Zestaw 46

1. W folderze Debug46 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 Popraw46 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 trzy dodatnie liczby całkowite , i . Funkcja ma zwrócić 2 jeśli jest podzielne jednocześnie przez i , 1 jeśli jest podzielne tylko przez jedną z liczb lub oraz zwraca 0 w przeciwnym wypadku. Stwórz przypadek testowy dla funkcji.

*Punktacja: 8 pkt.*

1. Napisz funkcję, która otrzymuje trzy argumenty:

* dwa wskaźniki na funkcje o jednym argumencie typu float zwracające wartość typu float,
* wartość x typu float.

Funkcja ma zwrócić 1 jeśli wartość pierwszej funkcji jest sześcianem wartości drugiej funkcji odpowiednio na liczbach od 1 do (np. , itd.) oraz ma zwracać -1 w przeciwnym wypadku. Stwórz dwa przypadki testowe dla funkcji, aby funkcja zwróciła obie możliwe wartości.

*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 oraz dwie -elementowe tablice tab1 i tab2 o elementach typu int. Funkcja ma zwrócić sumę wartości elementów stojących na elementach o parzystych indeksach w obu tablicach. Stwórz dwa przypadku testowe.

*Punktacja: 10 pkt.*