



# Algoritmi i programiranje

- Računske vežbe V termin -

# Sadržaj

- 5 zadataka
- Napomena:
  - 1-3 zadatak - Matematički problemi
  - 4-5 zadatak - Nizovi
    - Referenca: 214. strana, poglavlje 5.2, Uvod u programiranje i programski jezik C, dr Vladimir Ćirić

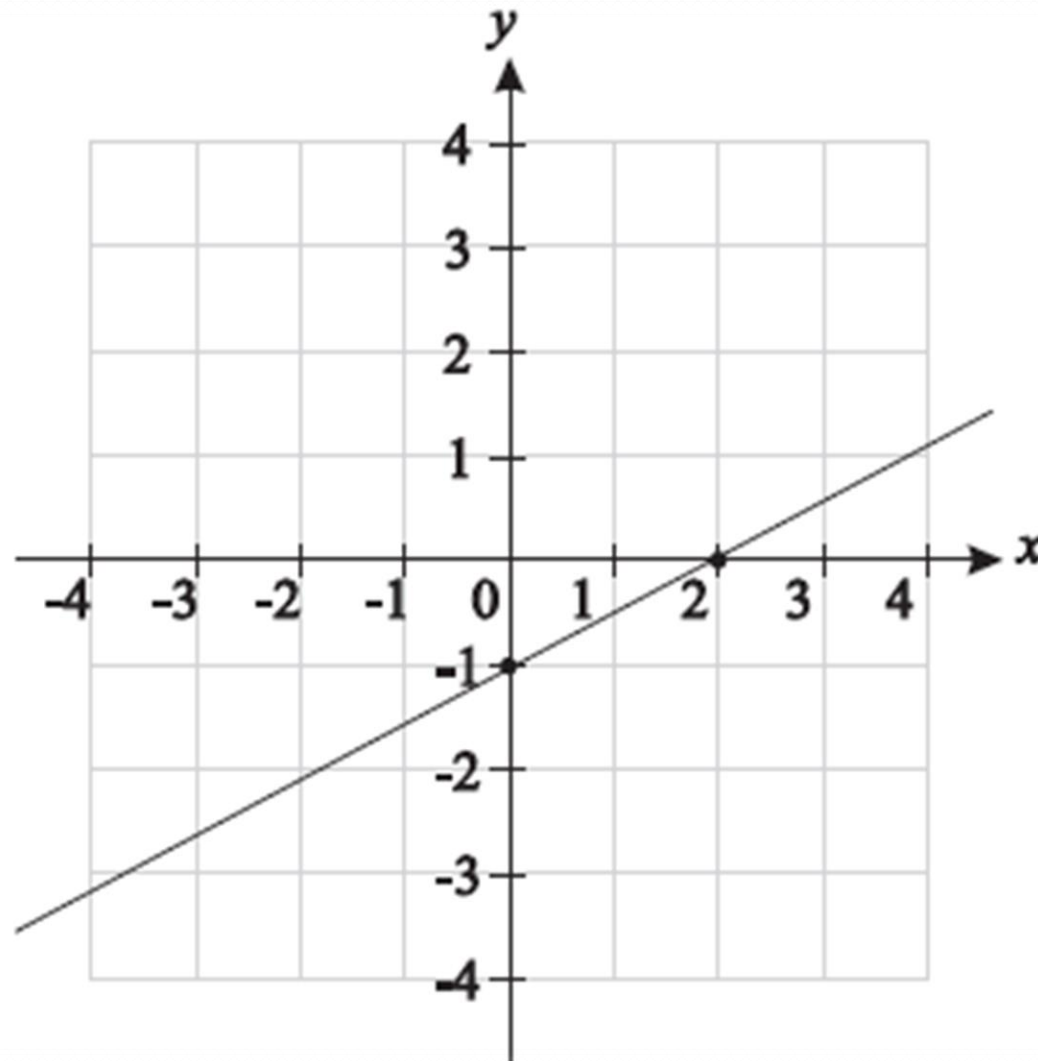
# Zadatak 1

- Nacrtati strukturni dijagram toka algoritma i na programskom jeziku C napisati strukturni program koji određuje i prikazuje sa koje strane prave sa slike 1 se nalazi tačka p. Koordinate (x, y) tačke p zadaje korisnik. Jednačina prave kroz dve tačke data je sa

$$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1} (x - x_1)$$

Napomena: tačke na pravoj proizvoljno uključiti u bilo koju oblast.

