

Egzamin (2023) - Wstęp do programowania

- Przykładowy Zestaw 1

Zadanie 1: 7 pkt. Zadanie 2: 8 pkt. Zadanie 3: 9 pkt. Zadanie 4: 8 pkt. Zadanie 5: 9 pkt. Zadanie 6: 9p pkt.

Punktacja: 46-50 pkt - bdb(5,0); 41-45 pkt - db+(4,5); 36-40 pkt - db(4,0); 31-35 pkt - dst+(3,5); 26-30 pkt - dst(3,0); 0-25 pkt - ndst (2,0).

Zad.1. W folderze Debug1 na pendrive znajduje się projekt z kodem w języku C. W pliku main.cpp w niektórych liniach są komentarze. Twoim zadaniem jest wpisanie wartości odpowiednich zmiennych po wykonaniu konkretnej linii kodu.

Zad.2. Napisz program, który wczyta od użytkownika ciąg liczb całkowitych i wypisze sumę liczb parzystych oraz sumę liczb nieparzystych.

Zad.3. Napisz rekurencyjną funkcję, której argumentem jest liczba całkowita n ($n > 0$). Funkcja ma zwrócić wartość funkcji zadanej wzorem: $f(x) = f(x-1) + 2 \cdot f(x-2)$, gdzie $f(0) = 0$, $f(1) = 1$. Stwórz przypadek testowy.

Zad.4. Napisz funkcję, która przyjmuje jako argument tablicę liczb wymiernych oraz jej rozmiar i zwraca największą ujemną liczbę w tej tablicy. W przypadku braku liczb ujemnych w tablicy, funkcja ma zwrócić zero. Stwórz przypadek testowy.

Zad.5. Napisz program, w którym wczytasz od użytkownika dodatnią liczbę całkowitą n , a następnie kolejno n liczb całkowitych tworzących ciąg a_1, a_2, \dots, a_n . Następnie stwórz wektor zawierający tylko te liczby z ciągu, które są podzielne przez 3. W wektorze nie powinny się znaleźć liczby podzielne przez 2.

Zad.6. Napisz program, który wczytuje ze standardowego wejścia nieujemną liczbę całkowitą n i wypisuje na standardowym wyjściu następującą wynik działania:

$$\lfloor \sqrt{0} \rfloor + \lfloor \sqrt{1} \rfloor + \dots \lfloor \sqrt{n} \rfloor$$

W rozwiązaniu nie korzystać z żadnych wbudowanych funkcji matematycznych (choć w razie potrzeby nieobowiązkowo możesz napisać samodzielnie własne funkcje).

Wskazówka: $\lfloor a \rfloor$ oznacza część całkowitą a - największą liczbę całkowitą nieprzekraczającą danej liczby.