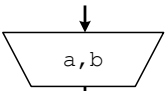

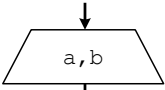
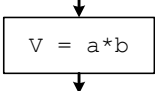
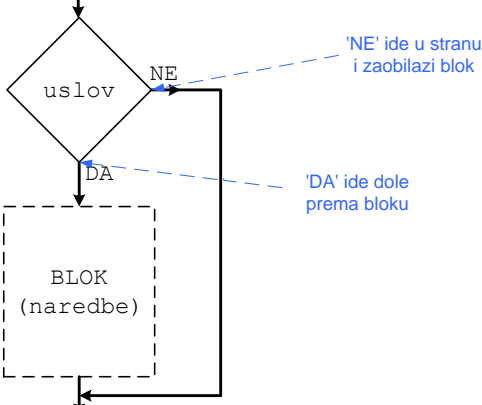
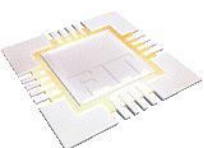


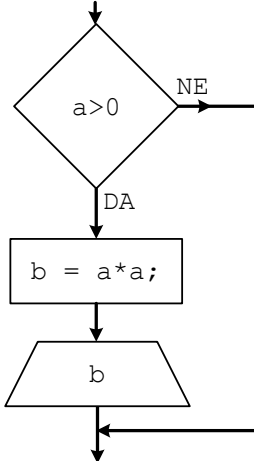
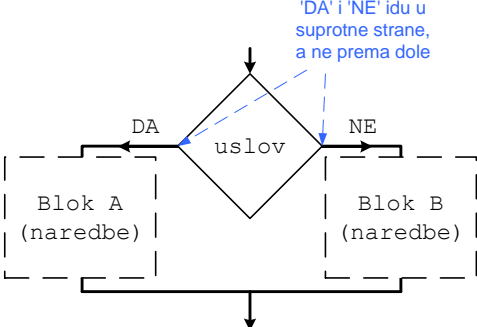
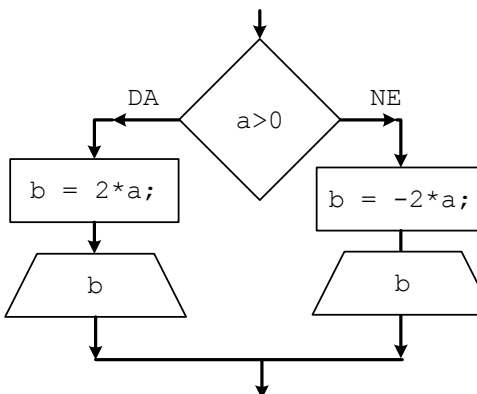
Zadaci - 5a (*if*- i *if-else* iskazi)

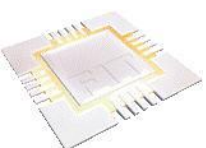
Algoritam predstavljen blok dijagramom

U narednoj tabeli dati su osnovni elementi tok dijagrama:

Naziv	Algoritam (blok dijagram)	C++ code	Opis
ulaz		<code>cin >> a >> b;</code>	vrijednost se unosi sa tastature
izlaz		<code>cout << "Greska";</code>	ispisuje se poruka na ekran
		<code>cout << a << endl << b;</code>	ispisuje se vrijednost varijabli a i b na ekran
izraz		<code>V = a * b;</code>	vrijednost (matematičkog izraza) se dodjeljuje varijabli v
<i>if</i> -iskaz (jednostruki izbor) - opšti primjer -		<pre>if (uslov) { naredba1; naredba2; ... }</pre> <div style="border: 1px solid black; padding: 5px; display: inline-block; margin-top: 10px;"> naredba1; naredba2; ... </div> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-top: 5px; color: blue; font-weight: bold;">Blok</div>	ako je uslov zadovoljen izvršava se Blok (naredbe), inače se preskače Blok



<p><i>if</i>-iskaz (jednostruki izbor) - konkretan primjer -</p>		<pre>if (a>0) { b = a*a; cout << b; ... }</pre> <div data-bbox="885 219 1168 331"> <div>Blok</div> </div>	
<p><i>if-else</i>-iskaz (dvostruki izbor) - opšti primjer -</p>		<pre>if (uslov) { naredba_1a; naredba_2a; ... } else { naredba_1b; naredba_2b; ... }</pre> <div data-bbox="885 721 1168 855"> <div>Blok A</div> </div> <div data-bbox="885 958 1168 1093"> <div>Blok B</div> </div>	<p>ako je uslov zadovoljen izvršava se <i>Blok A</i>, inače se izvršava <i>Blok B</i></p>
<p><i>if-else</i>-iskaz (dvostruki izbor) - konkretan primjer -</p>		<pre>if (a>0) { b = 2*a; cout << b; ... } else { b = -2*a; cout << b; ... }</pre> <div data-bbox="885 1236 1168 1370"> <div>Blok A</div> </div> <div data-bbox="885 1473 1168 1608"> <div>Blok B</div> </div>	



