

Wymagania edukacyjne na poszczególne oceny – Informatyka na czasie, klasa 2, 2023/2024

Dział	Temat działu	Liczba godzin	Osiągnięcia uczniów (wymagania): uczeń				
			konieczne (ocena dopuszczająca)	podstawowe (ocena dostateczna)	rozszerzające (ocena dobra)	dopelniające (ocena bardzo dobra)	wykraczające (ocena celująca)
1	Jak pobierać dane do arkusza kalkulacyjnego	2	<ul style="list-style-type: none"> - wyjaśnia, dlaczego warto stosować narzędzia wymiany danych - wymienia podstawowe zastosowania arkusza kalkulacyjnego - wyjaśnia podstawowe pojęcia związane z arkuszem kalkulacyjnym: skoroszyt, arkusz, adres komórki, formuła, funkcja, zakres adresów - sposoby, również między arkuszami - kopiuje formuły - stosuje funkcje: SUMA, ŚREDNIA, MAX, MIN, DŁ, JEŻELI - przedstawia dane w postaci wykresów 	<ul style="list-style-type: none"> - pobiera dane z różnych źródeł i przetwarza je - modyfikuje dane w arkuszu - wykorzystuje adresy komórek w formułach obliczeniowych - wyjaśnia różnice między formułami i funkcjami - korzysta z wbudowanych funkcji arkusza kalkulacyjnego - stosuje różne sposoby zaznaczania zakresów komórek - kopiuje dane z komórek i wkleja je na różne 	<ul style="list-style-type: none"> - wyszukuje samodzielnie w Internecie dane potrzebne do realizacji określonych zadań - importuje do arkusza dane z różnych źródeł, w tym ze stron WWW - poprawnie stosuje adresowanie względne, bezwzględne i mieszane 	<ul style="list-style-type: none"> - modyfikuje dane podczas ich importowania - dobiera typ wykresu do rodzaju danych - interpretuje otrzymane wyniki zgodnie z ustalonymi założeniami 	<ul style="list-style-type: none"> - wyszukuje w Internecie informacje na temat nowych funkcji i stosuje je w zadaniach - buduje złożone formuły pozwalające wykonywać obliczenia, rozwiązujące określone problemy
	2	Wyciągamy wiedzę z danych	4	<ul style="list-style-type: none"> - omawia różnicę między filtrowaniem i sortowaniem danych - filtruje i sortuje dane - tworzy tabele i stosuje w nich sortowanie i filtrowanie danych - opisuje możliwości tabel przestawnych - tworzy tabele przestawne 	<ul style="list-style-type: none"> - stosuje formuły arkusza kalkulacyjnego do losowego generowania zbiorów danych - stosuje funkcję INDEKS do zwracania wartości określonych komórek - aktualizuje tabelę przestawną po modyfikacji danych źródłowych 	<ul style="list-style-type: none"> - generuje zestawy danych za pomocą narzędzi online - tworzy fragmentatory i korzysta z osi czasu - interpretuje wyniki tabel i wykresów przestawnych 	<ul style="list-style-type: none"> - modyfikuje style tabel przestawnych - buduje tabele przestawne dla dużych zbiorów danych

		<ul style="list-style-type: none"> - filtruje dane w tabeli przestawnej - tworzy wykresy przestawne 	<ul style="list-style-type: none"> - stosuje gotowe style tabel przestawnych - podsumowuje dane w tabeli przestawnej na różne sposoby - stosuje różne sposoby wyświetlania wartości w tabeli przestawnej - grupuje i rozgrupowuje daty w tabelach przestawnych 				
3	Korespondencja seryjna	2	<ul style="list-style-type: none"> - tworzy dokument główny korespondencji seryjnej - umieszcza pola korespondencji seryjnej w tworzonych dokumentach - poprawnie scala dokumenty seryjne 	<ul style="list-style-type: none"> - tworzy bazę adresatów - stosuje reguły warunkowe do personalizacji listów seryjnych 	<ul style="list-style-type: none"> - drukuje listy seryjne - wysyła korespondencję seryjną za pomocą poczty elektronicznej 	<ul style="list-style-type: none"> - tworzy zestawy dokumentów seryjnych (listy, etykiety, koperty) 	<ul style="list-style-type: none"> - zarządza danymi adresatów korespondencji seryjnej w arkuszu kalkulacyjnym
4	Relacyjne bazy danych	5	<ul style="list-style-type: none"> - wyjaśnia podstawowe pojęcia związane z bazami danych: tabela, atrybut, rekord, pole, klucz główny, klucz obcy, relacja - wymienia różne zastosowania baz danych - tworzy tabele i definiuje relacje między nimi - tworzy raporty z użyciem kreatora 	<ul style="list-style-type: none"> - projektuje nieduże bazy danych - zarządza danymi w bazie danych w programie MS Access - modyfikuje dane zawarte w bazie danych 	<ul style="list-style-type: none"> - drukuje i eksportuje raporty do pliku 	<ul style="list-style-type: none"> - dostosowuje raport do aktualnych potrzeb 	<ul style="list-style-type: none"> - tworzy kwerendy, formularze oraz raporty w programie MS Access

