Uwaga: Należy rozwiązać cztery wybrane zadania.

Zadanie 1

Napisać funkcję bool Szukaj (int A[], int n, int liczba), która sprawdza, czy dokładnie jeden element tablicy A ma wartość równą szukanej liczbie. Funkcja zwraca true, jeśli zachodzi ten warunek, w przeciwnym wypadku zwraca false. Napisz program testujący opracowaną funkcję.

Zadanie 2

Zbuduj funkcję, która zastępuje napis poczynając od wskazanego miejsca do końca nowym tekstem.. Funkcja ma prototyp:

void replaceString(char *napis, char *zastap, int poczatek)

gdzie napis jest przetwarzanym ciągiem znaków, zastap nowym tekstem, poczatek jest indeksem pierwszego zastępowanego znaku.

Napisz program testujący opracowaną funkcję.

Zadanie 3

W tablicy Wyniki[20] przechowujemy wyniki egzaminu pewnej grupy studentów. Tablica zawiera następujące dane:

- Imie[30],
- Nazwisko[30],
- numerIndeksu.
- · liczba punktów.

Należy

- wczytać dane studentów i wyniki i umieścić w tablicy Wyniki (funkcja wczytaj);
- wypisać zawartość tablicy wyniki w czytelnej postaci (funkcja pokazWyniki);
- wypisać imię i nazwisko oraz wynik studenta, który uzyskał największą liczbę punktów (funkcja Najlepszy);
- wypisać numer indeksu i wynik studenta, który uzyskał najmniejszą liczbę punktów (funkcja Najslabszy),
- podać średnią liczbę punktów uzyskaną w tej grupie (funkcja SredniWynik).

Do zerowego terminu egzaminu przystąpiło tylko 4 studentów. Przetestować program dla tego egzaminu.

Zadanie 4

Dana jest lista struktur reprezentująca elementy rzadkiej tablicy dwuwymiarowej. Pojedyncza struktura tej listy zawiera trzy liczby: (x,y,v), gdzie x – numer wiersza, y – numer kolumny, v - wartość elementu tablicy o współrzędnych (x,y). Liczby x i y są typu całkowitego bez znaku, zmiennoprzecinkowa liczba v jest typu double.

Opracuj funkcję z argumentem w postaci odniesienia do listy, która zwraca jednowymiarową tablicę sum w wierszach. Napisz program testujący opracowaną funkcję.

Zadanie 5

Napisać program do obsługi komisu samochodowego. Program powinien umożliwiać przetwarzanie następujących informacji o poszczególnych samochodach:

- Marka samochodu (napis),
- Rok produkcji (liczba całkowita),
- Cena (liczba zmiennoprzecinkowa).

W pamięci komputera te informacje powinny być przechowywane w postaci listy ze wskaźnikiem do początku listy i do jej końca. Poniżej podano deklaracje struktur danych i funkcję <code>DodajAuto</code> dodającą kolejny element do listy. Program ma umożliwić użytkownikowi:

- Drukowanie listy samochodów dostępnych w komisie;
- Dodawanie nowego samochodu do listy;
- Zapisanie listy do pliku "Komis.txt";
- Odczytanie listy z pliku "Komis.txt";
- Usuwanie wskazanego samochodu z listy;
- Wybranie samochodów wyprodukowanych po wskazanym przez użytkownika roku;

Samochód opisać wybraną przez siebie strukturą.

```
const int MAX=30;
                                       struct AUTO {
struct AUTO {
                                        char *marka;
 char marka[MAX];
                                         int rok_prod;
 int rok_prod;
                                         double cena;
 double cena;
                                       } ;
} ;
                                       struct WEZEL {
struct WEZEL {
                                        AUTO info;
 AUTO info;
                                         WEZEL *nast;
 WEZEL *nast;
                                       } ;
                                       struct LISTA {
} ;
struct LISTA {
                                         WEZEL *pocz;
 WEZEL *pocz;
                                         WEZEL *kon;
 WEZEL *kon;
                                       } ;
} ;
```