

Wymagania edukacyjne i sposoby sprawdzania edukacyjnych osiągnięć uczniów z informatyki - klasy II zakres rozszerzony

I. Cele kształcenia – wymagania ogólne

1. Bezpieczne posługiwanie się komputerem i jego oprogramowaniem, wykorzystanie sieci komputerowej; komunikowanie się za pomocą komputera i technologii informacyjno-komunikacyjnych.
2. Wyszukiwanie, gromadzenie i przetwarzanie informacji z różnych źródeł; opracowywanie za pomocą komputera: rysunków, tekstów, danych liczbowych, motywów, animacji, prezentacji multimedialnych.
3. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, z zastosowaniem podejścia algorytmicznego.
4. Wykorzystanie komputera oraz programów i gier edukacyjnych do poszerzania wiedzy i umiejętności z różnych dziedzin oraz do rozwijania zainteresowań.
5. Ocena zagrożeń i ograniczeń, docenianie społecznych aspektów rozwoju i zastosowań informatyki

II. Cele kształcenia – wymagania szczegółowe

1. Posługiwanie się komputerem i jego oprogramowaniem, korzystanie z sieci komputerowej. Uczeń:

- 1) przedstawia sposoby Reprezentowania różnych form informacji w komputerze: liczb, znaków, obrazów, animacji, dźwięków;
- 2) wyjaśnia funkcje systemu operacyjnego i korzysta z nich; opisuje różne systemy operacyjne;
- 3) przedstawia warstwowy model sieci komputerowych, określa ustawienia sieciowe danego komputera i jego lokalizacji, prawidłowo posługuje się terminologią sieciową, opisuje zasady architektury klient-serwer, korzysta z usług w sieci,
- 4) zapoznaje się z możliwościami nowych urządzeń związanych z technologiami informacyjno - komunikacyjnymi, poznaje nowe programy i systemy oprogramowania.

2. Komunikowanie się za pomocą komputera i technologii informacyjno komunikacyjnych. Uczeń:

1) wykorzystuje zasoby i usługi sieci komputerowych w komunikacji z innymi użytkownikami, w tym do przesyłania i udostępniania danych; 2) bierze udział w dyskusjach w sieci (forum internetowe, czat).

3. Opracowywanie informacji za pomocą komputera, w tym: rysunków, tekstów, danych liczbowych, animacji, prezentacji multimedialnych i filmów. Uczeń:

- 1) opisuje podstawowe modele barw i ich zastosowanie;
- 2) określa własności grafiki rastrowej i wektorowej oraz charakteryzuje podstawowe formaty plików graficznych, tworzy i edytuje obrazy rastrowe i wektorowe z uwzględnieniem warstw i przekształceń;
- 3) przetwarza obrazy i filmy, np.: zmienia rozdzielczość, rozmiar, model barw, stosuje filtry;
- 4) wykorzystuje arkusz kalkulacyjny do obrazowania zależności funkcyjnych i do zapisywania algorytmów.

4. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:

- 1) analizuje, modeluje i rozwiązuje sytuacje problemowe z różnych dziedzin;
- 2) stosuje podejście algorytmiczne do rozwiązywania problemu;
- 3) formułuje przykłady sytuacji problemowych, których rozwiązanie wymaga podejścia algorytmicznego i użycia komputera;
- 4) dobiera efektywny algorytm do rozwiązania sytuacji problemowej i zapisuje go w wybranej notacji;
- 5) posługuje się podstawowymi technikami algorytmicznymi;
- 6) ocenia własności rozwiązania algorytmicznego (komputerowego), np. zgodność ze specyfikacją, efektywność działania;
- 7) opracowuje i przeprowadza wszystkie etapy prowadzące do otrzymania poprawnego rozwiązania problemu: od sformułowania specyfikacji problemu po testowanie rozwiązania;
- 8) posługuje się metodą „dziel i zwyciężaj” w rozwiązywaniu problemów;
- 9) stosuje rekurencję w prostych sytuacjach problemowych; 10) stosuje podejście zachłanne w rozwiązywaniu problemów; 11) opisuje podstawowe algorytmy i stosuje: a) algorytmy na liczbach całkowitych, np.:

