

Szkoła Podstawowa Specjalna nr 9 w Krakowie

Wymagania edukacyjne z informatyki w klasie 8 dla uczniów z niepełnosprawnością intelektualną

1. Ogólne zasady oceniania uczniów

- 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).
- 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.
- Oceny są jawne dla ucznia i jego rodziców (opiekunów prawnych).
- Na wniosek ucznia lub jego rodziców (opiekunów prawnych) nauczyciel uzasadnia ocenę w sposób określony w statucie szkoły.

Szczegółowe warunki i sposób oceniania wewnątrzszkolnego określa statut szkoły.

2. Kryteria oceniania poszczególnych form aktywności

Ocenie podlegają: ćwiczenia praktyczne, odpowiedzi ustne, aktywność i praca na lekcji, prace dodatkowe oraz szczególne osiągnięcia.

3. Wymagania edukacyjne na poszczególne oceny:

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:
DZIAŁ 1. Arkusz kalkulacyjny						
1.1. Formuły i adresowanie względne w arkuszu kalkulacyjnym	1. i 2. Formuły i adresowanie względne w arkuszu kalkulacyjnym	• omawia zastosowanie oraz budowę arkusza kalkulacyjnego • określa adres komórki • wprowadza dane różnego rodzaju do komórek arkusza kalkulacyjnego	• określa zasady wprowadzania danych do komórek arkusza kalkulacyjnego • dodaje i usuwa wiersze oraz kolumny w tabeli	• tworzy proste formuły obliczeniowe	• kopiuje utworzone formuły obliczeniowe	• samodzielnie tworzy 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	• rozumie różnice między adresowaniem względnym, bezwzględnym i mieszanym	• stosuje w arkuszu podstawowe funkcje: (SUMA, ŚREDNIA), wpisuje je ręcznie oraz korzysta z kreatora	• ustawia format danych komórki odpowiadający jej zawartości • w formułach stosuje adresowanie względne, bezwzględne i mieszane	• korzysta z biblioteki funkcji, aby wyszukiwać potrzebne funkcje • stosuje adresowanie względne, bezwzględne lub mieszane w zaawansowanych formułach obliczeniowych	• stosuje zaawansowane funkcje arkusza w tabelach tworzonych na własne potrzeby
1.3. Przedstawianie danych na wykresie	5. i 6. Przedstawianie danych na wykresie	• wstawia wykres do arkusza kalkulacyjnego	• modyfikuje poszczególne elementy wykresu	• dobiera odpowiedni wykres do rodzaju danych	• tworzy wykres dla więcej niż jednej serii danych	• tworzy wykresy dla dwóch serii danych
1.4. Zastosowania arkusza kalkulacyjnego	7. 8. Zastosowania arkusza kalkulacyjnego	• korzysta z arkusza kalkulacyjnego w celu stworzenia kalkulacji wydatków	• zapisuje w tabeli arkusza kalkulacyjnego dane otrzymane z prostych doświadczeń i przedstawia je na wykresie	• sortuje oraz filtruje dane w arkuszu kalkulacyjnym	• stosuje filtry niestandardowe	• 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	9., 10. i 11. Wprowadzenie do programowania w języku Python	• definiuje pojęcia: algorytm, program, programowanie • podaje kilka sposobów	• wymienia różne sposoby przedstawienia algorytmu: opis słowny, lista kroków	• wymienia przykładowe środowiska programistyczne • wyjaśnia, czym jest	• pisze proste programy w trybie skryptowym języka Python	• zapisuje algorytmy różnymi sposobami

		przedstawienia algorytmu	• poprawnie formułuje problem do rozwiązania	specyfikacja problemu • opisuje etapy		
--	--	-----------------------------	---	--	--	--

