

# Egzamin (termin zerowy 2023) - Programowanie strukturalne - Zestaw P31

*Zadanie 1: 6 pkt. Zadanie 2: 12 pkt. Zadanie 3: 14 pkt. Zadanie 4: 18 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).*

Rozwiązania mają być umieszczone zgodnie ze specyfikacją:

- Zadania powinny być umieszczone w archiwum .zip na udostępnionym pendrive.
- Nazwa archiwum powinna być wg schematu NUMERZESTAWU\_NUMERALBUMU.zip gdzie numer zestawu znajduje się na górze kartki z poleceniami. np. A23\_123456.zip
- We wnętrzu archiwum powinny znajdować się tylko same kody w języku C, pliki powinny posiadać dokładnie nazwy (z uwzględnieniem wielkości znaków): `zad1.c`, `zad2.c`, `zad3.c`, `zad4.c`.
- Maksymalna waga archiwum 10 MB.
- Archiwum powinno być bez hasła.
- W przypadku pominięcia danego zadania, należy dodać plik o nazwie sprecyzowanej wyżej (zawartość może być pusta).
- Kod zakomentowany nie będzie sprawdzany.

*Za zachowanie specyfikacji dokładnie otrzymuję się dodatkowe 2 punkty. Zadania znacznie odbiegające od specyfikacji mogą nie być sprawdzane.*

Polecenia są na odwrocie.

Zad.1. W folderze DebugXYZ (XYZ - losowowe znaki) znajduje się projekt z kodem w języku C. W pliku main.c w niektórych liniijkach są komentarze. Twoim zadaniem jest wpisanie wartości odpowiednich zmiennych po wykonaniu konkretnej linii kodu. Dopisanie nowych linijek czy zaburzenie struktury kodu oznacza zero punktów za polecenie. W przypadku znaków, należy zapisać sam znak w apostrofach np. 'c' (wielkość znaków ma znaczenie).

Zad.2. Stwórz funkcję, która przyjmuje jako argument napis. Jeżeli napis zawiera cyfrę parzystą, funkcja powinna zwrócić długość napisu powiększoną o liczbę parzystych cyfr w napisie. W przypadku braku cyfr parzystych, funkcja powinna zwrócić normalną długość napisu. Stwórz przypadek testowy.

Przykład: dla napisu “kod2468” funkcja powinna zwrócić 11, dla napisu “abc13” powinna zwrócić 5, dla napisu “2468” powinna zwrócić 8, dla napisu “wmii” powinna zwrócić 4.

Zad.3. Napisz funkcję, której argumentem jest dwuwymiarowa kwadratowa tablica tablic (zawierająca elementy typu `int`) i jej wymiar  $n, n > 0$ . Funkcja ma sumę indeksów najmniejszego elementu w tablicy. W przypadku kilku najmniejszych elementów, ma być to najmniejsza możliwa suma. Stwórz przypadek testowy.

Zad.4. Napisz funkcję, która otrzymuje jako argument listę bez głowy o elementach typu:

```
struct node {  
    int i;  
    struct node * next;  
};
```

Funkcja ma wyświetlić na konsoli w kolejnych wierszach wartości elementów na liście będących kwadratami liczb całkowitych. Stwórz przypadek testowy.

Przykład. Jeśli lista składa się z elementów 4,5,6,-34,0,25,11, to ma być wyświetlone w kolejnych wierszach: 4,0,25.