Zestaw 67

1. W folderze Debug67 znajduje się projekt z kodem w języku C. W pliku main.c w niektórych linijkach są komentarze. Twoim zadaniem jest wpisanie wartości odpowiednich zmiennych po wykonaniu konkretnej linii kodu.

*Punktacja: 1 pkt za każdą poprawną linijkę, łącznie 7 pkt.*

1. W folderze Popraw67 znajduje się kod w języku C, który nie spełnia zasad kompilacji. Popraw kod modyfikując dokładnie 1 linijkę tak, aby się kompilował. Zabronioną operacją jest komentowanie kodu. Do zmodyfikowanych linii zaliczają się zarówno linie istotne ze względu na kompilację jak i te nieistotne (np. dodanie spacji przed operatorem może być operacją nieistotną ze względu na kompilację, ale będzie liczone jako zmodyfikowana linijka).

*Punktacja: 7 pkt.*

1. Napisz funkcję, której parametrami są dwie dodatnie liczby całkowite i . Funkcja ma zwrócić największą potęgę trójki mniejszą lub równą . Stwórz dwa przypadki testowe dla funkcji.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | wynik |
| 2 | 2 | 4 | 3 |
| 5 | 5 | 10 | 9 |
| 15 | 12 | 27 | 27 |

*Punktacja: 8 pkt.*

1. Napisz funkcję, która otrzymuje trzy argumenty:

* dwa wskaźniki na funkcje o jednym argumencie typu int zwracające wartość typu int,
* wartość n typu int.

Funkcja ma zwrócić sumę wartości obu funkcji przekazanych jako wskaźnik w punkcie . Stwórz dwa różne przypadki testowe.

*Punktacja: 8 pkt.*

1. Napisz funkcję rekurencyjną, która dla otrzymanej w argumencie nieujemnej liczby całkowitej zwraca wartość elementu o indeksie ciągu zdefiniowanego w następujący sposób

Stwórz dwa przypadki testowe dla funkcji.

*Punktacja: 10 pkt.*

1. Napisz funkcję, która otrzymuje trzy argumenty: dodatnią liczbę całkowitą , liczbę całkowitą oraz -elementową tablicę tab o elementach typu int. Funkcja ma zwrócić średnią wartości elementów tablicy tab na indeksach będących dodatnimi wielokrotnościami liczby (czyli ) (jeśli takich elementów nie ma - zwróć zero). Stwórz dwa przypadki testowe dla funkcji.

*Punktacja: 10 pkt.*