

Kryteria i wymagania na poszczególne oceny - informatyka kl. VIII

1. Ogólne zasady oceniania uczniów

1. Ocenianie osiągnięć edukacyjnych ucznia polega na rozpoznawaniu przez nauczyciela postępów w opanowaniu przez ucznia wiadomości i umiejętności. Nauczyciel powinien analizować i oceniać poziom wiedzy i umiejętności ucznia w stosunku do wymagań edukacyjnych wynikających z podstawy programowej i realizowanych w szkole programów nauczania (opracowanych zgodnie z podstawą programową danego przedmiotu).
2. Nauczyciel ma za zadanie:
 - informować ucznia o poziomie jego osiągnięć edukacyjnych oraz o postępach w tym zakresie,
 - pomagać uczniowi w samodzielnym planowaniu jego rozwoju,
 - motywować ucznia do dalszych postępów w nauce,
 - informować rodziców (opiekunów prawnych) o postępach, trudnościach w nauce oraz specjalnych uzdolnieniach ucznia.
3. Oceny są jawne dla ucznia i jego rodziców (opiekunów prawnych).
4. Na wniosek ucznia lub jego rodziców (opiekunów prawnych) sprawdzone i ocenione pisemne prace kontrolne są udostępniane do wglądu uczniowi lub jego rodzicom (opiekunom prawnym).

2. Kryteria oceniania poszczególnych form aktywności

Ocenię podlegają: sprawdziany, kartkówki, ćwiczenia praktyczne, odpowiedzi ustne, praca na lekcji, prace dodatkowe oraz szczególne osiągnięcia.

Kartkówki są przeprowadzane w formie pisemnej, a ich celem jest sprawdzenie wiadomości i umiejętności ucznia z zakresu programowego ostatnich jednostek lekcyjnych (maksymalnie trzech).

- Nauczyciel nie ma obowiązku uprzedzania uczniów o terminie i zakresie programowym kartkówki.
- Kartkówka powinna być tak skonstruowana /zrealizowane trzy ostatnie tematy lekcji/, aby uczeń mógł wykonać wszystkie polecenia w czasie nie dłuższym niż 15 minut.
- Możliwość poprawy każdej oceny w terminie ustalonym z nauczycielem.

Odpowiedź ustna obejmuje zakres programowy aktualnie realizowanego działu. Oceniając ją, nauczyciel bierze pod uwagę:

- zgodność wypowiedzi z postawionym pytaniem,
- właściwe posługiwanie się pojęciami,

Aktywność i praca ucznia na lekcji są oceniane za pomocą plusów i minusów lub oceny.

- Plus uczeń może uzyskać m.in. za samodzielne wykonanie krótkiej pracy na lekcji, krótką poprawną odpowiedź ustną, aktywną pracę w grupie, pomoc koleżeńską na lekcji przy rozwiązywaniu problemu, przygotowanie do lekcji.
- Minus uczeń może uzyskać m.in. za nieprzygotowanie do lekcji (np. brak podręcznika, zeszytu, plików potrzebnych do wykonania zadania), brak zaangażowania na lekcji.
- Sposób przeliczania plusów i minusów na oceny jest umowny między nauczycielem a uczniami; i zgodny z ocenianiem wewnątrzszkolnym zawartym w statucie szkoły.

Prace dodatkowe obejmują dodatkowe zadania dla zainteresowanych uczniów, prace projektowe wykonane indywidualnie lub zespołowo, wykonanie pomocy naukowych, prezentacji. Oceniając ten rodzaj pracy, nauczyciel bierze pod uwagę m.in.:

- wartość merytoryczną pracy,
- stopień zaangażowania w wykonanie pracy,
- estetykę wykonania,
- wkład pracy ucznia,
- sposób prezentacji,
- oryginalność i pomysłowość pracy.

Szczególne osiągnięcia uczniów, w tym udział w konkursach przedmiotowych (szkolnych i międzyszkolnych), są oceniane zgodnie – z regulaminami konkursów oraz ocenianiem wewnątrzszkolnym (Statut Szkoły Podstawowej w Pustelniku, rozdział 8).

Wymagania stanowią podstawę do określenia poziomu opanowania przez uczniów podstawy programowej i służą do ustalania ocen klasyfikacyjnych z poszczególnych zajęć edukacyjnych i ich uzasadniania. Są one dostosowywane do indywidualnych potrzeb rozwojowych oraz możliwości uczniów.

3. Zasady uzupełniania braków i poprawiania ocen

1. Każdą ocenę można poprawić.
2. Nauczyciel informuje ucznia o otrzymanej ocenie z ostatniej pracy bezpośrednio po jej wystawieniu.
3. Rodzice (opiekunowie prawni) mogą uzyskać szczegółowe informacje o wynikach i postępach w pracy ucznia podczas indywidualnych kontaktów z nauczycielem /według harmonogramu spotkań przyjętych przez szkołę/.