# Zadatak 1 - Slika 1



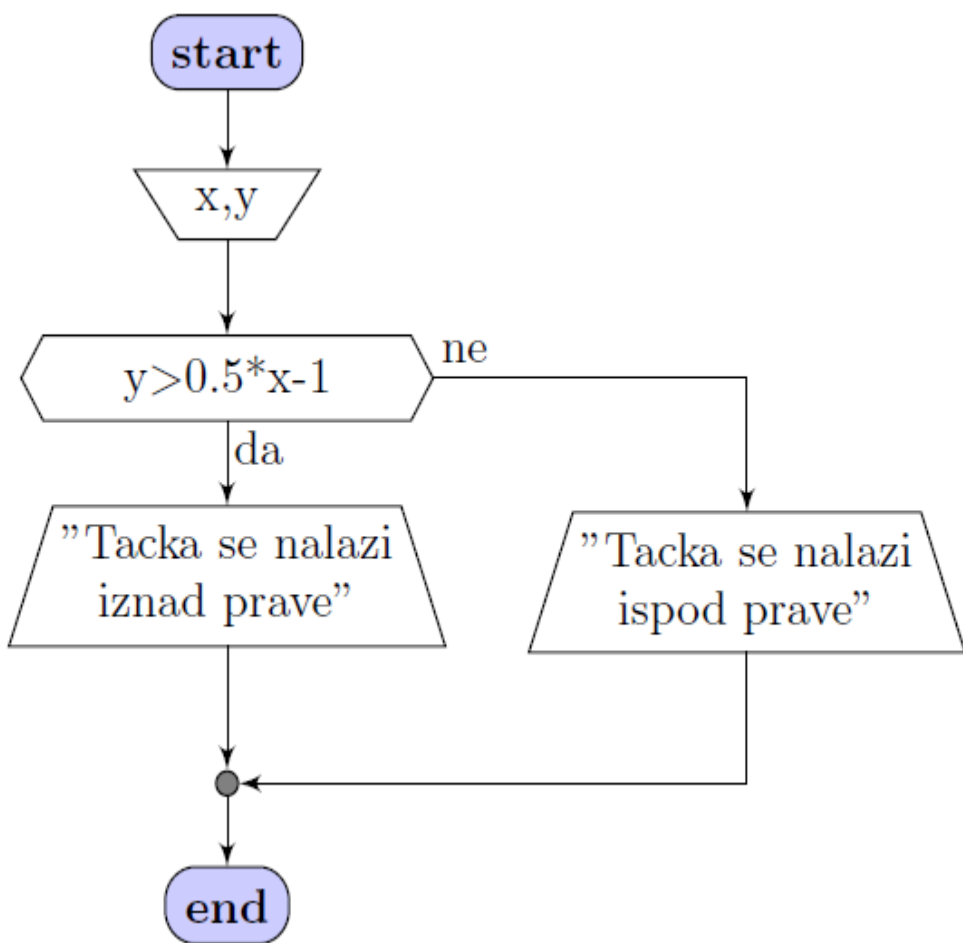
# Objašnjenje zadatka 1

Na osnovu slike 1, za tačke kroz koje prolazi prava izaberimo  $t_1 = (0, -1)$  i  $t_2 = (2, 0)$ . Zamenom ovih tačaka u jednačinu prave dobijamo jednačinu konkretne prave datu na slici kao

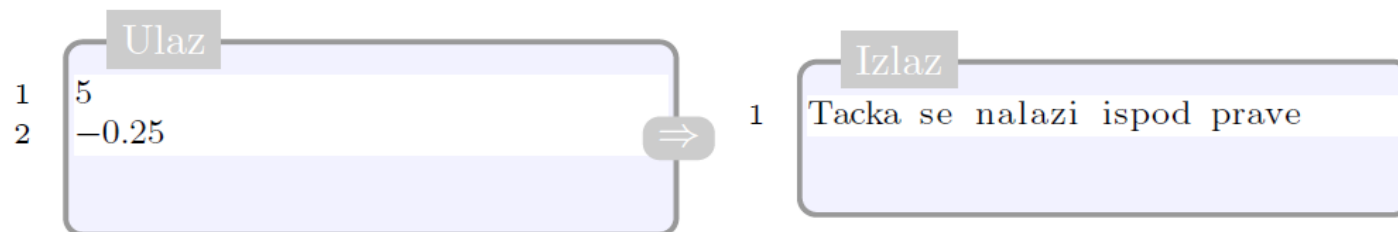
$$y = \frac{1}{2}x - 1$$

Za tačke koje nisu na pravoj važe nejednakosti  $y < \frac{1}{2}x - 1$ , za tačke ispod, i  $y > \frac{1}{2}x - 1$  za tačke iznad prave, što se može proveriti zamenom konkretne tačke, npr.  $t = (0, 0)$  u jednačinu prave.

