

# Programowanie strukturalne - Kolokwium 2 - Zestaw B71

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`, `zad5.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).

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

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

*Punktacja: 6 pkt.*

2. Napisz funkcję, której argumentem są dwa napisy. Funkcja powinna zwrócić informację ile znaków będących cyframi jest w krótszym napisie. Jeśli napisy są równej długości, to funkcja powinna zwrócić liczbę znaków cyfr z pierwszego napisu. Stwórz przypadek testowy. W zadaniu nie korzystaj z funkcji bibliotecznych poza instrukcjami wejścia/wyjścia.

*Punktacja: 10 pkt.*

3. Napisz funkcję, której argumentem jest dwuwymiarowa kwadratowa tablica tablic (zawierająca zmienne typu `int`) oraz jej wymiar  $n$ ,  $n > 1$ . Funkcja ma zwrócić iloczyn elementów znajdujących się na przekątnej pobocznej (suma indeksów wiersza i kolumny jest równa  $n - 1$ ). Stwórz przypadek testowy.

*Punktacja: 10 pkt.*

4. Napisz strukturę `Laptop` z polami `model` (tablica znaków długości 30) oraz `pojemnoscDysku` (typu `int`). Następnie napisz funkcję `rozszerzPojemnosc`, której argumentem jest wskaźnik do struktury typu `Laptop`. Funkcja ma dodać 500 do pola `pojemnoscDysku` w przekazanym argumencie. Stwórz przypadek testowy.

*Punktacja: 12 pkt.*

5. Napisz funkcję, która przyjmuje jako argument listę bez głowy o elementach typu:

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

Funkcja ma wyświetlić w kolejnych wierszach te elementy listy, które są większe od wartości pierwszego elementu. W przypadku gdy lista ma mniej niż dwa elementy, to funkcja ma nie wyświetlać. Stwórz przypadek testowy.

*Punktacja: 22 pkt.*