

Programowanie proceduralne lab 8

Zadanie 1. (3 pkt)

Utwórz strukturę Klient, zawierająca pola: imie(char[20]), nazwisko(char[32]), kod pocztowy (jako łańcuch znaków) i wiek. Utwórz pięcioelementową tablicę struktur, a następnie z przykładowego pliku tekstowego (można napisać go ręcznie, min 5 klientów i przekierować strumień we- (. /a.out <plik. dat) wczytujemy klientów do tablicy struktur.

a). Wykorzystaj funkcję qsort(), aby posortować tablicę rosnąco względem pola nazwisko, a następnie imię.

Zadanie 2. (4 pkt)

W programie utwórz strukturę wektor, oraz strukturę abc, która ma trzy pola: wektor oraz długość wektora, i pole tolerancja. Utwórz tablicę 10 struktur abc. Wypełnij liczbami losowymi pola wektor. Napisz funkcję, która na podstawie pola wektor uzupełni pole długość wektora i pole tolerancja (zdefiniowana poniżej)

a). Wykorzystaj funkcję bsearch() aby odszukać wszystkie wektory o zadanej przez użytkownika długości d zadaną przez użytkownika (w procentach) tolerancją t.

Należy odszukać wektorów, których długości zawarte są w przedziale

$[d - (t \text{ procent z wartości } d) ; d + (t \text{ procent z wartości } d)]$

Proszę wypisać na ekran tablicę struktur, przed i po sortowaniu oraz znalezione wektory o długościach z zadanego przedziału.

Zadanie 3. (3 pkt)

Proszę napisać funkcję o zmiennej liczbie argumentów:

..... va_tablice(.....);

do której przekazujemy dowolną ilość tablic jednowymiarowych, o dowolnym (różnym) rozmiarze. Funkcja ma zwrócić wskaźnik do tablicy dwuwymiarowej, której wierszami będą tablice przekazane do funkcji.

Funkcję proszę przetestować dla podanych zestawów tablic, utworzoną tablicę proszę wypisać w funkcji main()

1.tablice wejściowe :

```
int tab_A[] = {3, 2, 5, 4};
int tab_B[] = {12, 10, 13, 12, 18, 7};
int tab_C[] = {22, 20};
```

2.tablice wejściowe :

```
int tab_A1[] = {31, 22, 43, 34, 55, 26};
int tab_B1[] = {64, 66};
int tab_C1[] = {52, 50, 56, 55};
int tab_D1[] = {26, 62, 36, 63, 77, 68, 69};
```

