

# INFORMATYKA DLA PROGRAMISTY WYMAGANIA EDUKACYJNE

(rok szkolny 2024/2025 )

Przy ustalaniu oceny z zajęć edukacyjnych stosuje się kryteria ujęte w Statucie IV Liceum Ogólnokształcącego im. Tadeusza Kotarbińskiego w Gorzowie Wielkopolskim

## ROZDZIAŁ X ODDZIAŁ V § 84.

I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

### Treści nauczania – wymagania szczegółowe

#### I. Rozumienie, analizowanie i rozwiązywanie problemów.

1) planuje kolejne kroki rozwiązywania problemu, z uwzględnieniem podstawowych etapów myślenia komputacyjnego (określenie problemu, definicja modeli i pojęć, znalezienie rozwiązania, zaprogramowanie i testowanie rozwiązania);

2) stosuje przy rozwiązywaniu problemów z różnych dziedzin algorytmy poznane w szkole podstawowej oraz algorytmy:

a) na liczbach: badania pierwszości liczby, zamiany reprezentacji liczb między pozycyjnymi systemami liczbowymi, działań na ułamkach z wykorzystaniem NWD i NWW,

b) na tekstach: porównywanie tekstów, wyszukiwania wzorca w tekście metodą naiwną, szyfrowania tekstu metodą Cezara,

c) porządkowania ciągu liczb przez wstawianie i metodą bąbelkową,

d) obliczania wartości elementów ciągu metodą iteracyjną w tym wartości elementów ciągu Fibonacciego; 3) sprawdza poprawność działania algorytmów dla przykładowych danych.

e) wyróżnia w problemie podproblemy i charakteryzuje: metodę połowienia, stosuje podejście zachłanne i rekurencję;

f) porównuje działanie różnych algorytmów dla wybranego problemu, analizuje algorytmy na podstawie ich gotowych implementacji;

g) w zależności od problemu rozwiązuje go, stosując metodę wstępującą lub zstępującą;

h) do realizacji rozwiązania problemu dobiera odpowiednią metodę lub technikę algorytmiczną i struktury danych;

i) objaśnia dobrany algorytm, uzasadnia poprawność rozwiązania na wybranych przykładach danych i ocenia jego efektywność;

j) ilustruje i wyjaśnia rolę pojęć, obiektów i operacji matematycznych w projektowaniu rozwiązań problemów informatycznych i z innych dziedzin, posługuje się pojęciem logarytmu;

k) przedstawia sposoby reprezentowania w komputerze znaków, liczb, wartości logicznych;

l) objaśnia sposoby wykonywania przez komputer działań arytmetycznych i operacji logicznych;

m) wyjaśnia, jakie może być źródło błędów pojawiających się w obliczeniach komputerowych: błąd zaokrąglenia, błąd przybliżenia;

n) dyskutuje na temat roli myślenia komputacyjnego i jego metod, takich jak: abstrakcja, reprezentacja danych, dekompozycja problemu, redukcja, myślenie rekurencyjne, podejście heurystyczne w rozwiązywaniu problemów z różnych dziedzin.

#### II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych.

1) projektuje i programuje rozwiązania problemów z różnych dziedzin, stosując: instrukcje wejścia / wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne, funkcje z parametrami i bez parametrów, testuje poprawność programów dla różnych danych, w szczególności programuje algorytmy z punktu I.2);

- 2) projektuje i tworzy programy w procesie rozwiązywania problemów, wykorzystuje w programach dobrane do algorytmów struktury danych, w tym struktury dynamiczne i korzysta z dostępnych bibliotek dla tych struktur;
- 3) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu, uruchamianiu i testowaniu programów;
- 4) zapisuje za pomocą listy kroków lub pseudokodu i implementuje w wybranym języku programowania algorytmy poznane na wcześniejszych etapach oraz algorytmy:
  - a) algorytm Euklidesa w wersji iteracyjnej i rekurencyjnej wraz z zastosowaniami,
  - b) znajdowania określonego elementu w zbiorze uporządkowanym metodą binarnego wyszukiwania,
  - c) generowania liczb pierwszych metodą sita Eratostenesa,
  - d) jednoczesnego wyszukiwania elementu najmniejszego i największego,
  - e) sortowania ciągu liczb przez scalanie,
  - f) wyznaczania miejsc zerowych funkcji metodą połowienia,
  - g) obliczania przybliżonej wartości pierwiastka kwadratowego,
  - h) obliczania wartości wielomianu za pomocą schematu Hornera,
  - i) szybkiego potęgowania liczb w wersji iteracyjnej i rekurencyjnej,
- 5) wykorzystuje znane sobie algorytmy przy rozwiązywaniu i programowaniu rozwiązań następujących problemów:
  - a) rozkładania liczby na czynniki pierwsze,
  - b) wykonywania działań na liczbach w systemach innych niż dziesiętny,
  - c) znajdowania w ciągu podciągów o różnorodnych własnościach, np. najdłuższego spójnego podciągu niemalejącego, spójnego podciągu o największej sumie,
  - d) zamiany wyrażenia na postać w odwrotnej notacji polskiej i obliczanie jego wartości na podstawie tej postaci;
- 6) objaśnia oraz porównuje podstawowe metody i techniki algorytmiczne oraz struktury danych, wykorzystując przykłady problemów i algorytmów, w szczególności:
  - a) wyszukiwanie elementów liniowe i przez połowienie (do znajdowania elementów w zbiorze, sortowania przez wstawianie, przybliżonego rozwiązywania równań),
  - b) rekurencję (do generowania ciągów liczb, potęgowania, sortowania liczb, generowania fraktali),
  - c) metodę dziel i zwyciężaj (jednoczesne znajdowanie minimum i maksimum, sortowanie przez scalanie i szybkie),
  - d) podejście zachłanne (do wydawania reszty, szukania najkrótszej drogi),
  - e) programowanie dynamiczne (do szukania najdłuższego wspólnego podciągu),
  - f) metodę szyfrowania z kluczem publicznym i jej zastosowanie w podpisie elektronicznym,
  - g) struktury dynamiczne: stos, kolejka, lista (do realizacji algorytmu ONP),