa w wersji z odejmowaniem oraz algorytmów wykorzystujących podzielność liczb	• wyjaśnia, czym jest największy wspólny dzielnik dwóch liczb.	• omawia algorytm Euklidesa wykorzystujący odejmowanie liczb.	• przedstawia algorytm Euklidesa z odejmowaniem w postaci skryptu w języku Scratch.	• bada podzielność liczb naturalnych w języku Scratch • wyodrębnia cyfry danej liczby w języku Scratch.	• tworzy w języku Scratch skrypty przedstawiające na różne sposoby algorytm Euklidesa.
1.2. Algorytmy wyszukiwania i porządkowania	3. Wyszukiwanie największego elementu w zbiorze nieuporządkowanym	• przedstawia w postaci listy kroków algorytm wyboru większej z dwóch liczb.	• przedstawia w postaci listy kroków algorytm wyboru największej liczby ze zbioru.	• wyszukuje największą liczbę w podanym zbiorze • w języku Scratch tworzy skrypt wskazujący większą z dwóch podanych liczb.	• w języku Scratch tworzy skrypt wyszukujący największą liczbę w podanym zbiorze.	• tworzy algorytm wyszukujący najmniejszą liczbę w zbiorze i wykorzystuje go w przykładach z życia codziennego (np. wskazanie najwyższego ucznia w klasie).
	4. Metody porządkowania i wyszukiwania elementów zbioru	• przedstawia w postaci listy kroków algorytm porządkowania metodą przez wybieranie.	• porządkuje podane liczby w zbiorze nieuporządkowanym, korzystając z algorytmu porządkowania metodą przez wybieranie.	• wykorzystuje metodę wyszukiwania przez połowienie, aby odnaleźć określony element w zbiorze uporządkowanym • porządkuje podane liczby w zbiorze nieuporządkowanym przy zastosowaniu metody przez zliczanie.	• w języku Scratch tworzy prostą grę w odgadywanie liczby, wykorzystując do tego metodę wyszukiwania przez połowienie.	• tworzy algorytm porządkujący liczby według określonych kryteriów, np. oddzielnie liczby parzyste i nieparzyste.

1.3. Wprowadzenie do programowania w języku C++	5. i 6. Składnia języka i stosowanie zmiennych	w języku C++ tworzy prosty program wyświetlający tekst na ekranie.	- wskazuje różnice między kodem źródłowym a kodem wynikowym - omawia etapy tworzenia programu w języku C++.	- wprowadza zmienne do programów pisanych w języku C++ - wykonuje działania matematyczne na zmiennych w programach pisanych w języku C++.	- omawia podstawowe typy zmiennych w języku C++ - wyjaśnia działanie operatorów arytmetycznych stosowanych w języku C++.	tworzy programy komputerowe wspomagające rozwiązywanie zadań matematycznych, np. obliczające pola figur.
	7. i 8. Instrukcje warunkowe i iteracyjne w języku C++	pisze proste programy w języku C++.	- stosuje instrukcje warunkowe w programach pisanych w języku C++ - stosuje powtórzenia (iteracje) w programach pisanych w języku C++.	wyjaśnia działanie operatorów logicznych i porównania stosowanych w języku C++.	wykorzystuje instrukcje iteracyjne w języku C++ do wyszukiwania największej liczby w zbiorze.	tworzy program komputerowy sprawdzający podzielność jednej liczby przez drugą.
1.4. Stosowanie funkcji i tablic do zapisania algorytmów porządkowania i wyszukiwania w języku C++	9. Funkcje i tablice w języku C++	- tworzy procedury w języku Scratch - wyjaśnia, czym jest podprogram (funkcja, procedura) w programie komputerowym.	stosuje funkcje w języku C++, aby oddzielać od siebie logiczne bloki programu.	- wyjaśnia, jaką rolę odgrywa parametr funkcji - tworzy funkcje z wieloma parametrami.	tworzy proste programy z wykorzystaniem funkcji.	tworzy programy z zastosowaniem różnego typu funkcji.
	10. Tablice w języku C++	- wskazuje element w tablicy o wybranym indeksie - wskazuje indeks tablicy wybranego elementu - deklaruje tablice w C++ - inicjuje tablice poprzez wypisanie jej elementów w nawiasach klamrowych	- deklaruje stałą w języku C++ - omawia zasady deklarowania tablic w języku C++ - wyjaśnia sposób indeksowania w tablicach.	definiuje tablice w języku C++ i wprowadza do nich dane.	- wykonuje operacje na elementach tablicy z wykorzystaniem funkcji - deklaruje zmienne tablicowe jako zmienne globalne.	tworzy złożone programy z zastosowaniem tablic.
