

1. ЗАДАТАК

- Направити функцију која се назива **brojSlova** и која **враћа број** појављивања словчаних карактера у **stringу** који је прослеђен као аргумент.

```
int brojSlova(char* string){
    int brslova=0,i=0;
    while(string[i]!='\0'){
        if((string[i]>='a' && string[i]<='z')
            ||
            (string[i]>='A' && string[i]<='Z'))
            brslova++;
        i++;
    }
    return brslova;
}
```

- Дефинисати структуру **TRIGTACKA** која има податке **nadmVisina** (цео број), **geoSirina** (string), **geoDuzina** (string) и **oznaka** (string).

```
typedef struct TRIGTACKA{
    int nadmVisina;
    char* geoSirina;
    char* geoDuzina;
    char* oznaka;
} ttracka;
```

2. ЗАДАТАК

- Дефинисати структуру **CVOR** који може да чува податак било ког типа и да омогућава додавање новог чвора на крај листе.

```
typedef struct CVOR{
    void * podatak;
    struct CVOR* preth, *sled;
} cvor;
```

- Направити функцију за додавање податка типа структура **TRIGTACKA** у листу **lista** коришћењем структуре **CVOR**; функција се назива **dodajTacke** и има као аргументе податке **nadmVisina** (цео број), **geoSirina** (string), **geoDuzina** (string) и **oznaka** (string). Податак **lista** представљен је глобалном промењивом.

```

#include "definicije.h"
#include <stdlib.h>
#include <string.h>
cvor * lista=0;

void dodajTTacke(int nadmVisina,char* geoSirina,char* geoDuzina,char* oznaka){
    cvor * noviCvor;
    //napravi ttracku i napuni podacima
    ttrack* t=(ttrack*)malloc(sizeof(ttrack));
    t->nadmVisina=nadmVisina;
    t->geoSirina=(char*)malloc(strlen(geoSirina)+1);
    strcpy(t->geoSirina,geoSirina);
    t->geoDuzina=(char*)malloc(strlen(geoDuzina)+1);
    strcpy(t->geoDuzina,geoDuzina);
    t->oznaka=(char*)malloc(strlen(oznaka)+1);
    strcpy(t->oznaka,oznaka);
    //NAPRAVITI CVOR I ubaciti ttracku u cvor
    noviCvor=(cvor *)malloc(sizeof(cvor));
    noviCvor->podatak=t;
    noviCvor->preth=0;
    noviCvor->sled=0;
    //dodati cvor u listu
    if(lista==0)
        lista=noviCvor;
    else{
        cvor* pom=lista;
        while(pom->sled != 0)
            pom=pom->sled;
        pom->sled=noviCvor;
        noviCvor->preth=pom;
    }
}

```

- Направити функцију за сортирање листе *lista*, сачињена од чворова типа структуре **CVOR**, који имају податак типа структура **TRIGTACKA** по растућој надморској висини; функција се назива *sortirajPoVisini*. Податак *lista* представљен је глобалном променљивом.

```

void sortirajPoVisini(){
    cvor* novaLista=0,*novaGlava;
    if(lista==0) return;
    if(lista->sled==0) return;
    do{
        novaGlava=nadjiNajvisuTacku();
        if(novaLista==0)
            novaLista=novaGlava;
        else{
            cvor * pom=novaLista;
            while(pom->sled!=0){
                pom=pom->sled;
            }
            pom->sled=novaGlava;
        }
    }while(lista!=0);
    lista=novaLista;
}

```

```

//f-ja nalazi cvor sa ttackom koja ima najveću nadmVisinu
//vraca je i i izbacuje iz liste
cvor * nadjiNajvisuTacku(){
    ttacka* tekTacka,*najvisaTacka;
    cvor* najvisi=lista, *tek=lista;
    if(lista!=0){
        while(tek->sled != 0){
            tekTacka=(ttacka*)tek->sled->podatak;
            najvisaTacka=(ttacka*)najvisi->podatak;
            if(tekTacka->nadmVisina > najvisaTacka->nadmVisina){
                najvisi=tek->sled;
            }
            tek=tek->sled;
        }
        if(tek==lista){//slučaj 1 cvora u listi
            lista=0;
            return tek;
        }
    }
    if(najvisi->preth!=0)
        najvisi->preth->sled=najvisi->sled;
    if(najvisi->sled!=0)
        najvisi->sled->preth=najvisi->preth;
    najvisi->preth=0;
    najvisi->sled=0;
    return najvisi;
}

```