

Egzamin (termin zerowy 2023) - Programowanie strukturalne - Zestaw P15

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. Napisz bezargumentową funkcję `init_block`, która rezerwuje blok czterech zmiennych typu `float`. Funkcja ma ustawić kolejno w pamięci wartości 0.5, 1.5, 2.5 i 3.5. Na koniec funkcja powinna zwrócić wskaźnik na początkową zmienną z bloku. Stwórz przypadek testowy w `main` tak, aby wyświetlić na konsoli wartości zmiennych przechowywanych na bloku stworzonym wewnątrz funkcji.

Zad.3. Napisz strukturę `Komputer` z polami `model` (tablica znaków długości 50) oraz `moc` (typu `int`). Następnie napisz dwie funkcje i wywołaj każdą z nich co najmniej jeden raz:

- a) `initKomputer` - funkcja przyjmuje dwa argumenty: `model` i `moc`, i zwraca wskaźnik nowo-utworzoną strukturę ustawiającą składowe z przekazanych argumentów. Dodatkowo funkcja powinna sprawdzić, aby `model` był napisem długości co najmniej 4 i `moc` była większa niż 100. W przypadku nie spełnienia jednego z warunków, funkcja powinna zwracać `NULL`.
- b) `zwiększMoc` - funkcja, której argumentem jest wskaźnik do struktury typu `Komputer`. Funkcja ma dodać 50 do mocy w przekazanym argumencie.

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

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

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

Funkcja ma podwoić wszystkie elementy dodatnie na liście (o ile istnieją). Stwórz jeden przypadek testowy.