	11. Algorytmy porządkowania i wyszukiwania w języku C++	- testuje działanie programu sortującego dla różnych danych - testuje działanie programu wyszukiującego przez połowienie.	zapisuje w języku C++ algorytm porządkowania metodami przez wybieranie, zliczanie, połowienie.	- stosuje instrukcję <i>do... while...</i> do implementacji pętli - wymienia funkcje zastosowane w implementacji algorytmu porządkowania przez wybieranie	wykorzystuje tablice w języku C++ do realizacji algorytmów wyszukiwania i porządkowania.	wykorzystuje funkcje w języku C++ do tworzenia programów wykonujących kilka zadań, np. podstawowe działania arytmetyczne na dwóch liczbach (dodawanie,

				i w implementacji algorytmu porządkowania przez zliczanie wymienia funkcje zastosowane w realizacji algorytmu wyszukiwania przez połowienie.		odejmowanie, mnożenie, dzielenie).
1.5. Wprowadzenie do programowania w języku Python	5. i 6. Składnia języka i stosowanie zmiennych	stosuje odpowiednie polecenie języka Python, aby wyświetlić tekst na ekranie.	wyjaśnia różnice pomiędzy interaktywnym a skryptowym trybem pracy.	- wykonuje obliczenia w języku Python - omawia działanie operatorów arytmetycznych w języku Python.	pisze prosty program w trybie skryptowym języka Python	pisze program w języku Python wykorzystujący zmienne i służący do wykonywania podstawowych działań matematycznych.
	7. i 8. Instrukcje warunkowe i iteracyjne w języku Python	pisze proste programy w trybie skryptowym języka Python.	wykorzystuje zmienne w programach pisanych w języku Python.	- wykorzystuje instrukcje iteracyjne w programach pisanych w języku Python - wykorzystuje instrukcje warunkowe w programach pisanych w języku Python.	w języku Python pisze program realizujący algorytm wyszukiwania największej liczby w zbiorze.	pisze programy w języku Python wspomagające rozwiązywanie zadań matematycznych.
1.6. Stosowanie funkcji i list do zapisywania algorytmów porządkowania i wyszukiwania w języku Python	9. funkcje w języku Python	wykorzystuje procedury w języku Scratch do tworzenia prostych kompozycji	definiuje funkcje w języku Python i wyjaśnia ich działanie.	omawia różnice pomiędzy funkcjami zwracającymi wartość a funkcjami niezwracającymi wartości.	tworzy funkcję zwracającą wartość największej liczby z podanego zbioru.	tworzy program składający się z kilku funkcji wywoływanych w programie głównym w zależności od potrzeby.
	10. Listy w języku Python	tworzy listy w języku Python i wprowadza do nich dane.	wyświetla zawartość listy na ekranie.	pisze funkcję pozwalającą na wprowadzanie danych do listy.	wykorzystuje listy w języku Python do realizacji algorytmów wyszukiwania i porządkowania.	tworzy programy wspomagające rozwiązywanie zadań matematycznych i wykorzystujące funkcje i listy w języku Python.
	11. Algorytmy porządkowania i wyszukiwania w języku Python	- testuje działanie programu sortującego dla różnych danych - testuje działanie programu wyszukującego przez połowienie.	- zapisuje w języku Python algorytm porządkowania metodami: przez wybieranie, przez zliczanie, połowienie - omawia ogólną postać pętli iteracyjnej <i>while</i>.	- stosuje instrukcję <i>while</i> do implementacji pętli - wymienia funkcje zastosowane w implementacji algorytmów: porządkowania przez wybieranie,	- zagnieżdża pętle <i>for</i> - wyjaśnia różnice między instrukcją iteracyjną <i>while</i> a pętlą <i>for</i> - omawia funkcje zastosowane w implementacji algorytmów:	- samodzielnie modyfikuje programy sortujące metodą przez wybieranie, metodą przez zliczanie - samodzielnie modyfikuje program wyszukujący metodą przez połowienie.

				porządkowania przez zliczanie wymienia funkcje zastosowane w realizacji algorytmu wyszukiwania przez połowienie.	porządkowania przez wybieranie, porządkowania przez zliczanie omawia funkcje zastosowane w realizacji algorytmu wyszukiwania przez połowienie.	