Zadatak 31:

Napravite algoritam i predstavite ga dijagramom toka za sljedeći zadatak (tj. **nemojte pisati C++ kôd**):

Napravite program u kome ćete od korisnika zahtijevati unos dva broja. Neka to budu varijable *a* i *b* tipa *integer*. Program treba, ako je prvi broj veći od drugog, ispisati poruku: 'prvi broj je veći od drugog', inače treba ispisati 'prvi broj nije veći od drugog'. Koristite jedan if-else iskaz (dvostruki if-iskaz). Na kraju, program treba ispisati poruku 'Kraj programa' pomoću naredbe koja se nalazi izvan if-else iskaza.

Rješenje se nalazi na stranici 43.

Napomena:

U dijagramu toka algoritma koji predstavlja rezultat faze dizajna nije potrebno navoditi deklaracije varijabli. Varijable se deklariraju u programu a na osnovu kvalifikacije podataka u fazi dizajna. Za razliku od *source* koda u kome, prije prve upotrebe neke varijable, moramo deklarirati varijablu i tek tada joj možemo dodijeliti neku vrijednost (npr. `int a = 5;`), u algoritmu ćete odmah koristiti varijablu. Npr. nacrtate jedan pravougaonik i u njega napišete taj izraz (`a = 5`). Možete zaključiti, da u algoritmu ne navodimo o kojem se tipu podatka radi.

Ako na osnovu algoritma (a nemate ostale elemente iz faze dizajna) želite napisati *source code* onda ćete na osnovu logike morati odabrati koji je tip podatka za neku varijablu (koja je korištena u algoritmu, npr. `godiste = 84`) najpogodniji (npr.: `int godiste = 84;`).

U dijagramu toka algoritma se, takođe, **ne crtaju** 'pozdravne' *cout*-poruke, kao: „Zdravo, Unesite broj.” ili sl., jer se podrazumijevaju i nepotrebne su u dijagramu toka.

U prvih nekoliko zadataka sa if-iskazima rješavajte samo pomoću dijagrama toka. Kasnije će, nakon što dobro savladate dijagram toka, slijediti zadaci u kojima ćete pisati C++ kôd na osnovu prethodno nacrtani dijagrama. Primjetit ćete da najteži dio u rješavanju težih zadataka predstavlja crtanje ispravnog dijagram toka.

Zadatak 32:

Napravite algoritam i predstavite ga dijagramom toka za sljedeći zadatak (tj. **nemojte pisati C++ kôd**):

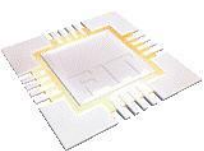
Prepravite prethodni program tako da se naredba za ispis poruke 'Kraj programa' nalazi u if-else-iskazu, tj. jedna naredba za ispis u slučaju 'DA' i jedna naredba za ispis u slučaju 'NE'.

Rješenje se nalazi na stranici 43.

Zadatak 33:

Napravite algoritam i predstavite ga dijagramom toka za sljedeći zadatak (tj. **nemojte pisati C++ kôd**):

Prepravite prethodni program tako da umjesto jednog if-else-iskaza (dvostrukog) koristite dva if-iskaza (jednostruka), Naredba za ispis neka se nalazi unutar if-iskaza.



Rješenje se nalazi na stranici 43.

Zadatak 34:

Napravite algoritam i predstavite ga dijagramom toka za sljedeći zadatak:

Napravite program koji će računati vrijednost $y = f(x)$ i ispisati je na ekran. Vrijednost x se trebati učitati sa tastature od korisnika.

$$y = \begin{cases} \sqrt{x+1} & \text{za } x > 0 \\ 0 & \text{za } x = 0 \\ x^2 & \text{za } x < 0 \end{cases}$$

Primjer rada programa:

unos $x = 8 \Rightarrow$ ispis $y = 3$
unos $x = 0 \Rightarrow$ ispis $y = 0$
unos $x = -4 \Rightarrow$ ispis $y = 16$

Program riješite pomoću tri if-iskaza (jednstruki if-iskaz). Sva tri iskaza trebaju biti povezani jedan ispod drugog.

Ako ispravno postavite uslove onda će se tačno jedan if-iskaz izvršiti jer, za bilo koju vrijednost koju varijable x može imati, uvijek će biti zadovoljen jedan od tri uslova.

Rješenje se nalazi na stranici 43.

