



STRUKTURE PODATAKA I ALGORITMI 1

VEŽBE 10

Tijana Ristović
Aleksa Cerovina
Filip Radovanović
Đorđe Nedić



```
#include <stdio.h>
typedef struct point {
    int x, y;
} POINT;

void get_point_wrong(POINT p) {
    printf("x = ");
    scanf("%d", &p.x);
    printf("y = ");
    scanf("%d", &p.y);
}

void get_point(POINT* p) {
    printf("x = ");
    scanf("%d", &p->x);
    printf("y = ");
    scanf("%d", &p->y);
}
```



```
main() {  
    POINT a = {0, 0};  
  
    printf("get_point_wrong\n");  
    get_point_wrong(a);  
    printf("a: x = %d, y = %d\n", a.x, a.y);  
  
    printf("get_point\n");  
    get_point(&a);  
    printf("a: x = %d, y = %d\n", a.x, a.y);  
}
```



- Napisati program koji sa standardnog ulaza učitava dva pravougaonika u ravni A, B i na standardni izlaz ispisuje površinu preseka pravougaonika, površinu unije pravougaonika, površinu razlike pravougaonika(A\B). Pravougaonik se zadaje preko x,y koordinata donjeg levog i gornjeg desnog temena pravougaonika. Stranice pravougaonika su paralelne koordinatnim osama.

```
#include <stdio.h>
#include <stdlib.h>
typedef struct _rect
{
    float x,y; /*koordinate donjeg levog ugla */
    float x1,y1; /*koordinate gornjeg desnog ugla */
} Pravougaonik ;

void ucitavanje (Pravougaonik *P); /*ucitavanje donjeg levog i
gornjeg desnog temena pravougaonika */
Pravougaonik Presek(Pravougaonik *A, Pravougaonik *B);/*presek
pravougaonika A i pravougaonika B */
float Povrsina (Pravougaonik *A);
```



```
main()
{
    Pravougaonik A,B ; /*zadati pravougaonici */
    Pravougaonik presekAB ; /*presek zadata dva pravougaonika */
    float povrsinaA; /*povrsina pravougaonika A*/
    float povrsinaB; /*povrsina pravougaonika B*/
    float povrsAB; /*povrsina preseka*/
    učitavanje(&A);
    učitavanje(&B);
    presekAB = Presek (&A , &B); /*odredjivanje A*B */
    /*izracunavanje površine datih pravougaonika i njihovog
    preseka */
    povrsinaA=Povrsina(&A);
    povrsinaB=Povrsina(&B);
    povrsAB= Povrsina (&presekAB);
    /*ispis rezultata */
    printf( "Povrsina unije : %g\n", povrsinaA+povrsinaB-povrsAB);
    printf( "Povrsina preseka : %g\n", povrsAB);
    printf( "Povrsina prve razlike : %g\n", povrsinaA - povrsAB);
    return 0;
}
```



```
void učitavanje (Pravougaonik * P)
{ /*učitavanje koordinata za odgovarajuća dva temena */
    scanf("%f%f%f%f", &P->x, &P->y, &P->x1, &P->y1);
}

float Povrsina (Pravougaonik *P )
{ /*vraca se rezultat mnozenja duzina dveju susednih stranica
  znajuci za paralelnost sa koordinatnim osama */
    return (P->x1 -P->x) * (P->y1 - P->y );
}
```



```
Pravougaonik Presek (Pravougaonik *A, Pravougaonik *B)
{
    /*Racuna se i vraca presek pravougaonika A i B.
    Ako se A i B ne seku , funkcija vraca pravougaonik cije
    su sve koordinate jednake 0 */
    Pravougaonik rezultat; /*presek zadata dva
    pravougaonika */
    /*odredjivanje koordinata donjeg levog ugla preseka-
    uociti tacka i strelica notaciju kod rezultat i kod A, B */
    rezultat.x = (A->x > B->x) ? A->x : B->x;
    rezultat.y = (A->y > B->y) ? A->y : B->y;
    /*odredjivanje koordinata gornjeg desnog ugla preseka =>
    relacijski operator manje */
    rezultat.x1 = (A->x1 < B->x1) ? A->x1 : B->x1;
    rezultat.y1 = (A->y1 < B->y1) ? A->y1 : B->y1;
    /*pravljenje korekcije u slucaju da je rezultat preseka
    prazan */
    if (rezultat.x >= rezultat.x1 || rezultat.y >=
    rezultat.y1 )
        rezultat.x=rezultat.y = rezultat.x1 = rezultat.y1 = 0;
    return rezultat;
}
```



- Šta je rezultat rada sledećeg programa?

```
#include <stdio.h>
union primer {
    int broj;
    char slovo;
    float broj_r;
};
main() {
    union primer p;
    printf("sizeof(int)=%ld \t", sizeof(int));
    printf("sizeof(char)=%ld \t", sizeof(char));
    printf("sizeof(float)=%ld \n", sizeof(float));
    printf("sizeof(union primer)=%ld\n", sizeof(union primer));
    p.broj=5;
    printf("p.broj=%d \n", p.broj);
    p.slovo='D';
    printf("p.slovo=%c\t p.broj=%d\n", p.slovo, p.broj);
    p.broj_r=1.23;
    printf("p.broj_r=%f\tp.slovo=%c\t\n", p.broj_r, p.slovo, p.broj);
}
```




- Primer definisanja unije:

```
#include <stdio.h>
#include <string.h>
struct radnik {
    char prezime[20];
    char ime[20];
    char plata_ili_nadnica;
    union {
        float plata;
        struct {
            int sati_rada;
            float satnica;
        } nadnica;
    } zarada;
} osoba[20];
```



- Primer definisanja unije:

```
main() {  
    strcpy(osoba[0].prezime, "Peric");  
    printf("%s\t", osoba[0].prezime);  
    osoba[0].zarada.plata=54358.5;  
    printf("%.2f\n", osoba[0].zarada.plata);  
    strcpy(osoba[1].prezime, "Lazic");  
    printf("%s\t", osoba[1].prezime);  
    osoba[1].zarada.nadnica.sati_rada=54;  
    osoba[1].zarada.nadnica.satnica=850.2;  
    printf("%.2f\n", osoba[1].zarada.nadnica.sati_rada*  
osoba[1].zarada.nadnica.satnica);  
}
```