

Strukture podataka i algoritmi 1
Jun, 2011

1. Napisati program za upravljanje sistemom zdravstvene zaštite na sledeći način. Program treba da sadrži podatke o svim osiguranicima na nekoj teritoriji, pri čemu svaki osiguranik treba da poseduje ime, jedinstveni lični broj osiguranika, lokaciju stanovanja (X i Y koordinata na mapi) i pokazivač na najbližu zdravstvenu stanicu (gledano vazdušnom linijom). Za svaku zdravstvenu stanicu su poznati jedinstveni broj stanice i lokacija (X i Y koordinata na mapi).
 - a. Napisati funkciju *UčitajStanice* koja iz datoteke učitava podatke o svim zdravstvenim stanicama na posmatranoj teritoriji. Ukupan broj stanica se definiše prilikom učitavanja.
(3 poena)
 - b. Napisati funkciju *DodajOsiguranika* koja učitava podatke o novom osiguraniku, dodaje ga u kolekciju osiguranika i povezuje sa odgovarajućom zdravstvenom stanicom. Ukupan broj osiguranika nije unapred poznat i menja se prilikom svakog dodavanja osiguranika.
(10 poena)
 - c. Napisati funkciju *ZatvoriStanicu* koja omogućava isključivanje određene stanice iz sistema, pri čemu se osiguranici koji joj pripadaju moraju zbrinuti u najbližoj stanici koja je ostala u sistemu.
(8 poena)
 - d. Napisati funkciju *Statistika* koja štampa broj osiguranika u svakoj zdravstvenoj stanici.
(4 poena)
2. Korišćenjem rekurzivne funkcije izračunati zbir svih neparnih brojeva manjih ili jednakih zadatom neparnom broju.
(5 poena)

U nastavku je dato rešenje jednog od studenata.

Rešenje 1

(Miloš Simić)

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>

typedef struct{
    int broj;
    float x;
    float y;
} Stanica;

typedef struct{
    char ime[35];
    int broj;
    float x;
    float y;
    Stanica *stanica;
} Osiguranik;

Stanica *stanice;
Osiguranik *osiguranici;
int no, ns;

void UcitajStanice(FILE *F){
    int i;
    fscanf(F, "%d", &ns);
    stanice=(Stanica*)malloc(ns*sizeof(Stanica));
    for(i=0;i<ns;i++)
        fscanf(F, "%d%f%f", &stanice[i].broj, &stanice[i].x, &stanice[i].y);
}

double Rastojanje(float x1, float y1, float x2, float y2){
    return sqrt((double)((x1-x2)*(x1-x2)+(y2-y1)*(y2-y1)));
}

void NajblizaStanica(Osiguranik *osiguranik){
    int i;
    double rastojanje, rastojanje_min;
    osiguranik->stanica=&stanice[0];
    rastojanje_min=Rastojanje(osiguranik->x, osiguranik->y, stanice[0].x, stanice[0].y);
    for(i=1;i<ns;i++)
        if((rastojanje=Rastojanje(osiguranik->x, osiguranik->y, stanice[i].x, stanice[i].y))<rastojanje_min)
            {rastojanje_min=rastojanje; osiguranik->stanica=stanice+i;}
}

void DodajOsiguranika( char *ime, int broj, float x, float y){
    osiguranici=(Osiguranik*)realloc(osiguranici, (no+1)*sizeof(Osiguranik));
    osiguranici[no].broj=broj;
    osiguranici[no].x=x;
    osiguranici[no].y=y;
    strcpy(osiguranici[no].ime, ime);
    NajblizaStanica(osiguranici+no);
    no++;
}

void ZatvoriStanicu(int broj){
```

```

    int i,j;
    for(i=0;i<ns;i++) if(stanice[i].broj==broj) break;
    j=i;
    for(;i<ns-1;i++) stanice[i]=stanice[i+1];
    ns--;
    for(i=0;i<no;i++)
        if(osiguranici[i].stanica>stanice+j) (osiguranici[i].stanica)--;
        else if(osiguranici[i].stanica==stanice+j) NajblizaStanica(osiguranici+i);
}

void Statistika(void){
    int i,j,br;
    for(i=0;i<ns;i++){
        br=0;
        for(j=0;j<no;j++)
            if(osiguranici[j].stanica==stanice+i) br++;
        printf("Stanica %d - broj osiguranika %d.\n",i+1,br);
    }
}

```