Zadatak 35:

Napravite algoritam i predstavite ga dijagramom toka za sljedeći zadatak:

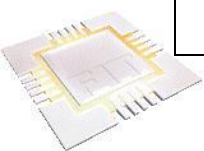
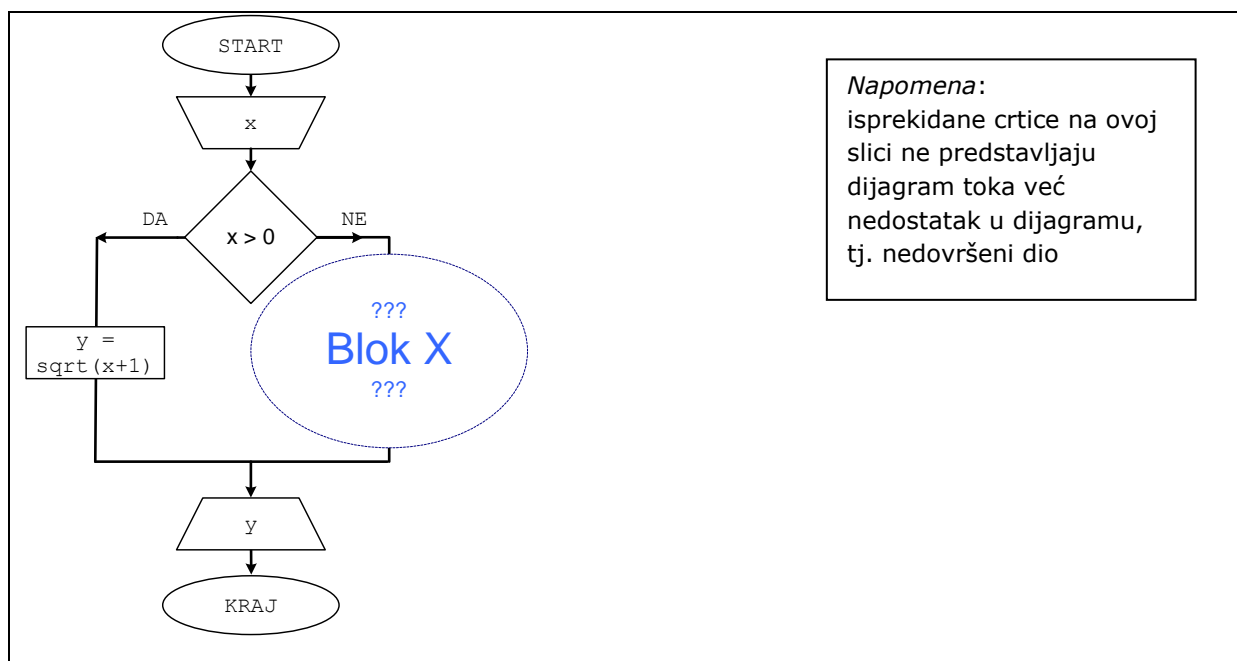
Prepravite prethodni program tako da umjesto tri if-iskaza koristite dva if-else-iskaza.

Rješenje se nalazi na stranici 44.

Mala pomoć:

U prvi if-else-iskaz upišete neki od tri uslova. Neka to bude $x > 0$. Ako je uslov zadovoljen izvršit će se izraz $y = \sqrt{x+1}$.

A šta će se izvršiti ako uslov nije zadovoljen? – Trebamo izračunati y . Mi još ne znamo po kojoj ćemo formuli izračunati vrijednost y . Tu situaciju možemo prikazati:



Kako riješiti Blok X ?

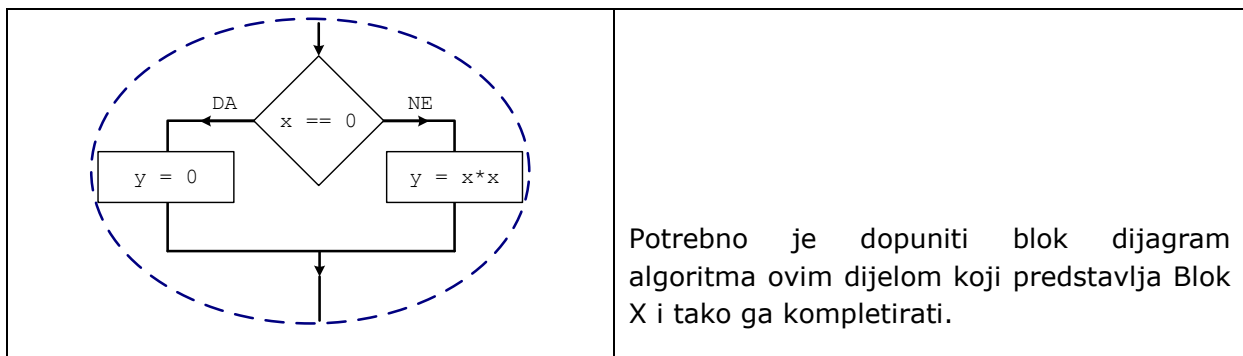
Blok X treba...