# Zadatak 1 – Rešenje



```
1  #include <stdio.h>
2
3  void main()
4  {
5      float x,y;
6      scanf("%f",&x);
7      scanf("%f",&y);
8      if(y>0.5*x - 1)
9          printf("Tacka se nalazi iznad prave\n");
10     else
11         printf("Tacka se nalazi ispod prave\n");
12 }
```



# Zadatak 2

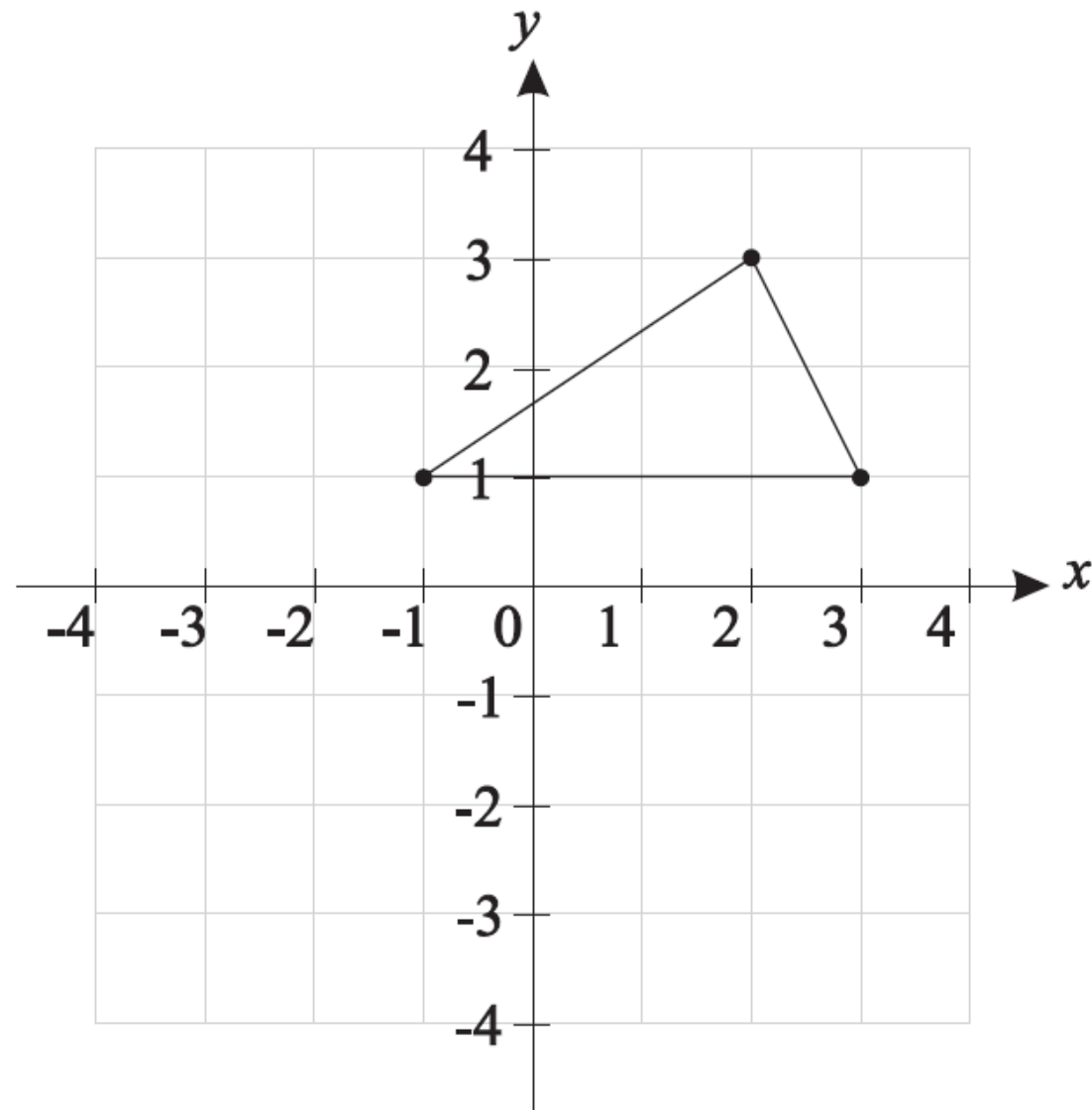
- Nacrtati strukturni dijagram toka algoritma i na programskom jeziku C napisati strukturni program koji određuje i prikazuje koliko se tačaka, od ukupno  $N$ , nalazi unutar, a koliko van trougla zadatog na slici. Parametar  $N$  zadaje korisnik, a za njim redom i koordinate tačaka  $(x, y)$ . Jednačina prave kroz dve tačke data je sa

$$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1} (x - x_1) .$$

Napomena: smatrati da su stranice trougla deo unutrašnje oblasti.