2. OBLICZENIA w ARKUSZU KALKULACYJNYM						
2.1. Komórka, adres, formuła	12. Podstawy pracy w arkuszu kalkulacyjnym	wprowadza dane różnego rodzaju do komórek arkusza kalkulacyjnego	- omawia zastosowania arkusza kalkulacyjnego - omawia budowę arkusza kalkulacyjnego	- wyjaśnia, do czego służy formuła obliczeniowa - tworzy proste formuły obliczeniowe	kopiuje utworzone formuły obliczeniowe pomiędzy komórkami tabeli, wykorzystując adresowanie względne.	samodzielnie tworzy skomplikowane formuły obliczeniowe i kopiuje je pomiędzy komórkami tabeli.
2.2. Projektowanie tabeli i stosowanie funkcji arkusza kalkulacyjnego	13. Zastosowanie podstawowych funkcji i formatowanie komórek w arkuszu kalkulacyjnym	- wprowadza różnego rodzaju dane do komórek arkusza kalkulacyjnego - formatuje zawartość komórek (wyrównanie tekstu oraz wygląd czcionki).	- tłumaczy zasady wprowadzania danych do komórek arkusza kalkulacyjnego - dodaje i usuwa wiersze oraz kolumny tabeli arkusza kalkulacyjnego.	- stosuje formułę SUMA do dodawania do siebie wartości wpisanych do wielu komórek - stosuje formułę ŚREDNIA, aby obliczyć średnią arytmetyczną z kilku liczb - ustawia format danych komórki odpowiadający jej zawartości.	- korzysta z biblioteki funkcji, aby wyszukiwać potrzebne formuły - używa sytuacji warunkowych w arkuszu kalkulacyjnym, korzystając z funkcji JEŻELI.	wykorzystuje arkusz kalkulacyjny w rozwiązywaniu problemów życia codziennego (np. obliczania średniej swoich ocen i przedstawienia jej zmian na wykresie).
2.3. Arkusz kalkulacyjny, czyli kalkulacje	14. Adresowanie bezwzględne i formatowanie komórek w arkuszu kalkulacyjnym	wprowadza dane do arkusza kalkulacyjnego.	stosuje formułę SUMA do dodawania do siebie zawartości komórek.	- kopiuje formułę pomiędzy komórkami, stosując adresowanie bezwzględne - stosuje opcję Zawijanie tekstu dla dłuższych tekstów wpisywanych do komórek.	wyjaśnia, w jaki sposób arkusz kalkulacyjny zaokrągla duże liczby do ich postaci wykładniczej (naukowej).	wykorzystuje arkusz kalkulacyjny do prowadzenia osobistego budżetu lub planowania kosztów jakiegoś wydarzenia.
2.4. Więcej o pracy w arkuszu kalkulacyjnym	15. Adresowanie mieszane, bramowanie i drukowanie tabeli	wprowadza dane do komórek arkusza kalkulacyjnego.	- stosuje obramowania dla komórek arkusza kalkulacyjnego i formatuje je według potrzeby - drukuję tabelę arkusza kalkulacyjnego.	kopiuje formuły pomiędzy komórkami z wykorzystaniem adresowania mieszanego.	w zależności od potrzeby stosuje adresowanie względne, bezwzględne lub mieszane, tworząc formuły obliczeniowe.	stosuje zaawansowane funkcje arkusza w tabelach tworzonych na własne potrzeby.

2.5. Przedstawianie danych w postaci wykresu	16. Projektowanie i tworzenie wykresów w arkuszu kalkulacyjnym	wstawia wykres do arkusza kalkulacyjnego.	omawia poszczególne elementy wykresu.	dobiera odpowiedni wykres do danych, które ma przedstawiać.	tworzy wykres dla więcej niż jednej serii danych.	modyfikuje w sposób estetyczny i kreatywny wygląd wykresu, dobierając jego elementy składowe, kolory i zastosowane czcionki.