- a) ...ako je $x < 0$ izvršiti naredbu: $y = x^2$
- b) ...ako je $x = 0$ izvršiti naredbu: $y = 0$

Koju će naredbu izvršiti Blok X ako je $x > 0$?

- ovo je nemoguć slučaj, jer će se Blok X izvršavati samo ako je $x \leq 0$, jer se o tome brine prvi *if*-iskaz

Znači, u Blok X treba ubaciti jedan *if-e/se*-iskaz sa nekim od gore dva navedena uslova (uslov a i uslov b). Neka to bude $x = 0$. Ostatak je jasan:



Zadatak 36:

Napravite algoritam i predstavite ga dijagramom toka za sljedeći zadatak:

Prepravite prethodni program tako da računa vrijednost y po slijedećoj matematičkoj formuli:

$$y = \begin{cases} \sqrt{x+1} & \text{za } x > 0 \\ z \cdot \sqrt{2} & \text{za } x = 0 \\ x^2 & \text{za } x < 0 \end{cases}$$

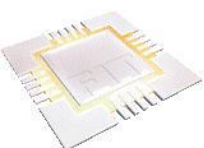
Npr.:

unos x = 8	⇒	ispis y = 3
unos x = 0	⇒	unos z = 8 ⇒ ispis y = 11.313
unos x = -4	⇒	ispis y = 16

Ako je korisnik za x unio broj 0, program treba samo tada tražiti unos i broja za z .

Dodatni zahtjev: Ako korisnik unese broj koji nije pozitivan (tj. da je manji od nule ili jednak nuli) program treba ispisati poruku: 'Upozorenje: Niste unijeli pozitivan broj.'

Rješenje se nalazi na stranici 44.



Zadatak 37:

Napravite algoritam i predstavite ga dijagramom toka za sljedeći zadatak:

Napravite program koji će tražiti od korisnika da unese broj n . Ako je n paran broj (tj. djeljiv sa 2), program će zahtijevati unos boja m . Program treba, u tom slučaju izračunati vrijednost $w = \frac{n}{m}$ i ispisati je na ekran. A ako je broj n neparan, program treba izračunati vrijednost...

$$y = \begin{cases} \sqrt{n} & \text{za } n \geq 0 \\ a^2 \cdot b & \text{za } n < 0 \end{cases}$$

...i ispisati tu vrijednost na ekran.

Napomena:

Program treba, od korisnika, tražiti unos brojeva a i b samo ako bude potrebe.

Pazite! U zadatku nije rečeno, ali se podrazumijeva, da vrijednost m mora biti različita od nule jer nije moguće dijeliti sa nulom. U slučaju da je broj m jednak nuli program treba ispisati neku poruku, kao: 'Greška: Nije moguće dijeliti sa 0' ili sl.

Rješenje se nalazi na stranici 44.

Mala pomoć:

Kad je broj p djeljiv sa brojem q ?

- Ako je vrijednost ostatka kod dijeljenja brojeva p i q jednaka 0.

Kako izračunati vrijednost ostatka kod dijeljenja dva broja?

- Pogledajte zadatak 30 i zaključite:

- Ako je vrijednost $n\%2$ jednaka 0, onda je n paran broj.
 - inače (ako je vrijednost $n\%2$ različita od 0) je n neparan broj.
- Ako je vrijednost $n\%2$ jednaka 1, onda je n neparan broj.
 - inače (ako je vrijednost $n\%2$ različita od 1) je n paran broj.

Zadatak 38:

Pomoću tabele (na strani 30) pretvorite riješeni algoritam za zadatak br. 31 u C++ kod.

Rješenje se nalazi na stranici 45.

Zadatak 39:

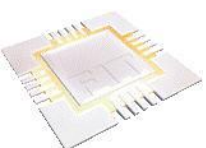
Pretvorite riješeni algoritam za zadatak br. 32 u C++ kod.

Rješenje se nalazi na stranici 45.

Zadatak 40:

Pretvorite riješeni algoritam za zadatak br. 33 u C++ kod.

Rješenje se nalazi na stranici 46.



Zadatak 41:

Pretvorite riješeni algoritam za zadatak br. 34 u C++ kod.

Rješenje se nalazi na stranici 46.

Zadatak 42:

Pretvorite riješeni algoritam za zadatak br. 35 u C++ kod.

Rješenje se nalazi na stranici 47.

Za pisanje koda sa ugniježđenim *if*-iskazima (*if*-iskaz u *if*-iskazu) preporučujemo da prvo napišite prazan osnovni *if*-iskaz (onaj čiji će uslov biti prvi provjeren) sa praznim vitičastim zagradama za oba slučaja („DA” i „NE”), a zatim da pomjerite kursor u *if*-iskaz i u njega pišete naredbe (npr.: *cout*- i *cin*-naredbe, drugi *if*-iskazi, izrazi, ...).