4. Uczeń ma obowiązek uzupełnić braki w wiedzy i umiejętnościach np. nieobecność, biorąc udział w zajęciach wyrównawczych lub drogą indywidualnych konsultacji z nauczycielem.

5. Sposób poprawiania klasyfikacyjnej oceny półrocznej lub rocznej reguluje Statut Szkoły Podstawowej w Pustelniku, rozdział 8.

Wymagania edukacyjne zostały dostosowane do możliwości psychofizycznych uczniów.

Wymagania programowe na poszczególne oceny 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 dobra) Uczeń:	Wymagania wykraczające (ocena celująca) Uczeń:
WYMAGANIA NA OCENĘ ŚRÓDROCZNĄ						
Dział 1. Edytor tekstu MS Word – przypomnienie i utrwalenie wiadomości z kl. VII						
Formatowanie tekstu	Formatowanie tekstu – utrwalenie zasad formatowania tekstu.	<ul style="list-style-type: none"> omawia zastosowanie oraz budowę edytora tekstu Word wymienia podstawowe zasady formatowania tekstu 	<ul style="list-style-type: none"> redaguje przygotowane dokumenty tekstowe, przestrzegając odpowiednich zasad dostosowuje formę tekstu do jego przeznaczenia korzysta z tabulatora do ustawiania tekstu w kolumnach ustawia wcięcia w dokumencie tekstowym, wykorzystując suwaki na linijce 	<ul style="list-style-type: none"> wykorzystuje kapitaliki i wersaliki do przedstawienia różnych elementów dokumentu tekstowego ustawia różne rodzaje tabulatorów, wykorzystując selektor tabulatorów sprawdza liczbę wyrazów, znaków, wierszy i akapitów w dokumencie tekstowym za pomocą <u>Statystyki wyrazów</u> 	<ul style="list-style-type: none"> kopiuje formatowanie pomiędzy fragmentami tekstu, korzystając z <u>Malarza formatów</u> sprawdza poprawność ortograficzną tekstu za pomocą słownika ortograficznego wyszukuje wyrazy bliskoznaczne, korzystając ze słownika synonimów zamienia określone wyrazy w całym dokumencie tekstowym, korzystając z opcji <u>Znajdź i zamień</u> 	<ul style="list-style-type: none"> przygotowuje estetyczne projekty dokumentów tekstowych do wykorzystania w życiu codziennym, takie jak: zaproszenia na uroczystości, ogłoszenia, podania, listy z pełnym sformatowaniem.
Tworzenie i formatowanie tabel	Tabela – tworzenie, formatowanie.	<ul style="list-style-type: none"> wstawia obrazy do tabeli z opisem 	<ul style="list-style-type: none"> zmienia położenie obrazu względem 	<ul style="list-style-type: none"> zmienia kolejność elementów 	<ul style="list-style-type: none"> osadza obraz w tabeli 	<ul style="list-style-type: none"> wstawia do tabeli inne, poza obrazami, obiekty osadzone,

		<ul style="list-style-type: none"> • wstawia tabele do dokumentu tekstowego 	<p>komórki tabeli</p> <ul style="list-style-type: none"> • formatuje tabele w dokumencie tekstowym • wstawia symbole do dokumentu tekstowego 	<p>graficznych w tabeli</p> <ul style="list-style-type: none"> • wstawia grafiki SmartArt do dokumentu tekstowego z tabelą 	<ul style="list-style-type: none"> • wstawia zrzut ekranu do dokumentu tekstowego • rozdziela tekst pomiędzy kilka pól tekstowych, tworząc łącza między nimi 	np. arkusz kalkulacyjny.
Edytor równań	Piszemy równania.	<ul style="list-style-type: none"> • omawia zastosowanie oraz budowę edytora równań • wymienia podstawowe operacje w edytorze 	<ul style="list-style-type: none"> • zmienia położenie narzędzi do pisania działań obliczeniowych • wstawia symbole do równania w dokumencie tekstowym 	<ul style="list-style-type: none"> • stosuje różnorodność obliczeniową • wykorzystuje dodatkowe symbole do obliczeń 	<ul style="list-style-type: none"> • swobodnie tworzy równania z wykorzystaniem dostępnych symboli 	<ul style="list-style-type: none"> • wstawia do równania inne symbole obliczeniowe poza podstawowymi.
DZIAŁ 2. 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, 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 kalkulacyjnym	3. i 4. 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 	<ul style="list-style-type: none"> • korzysta z biblioteki funkcji, aby wyszukiwać potrzebne funkcje • stosuje adresowanie względne, bezwzględne lub mieszane w zaawansowanych formułach obliczeniowych 	<ul style="list-style-type: none"> • stosuje zaawansowane funkcje arkusza w tabelach stworzonych na własne potrzeby