– reprezentacja liczb w dowolnym systemie pozycyjnym, w tym w dwójkowym i szesnastkowym,

– sprawdzanie, czy liczba jest liczbą pierwszą, doskonałą,

– rozkładanie liczby na czynniki pierwsze,

– iteracyjna i rekurencyjna realizacja algorytmu Euklidesa,

– iteracyjne i rekurencyjne obliczanie wartości liczb Fibonacciego,

– wydawanie reszty metodą zachłanną,

b) algorytmy wyszukiwania i porządkowania (sortowania), np.:

– jednoczesne znajdowanie największego i najmniejszego elementu w zbiorze:
algorytm naiwny i optymalny,

– algorytmy sortowania ciągu liczb: bąbelkowy, przez wybór, przez wstawianie liniowe lub binarne, przez scalanie, szybki, kubelkowy, c) algorytmy numeryczne, np.:

– obliczanie wartości pierwiastka kwadratowego,

– obliczanie wartości wielomianu za pomocą schematu Hornera,

– zastosowania schematu Hornera: reprezentacja liczb w różnych systemach liczbowych, szybkie podnoszenie do potęgi,

– wyznaczanie miejsc zerowych funkcji metodą połowienia,

– obliczanie pola obszarów

zamkniętych, d) algorytmy na tekstach,

np.:

– sprawdzanie, czy dany ciąg znaków tworzy palindrom, anagram,

– porządkowanie alfabetyczne,

– wyszukiwanie wzorca w tekście,

– obliczanie wartości wyrażenia podanego w postaci odwrotnej notacji polskiej, e)

algorytmy kompresji i szyfrowania, np.:

– kody znaków o zmiennej długości, np. alfabet Morse'a, kod Huffmana,

– szyfr Cezara,

– szyfr przestawieniowy,

- szyfr z kluczem jawnym (RSA),
 - wykorzystanie algorytmów szyfrowania, np. w podpisie elektronicznym,
 - f) algorytmy badające własności geometryczne, np.:
 - sprawdzanie warunku trójkąta,
 - badanie położenia punktów względem prostej,
 - badanie przynależności punktu do odcinka,
 - przecinanie się odcinków,
 - przynależność punktu do obszaru,
 - konstrukcje rekurencyjne: drzewo binarne, dywan Sierpińskiego, płatek Kocha;
- 12) projektuje rozwiązanie problemu (realizację algorytmu) i dobiera odpowiednią strukturę danych;
 - 13) stosuje metodę zstępującą i wstępującą przy rozwiązywaniu problemu;
 - 14) dobiera odpowiednie struktury danych do realizacji algorytmu, w tym struktury dynamiczne;
 - 15) stosuje zasady programowania strukturalnego i modularnego do rozwiązywania problemu;
 - 16) opisuje własności algorytmów na podstawie ich analizy;
 - 17) ocenia zgodność algorytmu ze specyfikacją problemu;
 - 18) oblicza liczbę operacji wykonywanych przez algorytm;
 - 19) szacuje wielkość pamięci potrzebnej do komputerowej realizacji algorytmu;
 - 20) bada efektywność komputerowych rozwiązań problemów;
 - 21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;
 - 22) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu i uruchamianiu programów;
 - 23) stosuje podstawowe konstrukcje programistyczne w wybranym języku programowania, instrukcje iteracyjne i warunkowe, rekurencję, funkcje i procedury, instrukcje wejścia i wyjścia, poprawnie tworzy strukturę programu;
 - 24) dobiera najlepszy algorytm, odpowiednie struktury danych i oprogramowanie do rozwiązania postawionego problemu;
 - 25) dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;

- 26) ocenia poprawność komputerowego rozwiązania problemu na podstawie jego testowania;
- 27) wyjaśnia źródło błędów w obliczeniach komputerowych (błąd względny, błąd bezwzględny);
- 28) realizuje indywidualnie lub zespołowo projekt programistyczny z wydzieleniem jego modułów, w ramach pracy zespołowej, dokumentuje pracę zespołu.

