

Egzamin (pierwszy termin) - Programowanie strukturalne - Zestaw 1

Zadanie 1: 10 pkt. Zadanie 2: 12 pkt. Zadanie 3: 13 pkt. Zadanie 4: 15 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).

1. W folderze Debug1 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 i instrukcji poza komentarzami będzie powodowało 0 pkt.
2. Napisz funkcję porównującą dwie tablice jednowymiarowe o takich samych rozmiarach o wartościach typu `int`. Funkcja ma zwrócić 1 jeśli sumy elementów nieparzystych z każdej tablicy są sobie równe, oraz ma zwrócić 0 w przeciwnym wypadku. Stwórz dwa przypadki testowe dla funkcji.
3. Napisz funkcję, której argumentem są dwie dwuwymiarowe tablice tablic (zawierające zmienne typu `int`) o tych samych wymiarach oraz ich wymiary n i m . Funkcja ma zwrócić ile elementów parzystych znajduje się w obu tablicach. Stwórz przypadek testowy.

Przykład. Dla tablic:

$$\begin{bmatrix} 2 & 3 & -3 \\ 1 & 4 & 8 \end{bmatrix}, \quad \begin{bmatrix} -3 & 0 & 2 \\ 7 & 4 & 0 \end{bmatrix}$$

ma być zwrócone 7.

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

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

oraz liczbę całkowitą w . Funkcja ma pomnożyć elementy na liście przez w . Stwórz jeden przypadek testowy.