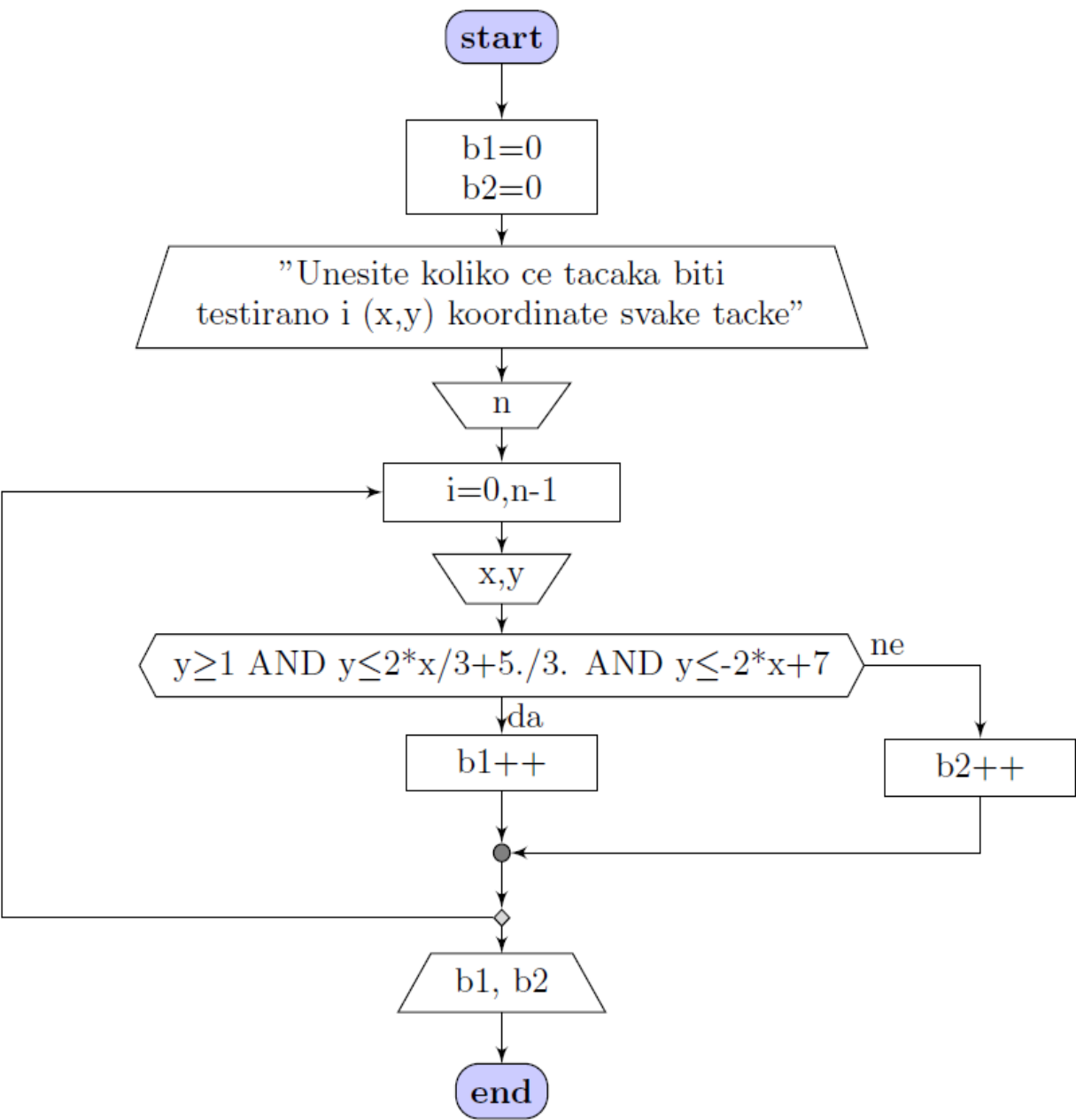


# Zadatak 2 - Slika 2





# Zadatak 2 – Rešenje 1/3



# Zadatak 2 – Rešenje 2/3

```
1  #include <stdio.h>
2  void main()
3  {
4      float x, y;
5      int n,i,b1=0,b2=0;
6      printf("Unesite_koliko_cetacka_biti_testirano_i_(x,y)_koordinate_svake_
tacke\n");
7      scanf("%d",&n);
8      for(i=0;i<n;i++)
9      {
10         scanf("%f",&x);
11         scanf("%f",&y);
12         if( y >= 1 && y <= 2*x/3 + 5./3. && y <= -2*x + 7 )
13             b1++;
14         else
15             b2++;
16     }
17     printf("Unutar_trougla_je_%d_tacka_a_van_trougla_je_%d_tacka\n",b1,b2);
18 }
```