Preporučujemo da sve naredbe u C++ kodu budu uvučene (pomoću tabulatora) u odnosu na vitičaste zagrade kojoj pripadaju:

```
if (uslov)
{
    naredba_1;
    naredba_2;
    if (uslov)
    {
        naredba_3;
    }
}
else
{
    naredba_4;
}
```

„*if*”, njegov „*else*” i njihove vitičaste zagrade (ako ih imaju) trebale bi biti podjednako poravnate sa lijeve strane.

Zadatak 43:

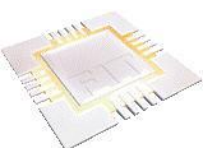
Pretvorite riješeni algoritam za zadatak br. 36 u C++ kod.

Rješenje se nalazi na stranici 47.

Zadatak 44:

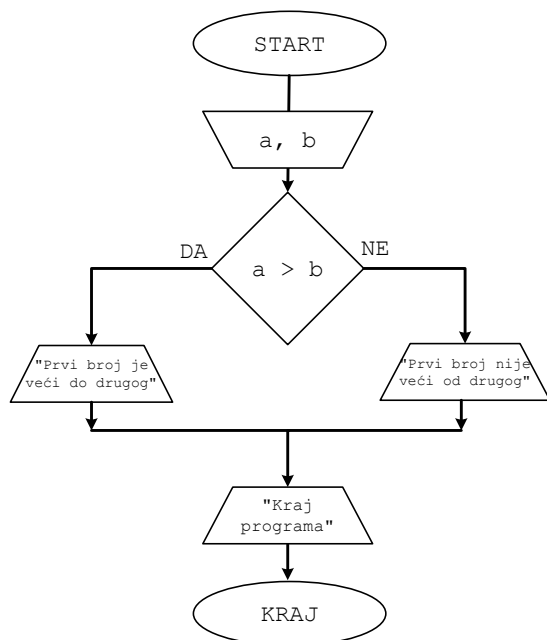
Pretvorite riješeni algoritam za zadatak br. 37 u C++ kod.

Rješenje se nalazi na stranici 48.

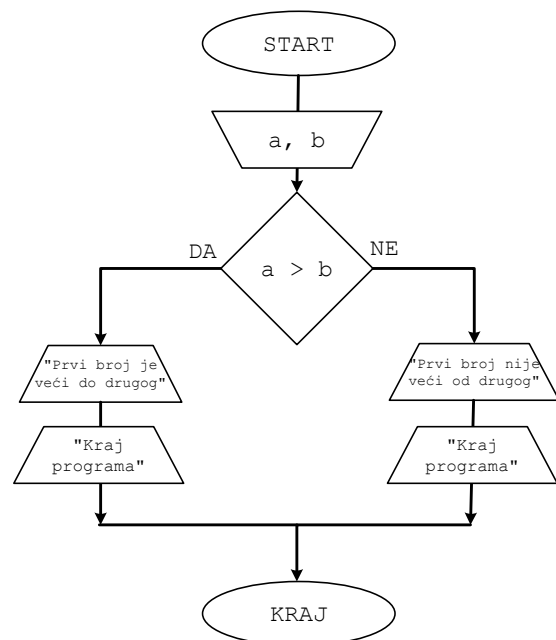


Rješenja

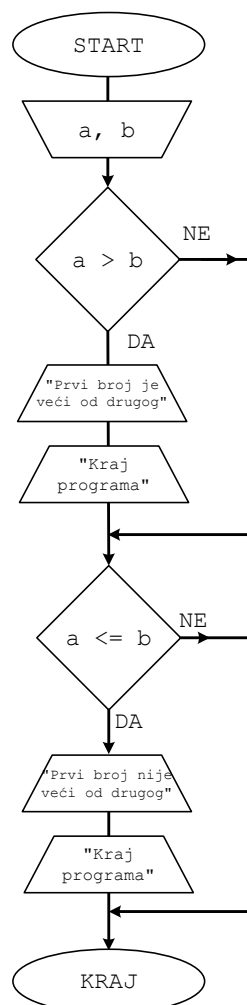
Rješenje zadatka br. 31:



