Programowanie strukturalne - Kolokwium 2 - Zestaw 35

**Każde zadanie powinno być w oddzielnym pliku. Rozwiązanie umieść na swoim indywidualnym repozytorium na Githubie (pojedyncze pliki) lub prześlij jako archiwum zip poprzez wiadomość prywatną na MS Teams. Szczegóły są opisane w zespole ćwiczeń w MS Teams.**

1. W folderze Debug235 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: 7 pkt.*

1. Napisz funkcję, która dostaje w argumencie napis. Funkcja ma zwrócić długość napisu, gdy napis nie zawiera cyfr w systemie szesnastkowym. Jeśli napis zawiera takową cyfrę, to za każdą cyfrę należy dodać 1 do długości napisu i to zwrócić. Stwórz przypadek testowy.

Przykład: dla napisu "jdET" funkcja ma zwrócić 6, dla napisu "abc3334" ma zwrócić 14, dla napisu "12" ma zwrócić 4, dla napisu "wtgh" ma zwrócić 4.

*Punktacja: 8 pkt.*

1. Napisz funkcję, której argumentem są dwie dwuwymiarowe tablice tablic (zawierająca zmienne typu int) o tych samych wymiarach oraz ich wymiary i . Funkcja ma zwrócić ile elementów jest różnych od siebie na odpowiadających miejscach w tablicach. Stwórz przypadek testowy.

Przykład. Dla tablic:

różne elementy są w kolumnie o indeksie 0 i 2, więc funkcja powinna zwrócić 8.

*Punktacja: 10 pkt.*

1. Stwórz strukturę Samolot o trzech polach model (napis) oraz liczbasilnikow (dowolny typ całkowity), liczbapasazerow (dowolny typ całkowity). Następnie stwórz funkcję, której argumentami jest tablica struktur Samolot oraz rozmiar tablicy. Funkcja ma zwrócić łączną liczbę pasażerów wszystkich samolotów z tablicy (przyjmij, że tablica nie jest pusta i wszystkie pola są zainicjowane). Stwórz przypadek testowy.

*Punktacja: 10 pkt.*

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

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

oraz liczbę całkowitą a. Funkcja ma zwrócić adres elementu o wartości równej a (w przypadku kilku elementów równych a, adres ostatniego takiego). W przypadku pustej listy lub braku elementów równych a, zwróć NULL. Stwórz jeden przypadek testowy.

*Punktacja: 15 pkt.*