			• 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	rozwiązywania problemów • opisuje etapy powstawania programu komputerowego • zapisuje proste polecenia języka Python		
2.2. Piszemy programy w języku Python	12., 13. i 14. 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	• wykorzystuje instrukcję warunkową <code>if</code> oraz <code>else</code> w programach • wykorzystuje iterację w konstruowanych algorytmach • wykorzystuje w programach instrukcję iteracyjną <code>for</code>	• konstruuje złożone sytuacje warunkowe (wiele warunków) w algorytmach • pisze programy zawierające instrukcje warunkowe, pętle oraz funkcje • czyta kod Źródłowy i opisuje jego działanie	• pisze programy w języku Python do rozwiązywanie zadań matematycznych
2.3. Algorytmy na liczbach naturalnych	15., 16. i 17. 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 <code>while</code>	• omawia algorytm Euklidesa i zapisuje go w wybranej postaci	• 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	• pisze programy wykorzystujące algorytmy Euklidesa (np. obliczający NWW) oraz wyodrębniania cyfr danej liczby
2.4. Algorytmy wyszukiwania	18. i 19. Algorytmy wyszukiwania	• wyjaśnia potrzebę wyszukiwania informacji w zbiorze • sprawdza działanie programów wyszukujących element w zbiorze	• zapisuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym, w tym elementu największego i najmniejszego	• implementuje algorytm wyszukiwania elementu w zbiorze nieuporządkowanym	• samodzielnie zapisuje w wybranej postaci algorytm wyszukiwania elementu w zbiorze	• samodzielnie modyfikuje algorytmy wyszukiwania

2.5. Algorytmy porządkowania	20. i 21. Algorytmy porządkowania	• wyjaśnia potrzebę porządkowania danych • sprawdza działanie programu sortującego dla różnych danych	• zapisuje w dowolnej formie algorytm porządkowania metodą przez wybieranie • omawia implementację algorytmu sortowania przez wybieranie	• omawia funkcje zastosowane w kodzie Źródłowym algorytmu sortowania przez wybieranie	• implementuje algorytm porządkowania metodą przez wybieranie	• samodzielnie modyfikuje programy sortujące metodą przez wybieranie
• DZIAŁ 4. Projekty						
4.1. Dokumentacja szkolnej imprezy sportowej	22. i 23. Dokumentacja szkolnej imprezy sportowej	• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, wykonując powierzone mu zadania o niewielkim stopniu trudności	• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej • wprowadza dane do zaprojektowanych tabel	• przygotowuje dokumentację imprezy, wykonuje obliczenia, projektuje tabele oraz wykresy • współpracuje w grupie podczas pracy nad projektem	• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, przygotowuje zestawienia, drukuje wyniki • współpracuje w grupie podczas pracy nad projektem	• bierze udział w przygotowaniu dokumentacji szkolnej imprezy sportowej, tworzy zestawienia zawierające zaawansowane formuły, wykresy oraz elementy graficzne • czynnie współpracuje w grupie podczas pracy nad projektem
4.2. Sterowanie obiektem na ekranie	24., 25. i 26. 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	• współpracuje podczas programowania wybranych funkcji i elementów gry	• wraz z innymi implementuje i optymalizuje kod Źródłowy gry, korzystając z wypracowanych założeń	• rozbudowuje grę o nowe elementy • czynnie współpracuje w grupie podczas pracy nad projektem

4. Dostosowania edukacyjne z informatyki dla uczniów o specjalnych potrzebach edukacyjnych

Uczniowie z orzeczeniem z poradni psychologiczno – pedagogicznej o potrzebie kształcenia specjalnego ze względu na niepełnosprawność intelektualną w stopniu lekkim realizują podstawę programową przewidzianą dla kształcenia ogólnego. Nauczyciel dostosowuje pracę ucznia poprzez:

- Uczeń pracuje podczas lekcji w miarę swoich możliwości
- Uczeń ma wydłużony czas pracy, mniejszą liczbę zadań do wykonania oraz niższy stopień trudności
- Uczeń zajmuje stanowisko pracy blisko nauczyciela w celu lepszego kontaktu
- Nauczyciel nadzoruje samodzielną pracę ucznia
- Nauczyciel pomaga w rozwiązywaniu zdań poprzez zadawanie naprowadzających pytań, ewentualnie uczeń pracuje w grupie z kolegami
- Ocenie podlega także zaangażowanie do nauki oraz aktywność na zajęciach

Termin poprawy ocen może zostać wydłużony

Od ucznia wymaga się podstawowych umiejętności i wiadomości, o których mowa w podstawie programowej.

W przypadku, kiedy uczeń sprostą owym wymaganiom oraz zdobywa bardzo dobre wyniki, wówczas wymagania rozszerza się w celu wyrównania jego wiadomości z pozostałymi uczniami.