6. Uczeń wykorzystuje komputer oraz programy i gry edukacyjne do poszerzania wiedzy i umiejętności z różnych dziedzin:

- 1) opracowuje indywidualne i zespołowe projekty przedmiotowe i międzyprzedmiotowe z wykorzystaniem metod i narzędzi informatyki;
- 2) korzysta z zasobów edukacyjnych udostępnianych na portalach przeznaczonych do kształcenia na odległość.

7. Uczeń wykorzystuje komputer i technologie informacyjno-komunikacyjne do rozwijania swoich zainteresowań, opisuje zastosowania informatyki, ocenia zagrożenia i ograniczenia, docenia aspekty społeczne rozwoju i zastosowań informatyki:

- 1) opisuje najważniejsze elementy procesu rozwoju informatyki i technologii informacyjno-komunikacyjnych;
- 2) wyjaśnia szanse i zagrożenia dla rozwoju społecznego i gospodarczego oraz dla obywateli, związane z rozwojem informatyki i technologii informacyjno-komunikacyjnych; 3) stosuje normy etyczne i prawne związane z rozpowszechnianiem programów komputerowych, bezpieczeństwem i ochroną danych oraz informacji w komputerze i w sieciach komputerowych;
- 4) omawia zagadnienia przestępczości komputerowej, w tym piractwo komputerowe, nielegalne transakcje w sieci;
- 5) przygotowuje się do świadomego wyboru kierunku i zakresu dalszego kształcenia informatycznego.

Wymagania edukacyjne niezbędne do uzyskania poszczególnych ocen

1. **Ocenę dopuszczającą** otrzymuje uczeń, który słabo radzi sobie z pracą przy komputerze, w jego poczynaniach widać braki w zakresie wiedzy i umiejętności określonych w minimum programowym, ale nie przekreślają one możliwości dalszego zdobywania wiedzy.
2. **Ocenę dostateczną** otrzymuje uczeń wykazujący braki w umiejętnościach i wiedzy zawartej w podstawie programowej, nadrabia jednak pracowitością i chęcią wykonania ćwiczeń. Uczeń wykonuje swoją pracę poprawnie pod względem użycia funkcji

programów, ale mało estetycznie i z błędami. Projekt pozostaje niewykończony. Uczeń stosuje jedynie podstawowe funkcje używanego oprogramowania.

3. **Ocenę dobrą** otrzymuje uczeń, który samodzielnie wykonuje ćwiczenia, którego prace zawierają drobne błędy, lecz są wykonane estetycznie. Uczeń wykazuje znajomość poznawanych programów komputerowych i ich średnio zaawansowanych funkcji. Stosuje klasyczne rozwiązania, wzorowane na istniejących projektach.
4. **Ocenę bardzo dobrą** otrzymuje uczeń biegle posługujący się oprogramowaniem i urządzeniami peryferyjnymi, dobrze dobierający materiał do projektów, umiejący zaproponować kilka alternatywnych rozwiązań, wykonujący projekt bezbłędnie i estetycznie.
5. **Ocenę celującą** otrzymuje uczeń, który w czasie ćwiczenia stosuje zaawansowane funkcje programu i sprzętu, w pełni opanował materiał zawarty w podstawie programowej i wykorzystuje go w zadaniach w sposób kreatywny, niestandardowy. Wykonuje powierzone zadania i projekty na wysokim poziomie, estetycznym, dobrze skomponowanym. Uczeń umie także zaproponować własne, oryginalne pomysły, a jego projekty są funkcjonalne i wykończone.