P1	Współdziałanie aplikacji – projekt zespołowy	4	<ul style="list-style-type: none"> - współpracuje w grupie, korzystając z narzędzi online - korzysta z programów graficznych podczas pracy nad zadaniem projektowym 	<ul style="list-style-type: none"> - stosuje funkcje arkusza kalkulacyjnego do przetwarzania danych - korzysta z zasobów internetowych, wyszukując potrzebne informacje - stosuje zasady netykiety i korzysta z niej w komunikacji zdalnej 	<ul style="list-style-type: none"> - przyjmuje rolę lidera odpowiedzialnego za zespół i projekt 	<ul style="list-style-type: none"> - testuje rozwiązania wypracowane w grupie 	<ul style="list-style-type: none"> - prezentuje efekty pracy grupowej na forum klasy
5	Od problemu do programu	4	<ul style="list-style-type: none"> - wyjaśnia pojęcie algorytmu - podaje przykłady algorytmów niekomputerowych - wymienia cechy poprawnego algorytmu - wyjaśnia na przykładzie pojęcie specyfikacji problemu - wyjaśnia pojęcia związane z algorytmiką i programowaniem: schemat blokowy, lista kroków, kod źródłowy, kod wynikowy, kompilator, interpreter, słowa kluczowe, funkcje, plik wykonywalny - wyjaśnia pojęcie zmiennej i typu zmiennej - wymienia zasady tworzenia kodu źródłowego w wybranym języku programowania 	<ul style="list-style-type: none"> - tworzy algorytm wyznaczania pierwiastka kwadratowego - zapisuje algorytm Herona w postaci listy kroków - zapisuje algorytm w postaci kodu źródłowego - kompiluje zapisany kod źródłowy - tworzy program sprawdzający warunek trójkąta - stosuje podstawowe konstrukcje wybranego języka programowania: instrukcje wejścia i wyjścia, operatory arytmetyczne i logiczne oraz instrukcję warunkową 	<ul style="list-style-type: none"> - tworzy samodzielnie programy, wykorzystując poznane instrukcje wybranego języka programowania - instrukcje warunkowe 	<ul style="list-style-type: none"> - znajduje i poprawia błędy w kodzie źródłowym programu - stosuje w swoich programach zagnieżdżone 	<ul style="list-style-type: none"> - pisze programy rozwiązujące zadania matematyczne i fizyczne oraz problemy z napisami

6	Wyszukujemy i sumujemy	1	<ul style="list-style-type: none"> - wyjaśnia, na czym polega iteracyjne rozwiązywanie problemu - stosuje w swoich programach podstawowe rodzaje pętli: while i for - opisuje zasady użycia pętli w programach - 	<ul style="list-style-type: none"> - analizuje w tabeli działanie algorytmu krok po kroku - opisuje, jak komputer porównuje dwie wartości - tworzy program wyszukiwania największej liczby z zadanego ciągu liczb 	<ul style="list-style-type: none"> - stosuje różne rodzaje pętli, dostosowując wybór do rozwiązywanego problemu - używa w algorytmach rozwiązania z wartownikiem, czyli specjalną wartością wskazującą koniec ciągu wczytywanych do programu wartości 	<ul style="list-style-type: none"> - stosuje w kodzie operatory inkrementacji i dekrementacji 	<ul style="list-style-type: none"> - analizuje kod źródłowy i poprawia ewentualne błędy
7	Binarny system liczbowy	3	<ul style="list-style-type: none"> - zapisuje liczby w binarnym systemie liczbowym oraz w systemie szesnastkowym - wyjaśnia pojęcia związane z systemami liczbowymi: system pozycyjny, podstawa systemu liczbowego 	<ul style="list-style-type: none"> - zapisuje algorytm konwersji między systemami liczbowymi w postaci programu komputerowego - omawia algorytm konwersji liczb między systemami dziesiętnym i binarnym 	<ul style="list-style-type: none"> - stosuje operację dzielenia całkowitego w rozwiązywaniu problemów 	<ul style="list-style-type: none"> - tworzy algorytmy konwersji między różnymi systemami liczbowymi 	<ul style="list-style-type: none"> - programuje algorytmy konwersji między różnymi systemami liczbowymi, stosując strukturę tablicy lub listy
8	Czy ta liczba jest pierwsza?	3	<ul style="list-style-type: none"> - definiuje liczby złożone i liczby pierwsze - podaje przykłady użycia liczb pierwszych - omawia algorytmy sprawdzające podzielność liczb 	<ul style="list-style-type: none"> - bada podzielność wybranych liczb, programując poznane algorytmy w wybranym języku - grupuje instrukcje w funkcje i wyjaśnia cel stosowania funkcji 	<ul style="list-style-type: none"> - wyjaśnia pojęcia liczb doskonałych, bliźniaczych, zaprzyjaźnionych - wyjaśnia praktyczne znaczenie liczb pierwszych w informatyce 	<ul style="list-style-type: none"> - tworzy samodzielnie programy dla poznanych algorytmów 	<ul style="list-style-type: none"> - pisze programy wykorzystujące poznane rodzaje liczb pierwszych
9	Algorytm Euklidesa i działania na ułamkach	3	<ul style="list-style-type: none"> - wyjaśnia pojęcia: NWD, NWW - podaje przykłady zastosowania algorytmu Euklidesa - zapisuje algorytm Euklidesa w postaci listy kroków 	<ul style="list-style-type: none"> - tworzy program pozwalający na dodawanie ułamków - stosuje odpowiednie konstrukcje wybranego języka programowania do implementacji omawianych zagadnień 	<ul style="list-style-type: none"> - tworzy programy realizujące działania na ułamkach - dzieleniem 	<ul style="list-style-type: none"> - opisuje algorytm Euklidesa i tworzy realizujący go program w wybranym języku programowania 	<ul style="list-style-type: none"> - poznaje inne zastosowania algorytmu Euklidesa, wykorzystując informacje zawarte w Internecie lub innych źródłach

			(w tym: funkcję, która nie zwraca wartości)		- opisuje różnicę w sprawności dwóch wersji algorytmu Euklidesa: z odejmowaniem i z	
Razem	31					