

Egzamin (pierwszy termin 2023) - Programowanie strukturalne - Zestaw E34

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. Dane są następujące wyrazy i znaki:

```
void int int suma tab a a [ ] ( ) , [ ]
```

Ułóż je we właściwej kolejności (zachowując podaną krotność), aby otrzymać nagłówek funkcji `suma`, która dostaje jako argumenty kolejno tablicę elementów i liczbę całkowitą. Następnie dodaj dowolną implementację funkcji i stwórz dla niej przypadek testowy.

Zad.2. Napisz funkcję `double_value`, która przyjmuje jako argument wskaźnik na zmienną typu `int`. Funkcja powinna podwoić wartość zmiennej wskazywanej przez wskaźnik i zaktualizować wartość wskazywaną przez wskaźnik. Stwórz przypadek testowy w `main` tak, aby wyświetlić na konsoli wartość zaktualizowanej zmiennej.

Zad.3. Napisz strukturę `Samochod` z polami `marka` (tablica znaków długości 50) 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 na nowo utworzoną strukturę, ustawiając składowe z przekazanych argumentów. Dodatkowo funkcja powinna sprawdzić, aby marka była napisem długości co najmniej 2 i przebieg był większy niż 0. W przypadku nie spełnienia jednego z warunków, funkcja powinna zwracać `NULL`.
- b) `zwiększPrzebieg` - 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 otrzymuje jako argument listę bez głowy o elementach typu:

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

Funkcja ma wyzerować elementy nieparzyste na liście. Stwórz przypadek testowy.