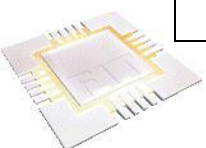
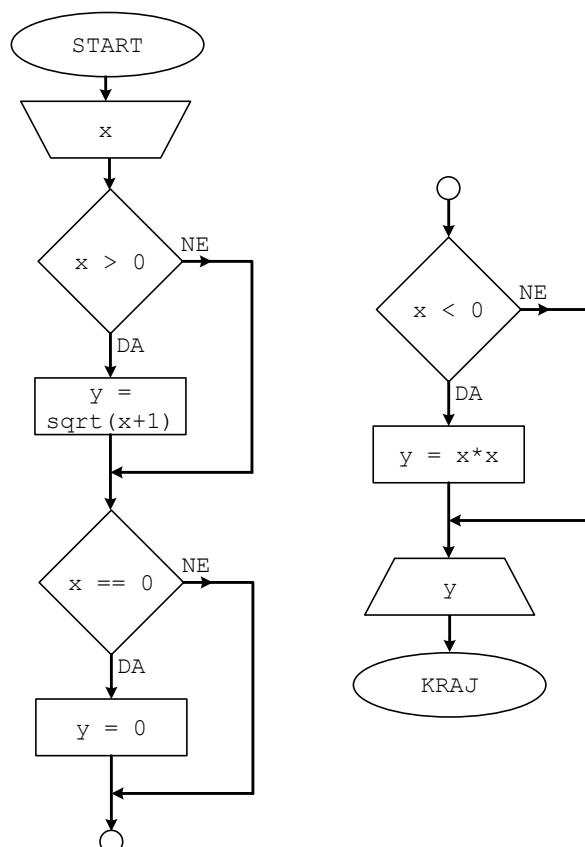
Rješenje zadatka br. 32:



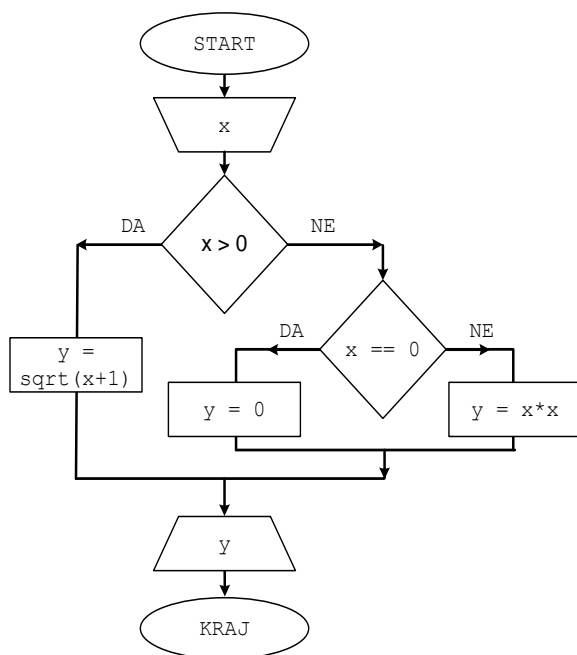
Rješenje zadatka br. 33:



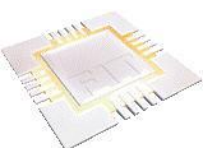
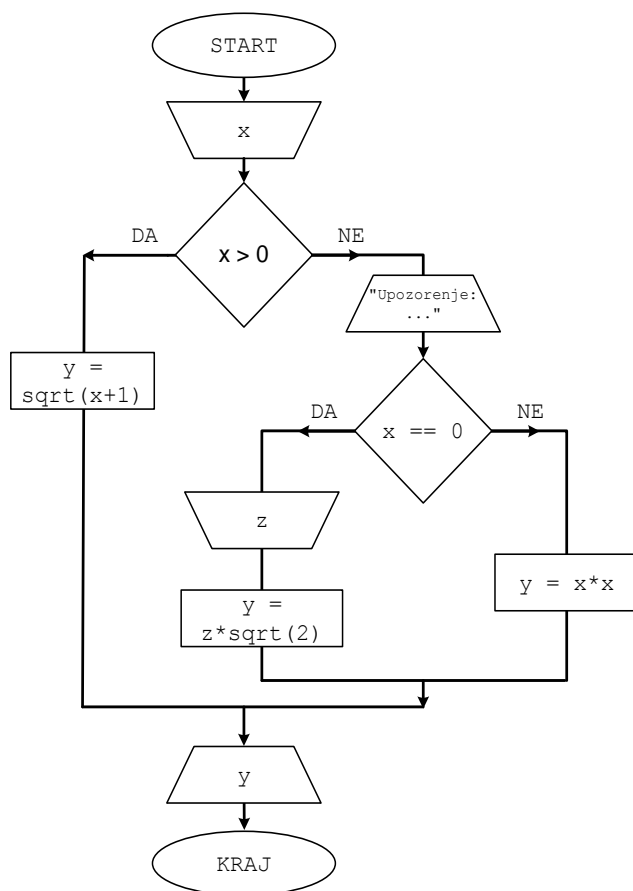
Rješenje zadatka br. 34:



