

# Egzamin (termin zerowy 2023) - Programowanie strukturalne - Zestaw P14

*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 - 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).

Zad.2. Napisz funkcję `sum_of_squares`, która ma dwa argumenty. Pierwszym argumentem jest wskaźnik `num1` na stałą wartość typu `int`, a drugim argumentem jest stały wskaźnik `num2` na zmienną typu `int`. Funkcja `sum_of_squares` ma zwrócić liczbę całkowitą zawierającą sumę kwadratów wartości wskazywanej przez pierwszy i drugi wskaźnik. Stwórz przypadek testowy dla funkcji.

Zad.3. Napisz strukturę `Samochod` z polami `marka` (tablica znaków długości 5) oraz `przebieg` (typu `int`). Następnie napisz dwie funkcje i wywołaj każdą z nich co najmniej jeden raz:

- a) `initSamochod` - funkcja przyjmuje dwa argumenty: markę i przebieg, i zwraca wskaźnik nowo-utworzoną strukturę ustawiającą składowe z przekazanych argumentów. Dodatkowo funkcja powinna sprawdzić, aby marka była napisem długości co najmniej 3 i przebieg był większy niż 1000. W przypadku nie spełnienia jednego z warunków, funkcja powinna zwracać `NULL`.
- b) `dodajKilometry` - funkcja, której argumentem jest wskaźnik do struktury typu `Samochod`. Funkcja ma dodać 1000 do przebiegu w przekazanym argumencie.

Upewnij się, że drugą funkcję możesz wywołać w `main`.

Zad.4. Napisz funkcję, która przyjmuje jako argument listę z głową o elementach typu:

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

Funkcja ma zwrócić sumę elementów nieparzystych z listy. Stwórz jeden przypadek testowy.