2.6. Wstawianie tabel i wykresów arkusza kalkulacyjnego do dokumentów tekstowych	17. Wstawianie tabel i wykresów do dokumentu tekstowego	kopiuje tabelę lub wykres arkusza kalkulacyjnego od schowka i wkleja ją w dokumencie tekstowym.	odróżnia wstawianie tabeli lub wykresu arkusza kalkulacyjnego do dokumentu tekstowego jako obiektu osadzonego i jako obiektu połączanego.	wstawia tabelę lub wykres arkusza kalkulacyjnego do dokumentu tekstowego jako obiekt osadzony albo jako obiekt połączony, w zależności od potrzeb.	wykorzystuje opcję Obiekt do wstawiania tabeli arkusza kalkulacyjnego do dokumentu tekstowego.	przygotowuje dokumenty (sprawozdania, raporty, referaty), wykorzystując wklejanie tabel i wykresów arkusza kalkulacyjnego do dokumentów tekstowych.
2.7. Zastosowanie arkusza kalkulacyjnego	18. Zastosowanie arkusza kalkulacyjnego – algorytmy	wprowadza dane różnego rodzaju do komórek arkusza kalkulacyjnego.	formatuje tabelę arkusza kalkulacyjnego.	wykorzystuje funkcję JEŻELI do tworzenia algorytmów z warunkami w arkuszu kalkulacyjnym.	kopiuje formuły pomiędzy komórkami, aby zastosować algorytm iteracji.	przedstawia dowolny algorytm z warunkami lub iteracyjny w postaci tabeli.
	19. Zastosowanie arkusza kalkulacyjnego – nauki przyrodnicze	wprowadza dane różnego rodzaju do komórek arkusza kalkulacyjnego.	formatuje tabelę arkusza kalkulacyjnego.	- tworzy tabelę do wpisywania wyników pomiarów doświadczeń - tworzy formuły obliczeniowe dla wprowadzonych danych, wykorzystując wzory fizyczne.	przedstawia wyniki swoich obliczeń na wykresach różnego typu.	korzysta z arkusza kalkulacyjnego do analizowania doświadczeń z fizyki lub chemii.
	20. Zastosowanie arkusza kalkulacyjnego – symulacja modelu	wprowadza dane różnego rodzaju do komórek arkusza kalkulacyjnego.	formatuje tabelę arkusza kalkulacyjnego.	wykorzystuje funkcje losującą, aby symulować rzuty sześcienną kostką do gry.	- wykorzystuje formułę LICZBA.CAŁK, aby zamieniać ułamki dziesiętne na liczby całkowite - używa funkcji LICZ.JEŻELI aby sumować liczbę powtórzeń rzutów kostką.	przygotowuje w arkuszu kalkulacyjnym tabele do prowadzenia różnego rodzaju gier losowych.
	21. Zastosowanie arkusza kalkulacyjnego – operacje bazodanowe	stosuje arkusz kalkulacyjny do porządkowania danych.	wykorzystuje arkusz kalkulacyjny do prostego filtrowania danych.	omawia zasady przygotowania tabeli do filtrowania danych.	przedstawia działania potrzebne do porządkowania różnych danych.	opracowuje zbiór kryteriów niezbędnych do wyświetlania danych.

4.1. Prezentacje multimedialne i filmy	28. Cechy dobrej prezentacji multimedialnej	• dodaje nowe slajdy do prezentacji multimedialnej • dodaje teksty i obrazy do slajdów.	• zmienia wygląd prezentacji, ustalając jej podstawowe kolory.	• dodaje do prezentacji animacje i przejścia.	• umieszcza w prezentacji filmy i dźwięk.	• wykorzystując wiele rozmaitych elementów multimedialnych, wykonuje atrakcyjną oraz poprawną merytorycznie prezentację multimedialną.
	29. Montaż filmów wideo	• dodaje do prezentacji multimedialnej klip wideo dostępny na dysku komputera.	• przycina fragmenty filmu wideo.	• dodaje do filmu teksty i obrazy • dodaje do filmu efektowne przejścia.	• umieszcza w prezentacji multimedialnej własne nagrania wideo i dźwiękowe.	• wykorzystując wiele rozmaitych elementów multimedialnych, wykonuje atrakcyjną oraz poprawną merytorycznie prezentację multimedialną.
4.2. Historia i rozwój informatyki – projekt	30. Historia i rozwój informatyki	• tworzy prezentację multimedialną.	• współpracuje z innymi podczas tworzenia prezentacji multimedialnej • wyszukuje w internecie materiały do prezentacji • wykorzystuje chmurę do dzielenia się materiałami.	• rozdziela pomiędzy członków grupy zadania niezbędne do wykonania projektu.	• krytycznie ocenia wartość informacji znalezionych w internecie – weryfikuje je korzystając z różnych źródeł.	• podczas pracy nad projektem wykazuje się wysokim poziomem estetyki i kreatywności.