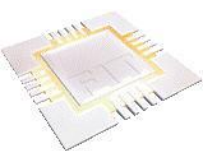
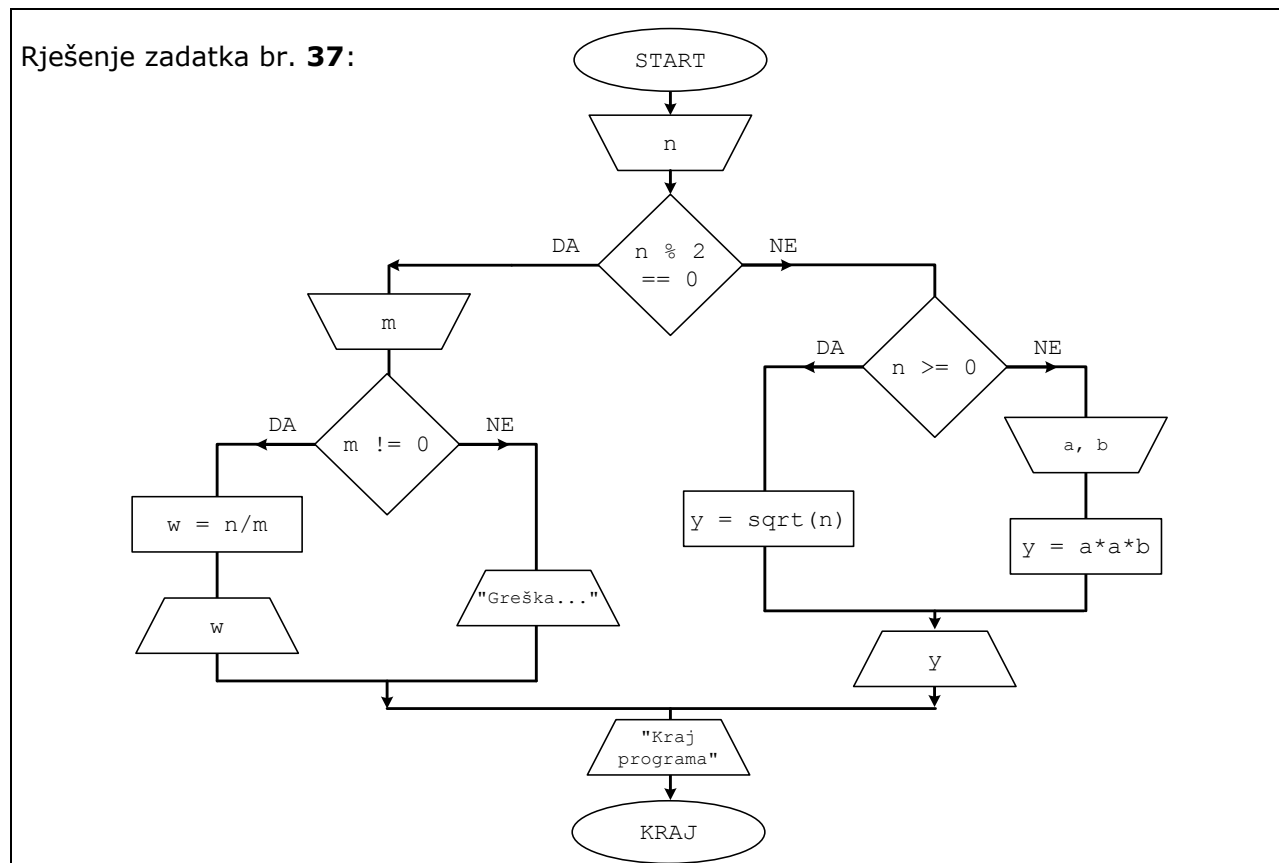
Rješenje zadatka br. 35:



Rješenje zadatka br. 36:



Rješenje zadatka br. 37:



Rješenje zadatka br. 38 (31):

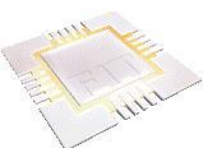
```
1:  #include <iostream>
2:  using namespace std;
3:
4:  void main()
5:  {
6:      int a, b;          // deklaracija varijabli se ne crta u algoritmu
7:      cout << "Unesite dva broja \n"; // pozdravna poruka se ne crta u algor.
8:      cin >> a >> b;
9:
10:     if (a > b)          // ovdje se ne stavlja tačka-zarez!
11:     {
12:         cout << "Prvi broj je veci od drugog \n";
13:     }
14:     else                // ovdje se ne stavlja tačka-zarez!
15:     {
16:         cout << "Prvi broj nije veci od drugog \n";
17:     }
18:
19:     cout << "Kraj programa \n";
20: }
```