				adresowanie względne, bezwzględne i mieszane		
1.3. Przedstawianie danych na wykresie	5. i 6. 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	7. 8. 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
WYMAGANIA EDUKACYJNE NA OCENĘ KONCOWOROCZNAROCZNĄ Z UWZGLĘDNIENIEM ŚRÓDROCZNEJ						
DZIAŁ 3. Programowanie w języku Python						
2.1. Wprowadzenie do programowania w języku Python	9., 10. i 11. 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, 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 	<ul style="list-style-type: none"> wymienia przykładowe środowiska programistyczne wyjaśnia, czym jest specyfikacja problemu opisuje etapy rozwiązywania problemów opisuje etapy powstawania programu komputerowego zapisuje proste polecenia języka Python 	<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
2.2. Piszemy programy w języku Python	12., 13. i 14. Piszemy programy w języku Python	<ul style="list-style-type: none"> tłumaczy, do czego używa się zmiennych w programach pisze proste programy w 	<ul style="list-style-type: none"> wykonuje obliczenia w języku Python omawia działanie operatorów 	<ul style="list-style-type: none"> wykorzystuje instrukcję warunkową <code>if</code> oraz <code>if else</code> w programach wykorzystuje iterację 	<ul style="list-style-type: none"> konstruuje złożone sytuacje warunkowe (wiele warunków) w algorytmach 	<ul style="list-style-type: none"> pisze programy w języku Python do rozwiązywanie zadań matematycznych tworzy program składający

		trybie skryptowym języka Python z wykorzystaniem zmiennych	arytmetycznych <ul style="list-style-type: none"> • stosuje listy w języku Python oraz operatory logiczne 	w konstruowanych algorytmach <ul style="list-style-type: none"> • wykorzystuje w programach instrukcję iteracyjną <code>for</code> • definiuje funkcje w języku Python 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 	się z kilku funkcji wywoływanych w programie głównym
2.3. Algorytmy na liczbach naturalnych	15., 16. i 17. Algorytmy na liczbach naturalnych	<ul style="list-style-type: none"> • wyjaśnia działanie operatora modulo • wyjaśnia algorytm badania podzielności liczb 	<ul style="list-style-type: none"> • zapisuje w postaci listy kroków algorytm badania podzielności liczb naturalnych • wykorzystuje w programach instrukcję iteracyjną <code>while</code> 	<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, oraz wypisujące cyfry danej liczby 	<ul style="list-style-type: none"> • pisze programy wykorzystujące algorytmy Euklidesa (np. obliczający NWW) oraz wyodrębniania cyfr danej liczby
2.4. Algorytmy wyszukiwania	18. i 19. Algorytmy wyszukiwania	<ul style="list-style-type: none"> • wyjaśnia potrzebę wyszukiwania informacji w zbiorze • 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 	<ul style="list-style-type: none"> • implementuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym 	<ul style="list-style-type: none"> • samodzielnie zapisuje w wybranej postaci algorytm wyszukiwania elementu w zbiorze 	<ul style="list-style-type: none"> • samodzielnie modyfikuje i optymalizuje algorytmy wyszukiwania
2.5. Algorytmy porządkowania	20. i 21. 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 algorytmu sortowania przez wybieranie • stosuje pętle zagnieżdżone i wyjaśnia, jak działają 	<ul style="list-style-type: none"> • omawia funkcje zastosowane w kodzie źródłowym algorytmu sortowania przez wybieranie 	<ul style="list-style-type: none"> • implementuje algorytm porządkowania metodą przez wybieranie • wprowadza modyfikacje w implementacji algorytmu porządkowania przez wybieranie 	<ul style="list-style-type: none"> • samodzielnie modyfikuje i optymalizuje programy sortujące metodą przez wybieranie

• DZIAŁ 4. Projekty						
4.1. Dokumentacja szkolnej imprezy sportowej	22. i 23. 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 obiektem na ekranie	24., 25. i 26. Sterowanie obiektem na ekranie	<ul style="list-style-type: none"> • aktywnie uczestniczy w pracach zespołu, realizuje powierzone zadania o niewielkim stopniu trudności • testuje grę na różnych etapach • współpracuje w grupie podczas pracy nad projektem 	<ul style="list-style-type: none"> • bierze udział w pracach nad wypracowaniem koncepcji gry • współpracuje w grupie podczas pracy nad projektem 	<ul style="list-style-type: none"> • programuje wybrane funkcje i elementy gry • opracowuje opis gry 	<ul style="list-style-type: none"> • implementuje i optymalizuje kod źródłowy gry, korzystając z wypracowanych założeń 	<ul style="list-style-type: none"> • rozbudowuje grę o nowe elementy • współpracuje w grupie podczas pracy nad projektem, przyjmuje funkcję lidera

Po zrealizowaniu podstawy programowej będą przeprowadzone zajęcia lekcyjne /w miarę dostępnego czasu/ z zakresu **projektowania przedmiotów 3D oraz obsługa programu i drukarki 3D.**