# Zadatak 2 – Rešenje 3/3

Ulaz

```
1 6
2 2.0 2.0
3 1.0 2.0
4 3.0 1.0
5 1.0 1.0
6 4.0 3.0
7 -2.0 1.0
```



Izlaz

```
1 Unesite koliko ce tacaka biti
   testirano i (x,y)
   koordinate svake tacke
2 Unutar trougla je 4 tacaka a
   van trougla je 2 tacaka
```

# Zadatak 3

- Dato je  $N$  tačaka u ravni koordinatama  $(x, y)$ . Nacrtati strukturni dijagram toka algoritma i na programskom jeziku C napisati strukturni program koji za svaku tačku ispituje da li pripada nekoj od oblasti A, B ili C. Na izlazu prikazati redni broj tačke, njene koordinate i oznaku oblasti kojoj pripada (\*ako ne pripada nijednoj od zadatih oblasti). Oblasti su definisane kružnicama, prikazanim na slici 3, koje su opisane jednačinama na sledeći način:

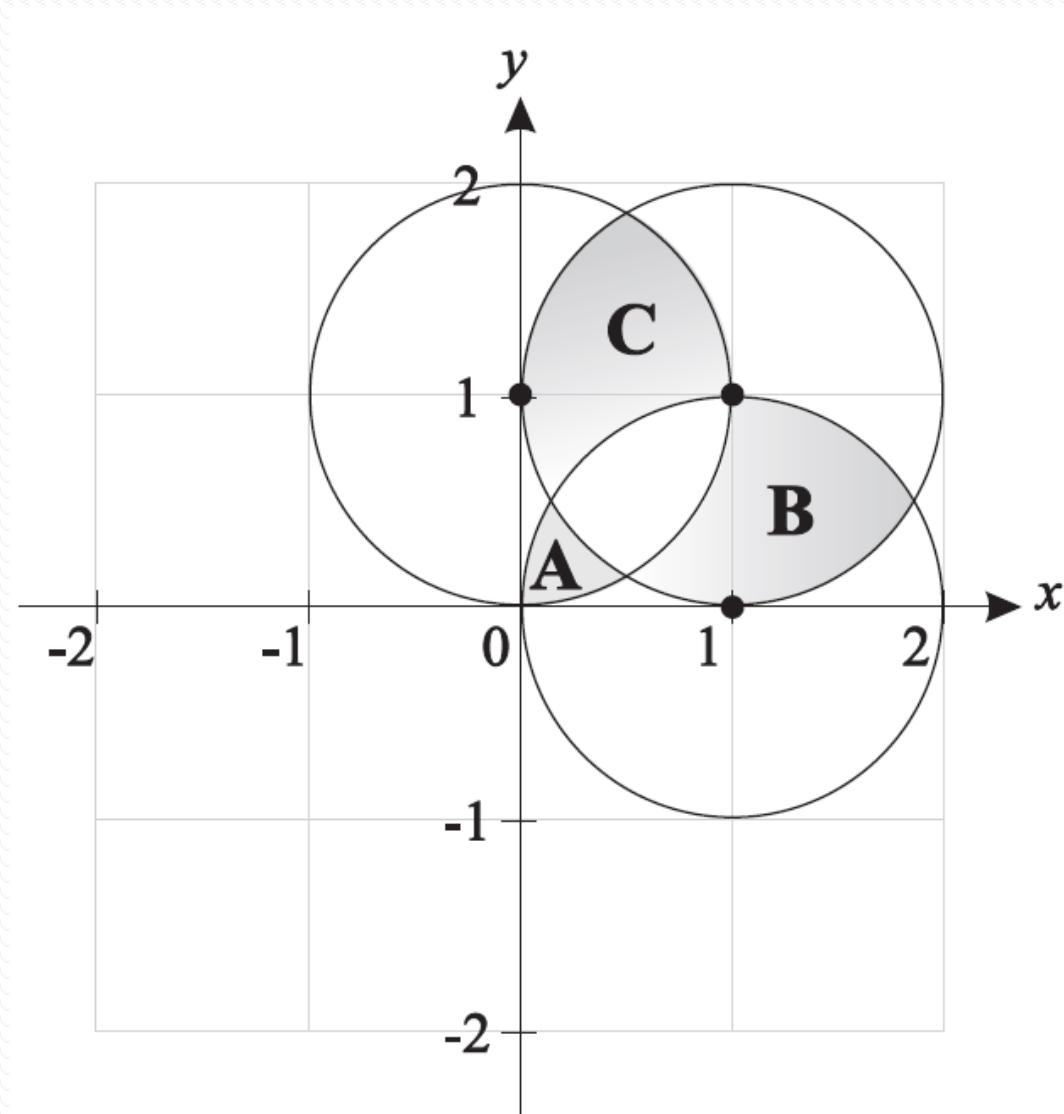
$$K_1 : (x - 1)^2 + y^2 = 1$$

$$K_2 : (x - 1)^2 + (y - 1)^2 = 1$$

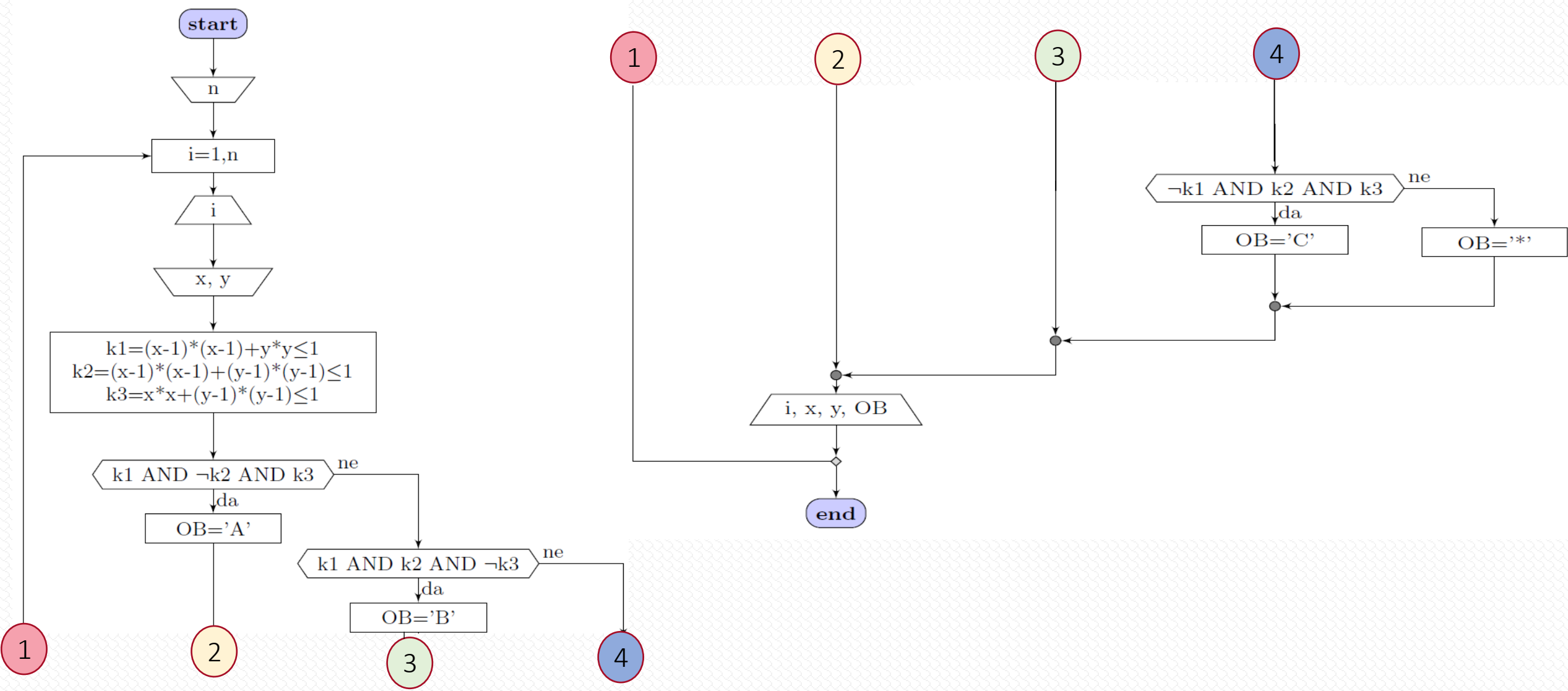
$$K_3 : x^2 + (y - 1)^2 = 1$$

Napomena: za tačke na kružnici smatrati da pripadaju unutrašnjosti kruga.

# Zadatak 3 - Slika 3



# Zadatak 3 – Rešenje 1/3



# Zadatak 3 – Rešenje 2/3

```
1  #include <stdio.h>
2  void main()
3  {
4      int n, i, k1, k2, k3;
5      char OB;
6      float x,y;
7      printf("Unesite koliko tacaka cete testirati:\n");
8      scanf("%d", &n);
9      for(i = 1; i <= n; i++)
10     {
11         printf("Unesite x i y koodrinat %d. tacke:\n", i);
12         scanf("%f%f",&x,&y);
13         k1 = (x-1)*(x-1) + y*y <= 1;
14         k2 = (x-1)*(x-1) + (y-1)*(y-1) <= 1;
15         k3 = x*x + (y-1)*(y-1) <= 1;
16         if(k1 && !k2 && k3)
17             OB = 'A';
18         else if(k1 && k2 && !k3)
19             OB = 'B';
20         else if (!k1 && k2 && k3)
21             OB = 'C';
22         else
23             OB = '*';
24         printf("%d. tacka sa koordinatama %.2f,%.2f je u oblasti \n", i,x,y,OB);
25     }
26 }
```