Ako se u bloku nalazi samo jedna naredba, onda se mogu izostaviti vitičaste {} zagrade. To smo mogli učiniti i u prethodnom rješenju:

```
1:  #include <iostream>
2:  using namespace std;
3:
4:  void main()
5:  {
6:      int a, b;
7:      cout << "Unesite dva broja \n";
8:      cin >> a >> b;
9:
10:     if (a > b)
11:         cout << "Prvi broj je veci od drugog \n";
12:     else
13:         cout << "Prvi broj nije veci od drugog \n";
14:
15:     cout << "Kraj programa \n";
16: }
```

Rješenje zadatka br. 39 (32):

```
1:  #include <iostream>
2:  using namespace std;
3:
4:  void main()
5:  {
6:      int a, b;
7:      cout << "Unesite dva broja \n";
8:      cin >> a >> b;
9:
10:     if (a > b)
11:     {
12:         cout << "Prvi broj je veci od drugog \n";
13:         cout << "Kraj programa \n";
14:     }
15:     else
16:     {
17:         cout << "Prvi broj nije veci od drugog \n";
18:         cout << "Kraj programa \n";
19:     }
20: }
```

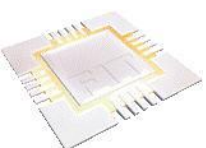


Rješenje zadatka br. **40** (33):

```
1:  #include <iostream>
2:  using namespace std;
3:
4:  void main()
5:  {
6:      int a, b;
7:      cout << "Unesite dva broja \n";
8:      cin >> a >> b;
9:
10:     if (a > b)
11:     {
12:         cout << "Prvi broj je veci od drugog \n";
13:         cout << "Kraj programa \n";
14:     }
15:
16:     if (a <= b)
17:     {
18:         cout << "Prvi broj nije veci od drugog \n";
19:         cout << "Kraj programa \n";
20:     }
21: }
```

Rješenje zadatka br. **41** (34):

```
1:  #include <iostream>
2:  using namespace std;
3:  #include <math.h>
4:
5:  void main()
6:  {
7:      float x, y;
8:      cout << "Unesite x: ";
9:      cin >> x;
10:
11:     if (x > 0)
12:         y = sqrt(x+1);
13:
14:     if (x == 0)
15:         y = 0;
16:
17:     if (x < 0)
18:         y = x * x;
19:
20:     cout << "y = " << y << endl;
21: }
```

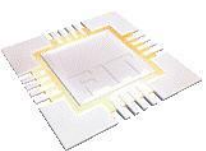


Rješenje zadatka br. **42** (35):

```
1:  #include <iostream>
2:  using namespace std;
3:  #include <math.h>
4:
5:  void main()
6:  {
7:      float x, y;
8:      cout << "Unesite x: ";
9:      cin >> x;
10:
11:      if (x > 0)
12:          y = sqrt(x+1);
13:      else
14:      {
15:          if (x == 0)
16:              y = 0;
17:          else
18:              y = x * x;
19:      }
20:
21:      cout << "y = " << y << endl;
22: }
```

Rješenje zadatka br. **43** (36):

```
1:  #include <iostream>
2:  using namespace std;
3:  #include <math.h>
4:
5:  void main()
6:  {
7:      float x, y;
8:      cout << "Unesite x: ";
9:      cin >> x;
10:
11:      if (x > 0)
12:          y = sqrt(x+1);
13:      else
14:      {
15:          cout << "Upozorenje: Broj nije pozitivan" << endl;
16:
17:          if (x == 0)
18:          {
19:              int z;
20:              cout << "Unesite z: ";
21:              cin >> z;
22:              y = z * sqrt(float(2));
23:          }
24:          else
25:              y = x * x;
26:      }
27:
28:      cout << "y = " << y << endl;
29: }
```



Rješenje zadatka br. 44 (37):

```
1:  #include <iostream>
2:  using namespace std;
3:  #include <math.h>
4:
5:  void main()
6:  {
7:      int n;
8:      cout << "Unesite n: ";
9:      cin >> n;
10:
11:      if (n % 2 == 0)
12:      {
13:          int m; // u zadatku nije rečeno koji tip podatka m mora biti,
14:          cout << "Unesite m: "; // a mi ćemo se odlučiti za int
15:          cin >> m;
16:
17:          if (m != 0)
18:          {
19:              float w; // ovo možemo deklarirati i na početku programa
20:              w = n / float(m);
21:              cout << "w = " << w << endl;
22:          }
23:          else
24:              cout << "Greska: Nije moguće dijeliti sa 0 \n";
25:      }
26:      else
27:      {
28:          float y; // ovo možemo deklarirati i na početku programa
29:
30:          if (n >= 0)
31:          {
32:              y = sqrt(float(n));
33:          }
34:          else
35:          {
36:              int a, b;
37:              cout << "Unesite a i b: ";
38:              cin >> a >> b;
39:              y = a * a * b;
40:          }
41:
42:          cout << "y = " << y << endl;
43:      }
44:
45:      cout << "Kraj programa \n";
46:  }
```

