

## Zestaw 32

1. W folderze Debug32 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.

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

2. W folderze Popraw32 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.*

3. Napisz funkcję, której parametrami są trzy dodatnie liczby całkowite  $n$ ,  $m$  i  $k$ . Funkcja ma zwrócić sumę liczb większych niż  $n$ , mniejszych lub równych  $m$  i niepodzielnych przez  $k$ . Stwórz dwa przypadki testowe dla funkcji.

*Punktacja: 8 pkt.*

4. Napisz bezargumentową funkcję, która rezerwuje blok dwóch zmiennych typu `int`. Funkcja ma ustawić kolejno w pamięci wartości 5 i -5. Na koniec funkcja powinna zwrócić wskaźnik na początek bloku. Stwórz przypadek testowy w `main` tak, aby wyświetlić na konsoli wartości zmiennych przechowywanych na bloku stworzonym wewnątrz funkcji.

*Punktacja: 8 pkt.*

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

$$a_0 = a_1 = 1$$

$$a_{3n} = a_n, n > 0$$

$$a_{3n+1} = a_{3n} + 1, n > 0$$

$$a_{3n+2} = a_{3n+1} + 2, n \geq 0$$

Stwórz dwa przypadki testowe dla funkcji.

Przypadki testowe:

$n$	$a_n$
2	3
3	1
4	2
5	4

*Punktacja: 10 pkt.*

6. Napisz funkcję, która otrzymuje trzy argumenty: dodatnią liczbę całkowitą  $n$ , liczbę całkowitą  $m$  oraz  $n$ -elementową tablicę `tab` o elementach typu `int`. Funkcja ma zwrócić liczbę ile razy występuje liczba

$m$  w tablicy `tab`. Stwórz dwa przypadki testowe dla funkcji.

*Punktacja: 10 pkt.*