# Zadatak 3 – Rešenje 3/3

## Ulaz

```
1 4
2 0.2 0.3
3 0.2 0.68
4 0.85 0.12
5 0.95 1.86
```



## Izlaz

```
1 Unesite koliko tacaka cete
   testirati:
2 Unesite x i y koodrinate 1.
   tacke:
3 1. tacka sa koordinatama
   0.20,0.30 je u oblasti A
4 Unesite x i y koodrinate 2.
   tacke:
5 2. tacka sa koordinatama
   0.20,0.68 je u oblasti C
6 Unesite x i y koodrinate 3.
   tacke:
7 3. tacka sa koordinatama
   0.85,0.12 je u oblasti B
8 Unesite x i y koodrinate 4.
   tacke:
9 4. tacka sa koordinatama
   0.95,1.86 je u oblasti *
```

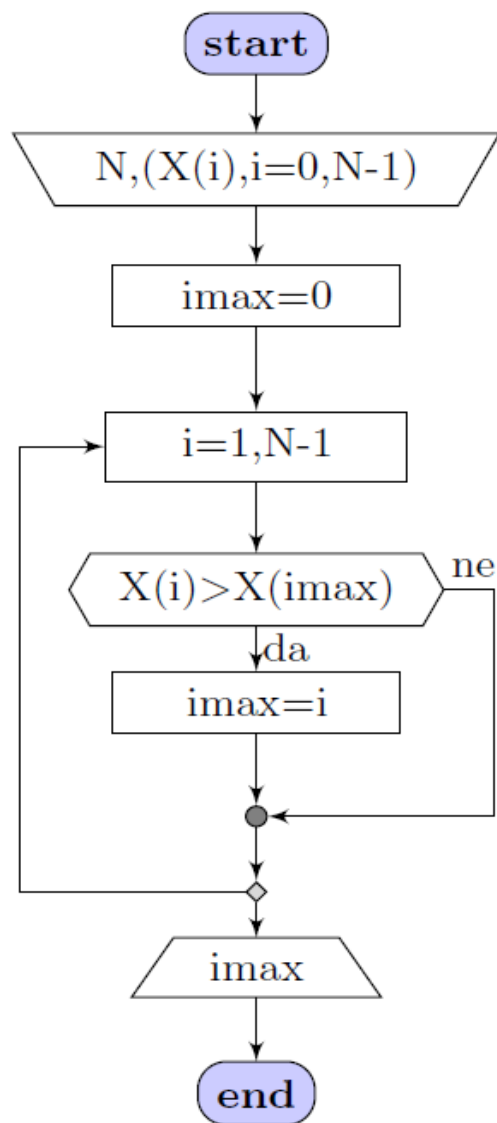
# NIZOVI

Referenca: 214. strana, poglavlje 5.2, Uvod u programiranje i programski jezik C, dr Vladimir Ćirić

# Zadatak 4

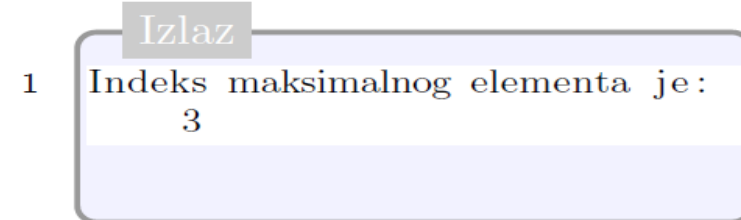
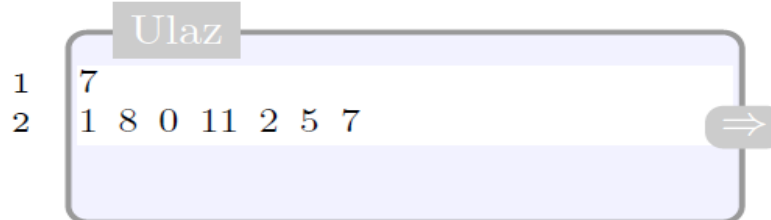
- Nacrtati strukturni dijagram toka algoritma i na programskom jeziku C napisati strukturni program koji određuje i prikazuje indeks maksimalnog elementa u celobrojnom nizu  $X$  sa  $N$  elemenata.

# Zadatak 4 - Rešenje



```

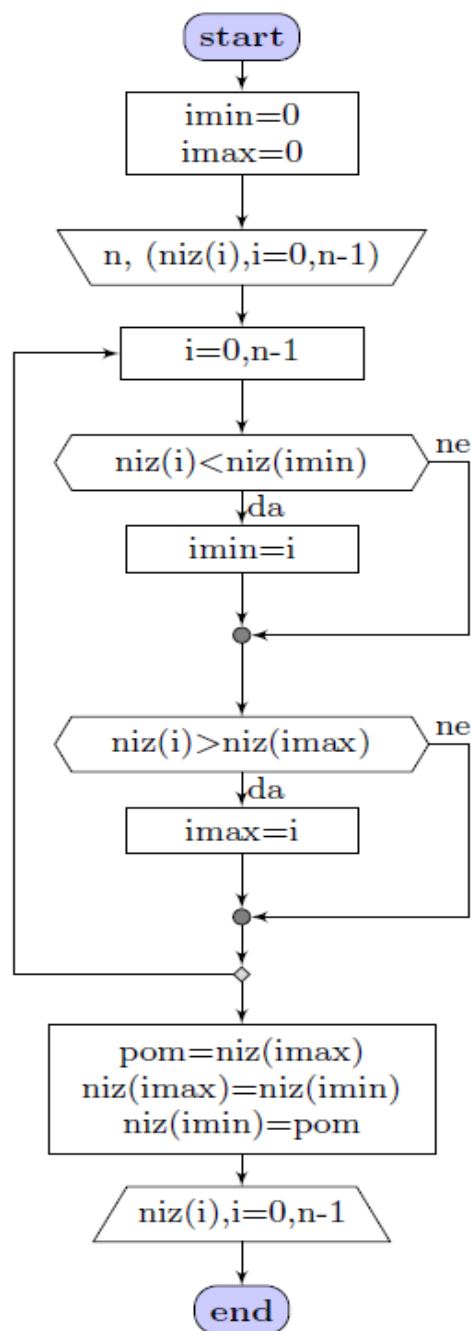
1  #include <stdio.h>
2  void main()
3  {
4      int i, imax, X[30], N;
5      scanf ("%d", &N);
6      for ( i = 0; i < N; i++)
7          scanf ("%d", &X[i]);
8      imax=0;
9      for ( i = 1; i < N; i++)
10         if ( X[i] > X[imax])
11             imax=i;
12     printf ("Indeks maksimalnog elementa je: %d", imax);
13 }
  
```



# Zadatak 5

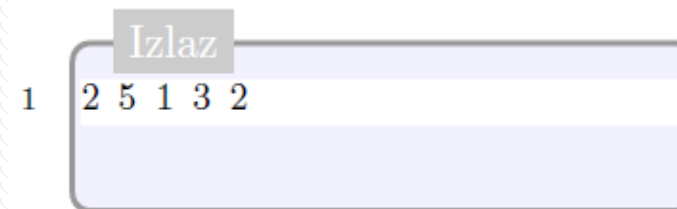
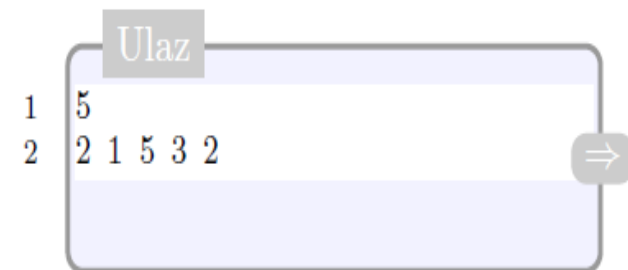
- Nacrtati strukturni dijagram toka algoritma i na programskom jeziku C napisati strukturni program koji vrši zamenu mesta minimalnom i maksimalnom elementu niza. Broj elemenata niza i elemente niza zadaje korisnik. Prikazati niz nakon transformacije.

# Zadatak 5 - Rešenje



```

1  #include <stdio.h>
2  void main()
3  {
4      int niz[100], n, i, imin, imax, pom;
5      imin=0;
6      imax=0;
7      scanf("%d", &n);
8      for(i = 0; i < n; i++)
9          scanf("%d", &niz[i]);
10     for(i=0; i < n; i++)
11     {
12         if(niz[i] < niz[imin])
13             imin = i;
14         if(niz[i] > niz[imax])
15             imax = i;
16     }
17     pom = niz[imax];
18     niz[imax] = niz[imin];
19     niz[imin] = pom;
20     for(i=0; i < n; i++)
21         printf("%d ", niz[i]);
22 }
  
```





# PITANJA

Forum na sajtu predmeta  
[cs.elfak.ni.ac.rs/nastava](http://cs.elfak.ni.ac.rs/nastava)