

Vežba 1

Skup pravila koja se koriste za rešavanje određenog tipa zadatka, zove se algoritam. Svako pojedinačno pravilo zove se algoritamski korak.

Algoritam je uputstvo kako rešiti neki zadatak ili obaviti neki posao. Algoritam svodi celokupan zadatak na rešavanje više jednostavnijih, manjih zadataka. Algoritmi se mogu prikazati dijagramom toka ili pseudo kodom.

Postoje dve vrste dijagrama toka: standardni i strukturirani.

Dijagram toka služi za vizualizaciju (slikovni prikaz algoritma) i pojednostavljivanje zadatka.

Pseudo kod se sastoji od kratkih izraza na govornom jeziku koji opisuju i ukratko objašnjavaju pojedine zadatke algoritma. Pseudo kod liči na programski kod ali nije napisan ni u jednom programskom jeziku i ne može se izvršavati ni na jednom računaru. Pseudo kod, isto kao dijagram toka, služi za opis i pojednostavljivanje programa.

Zadatak: Skuvati kafu.



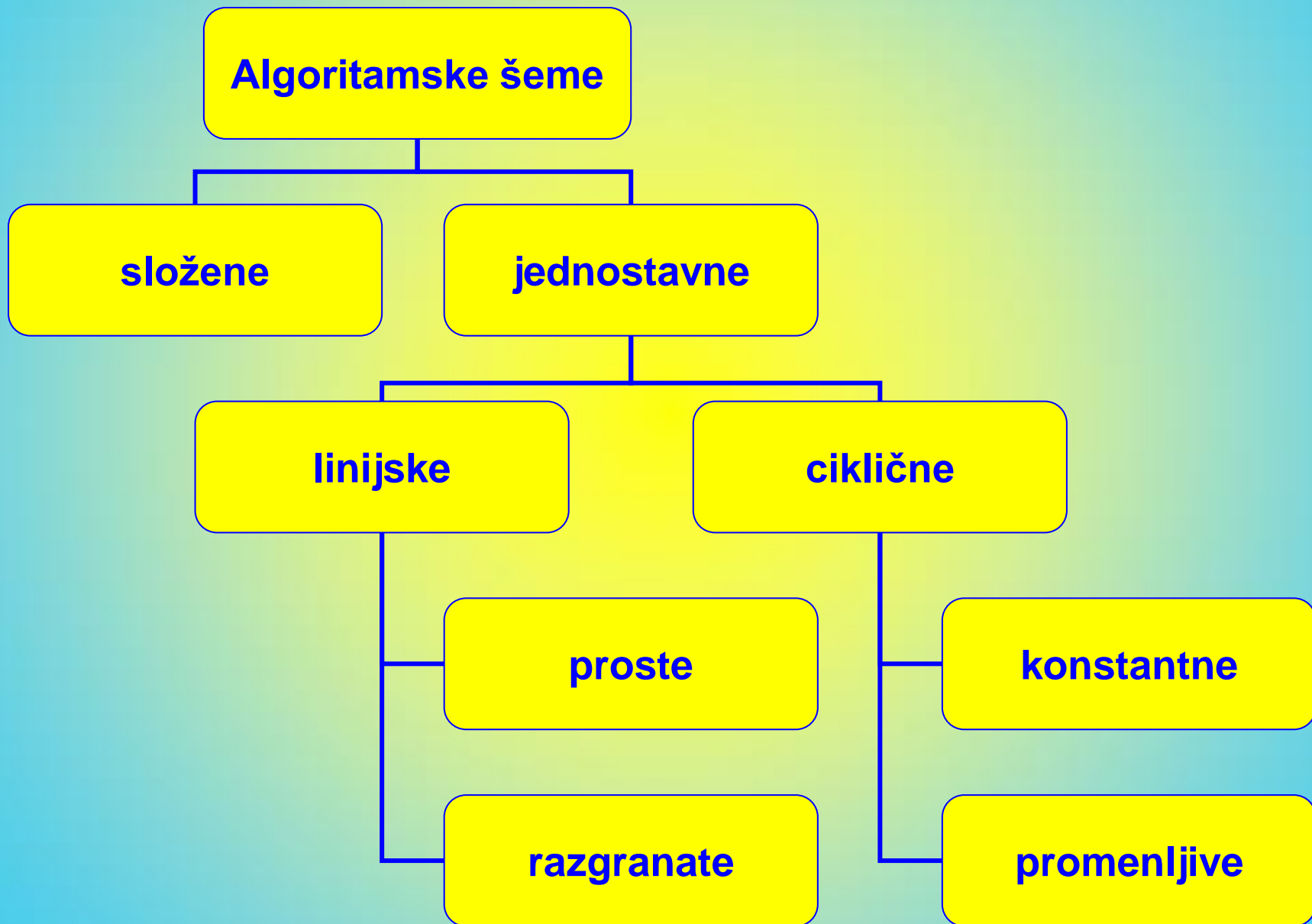
Dijagram toka



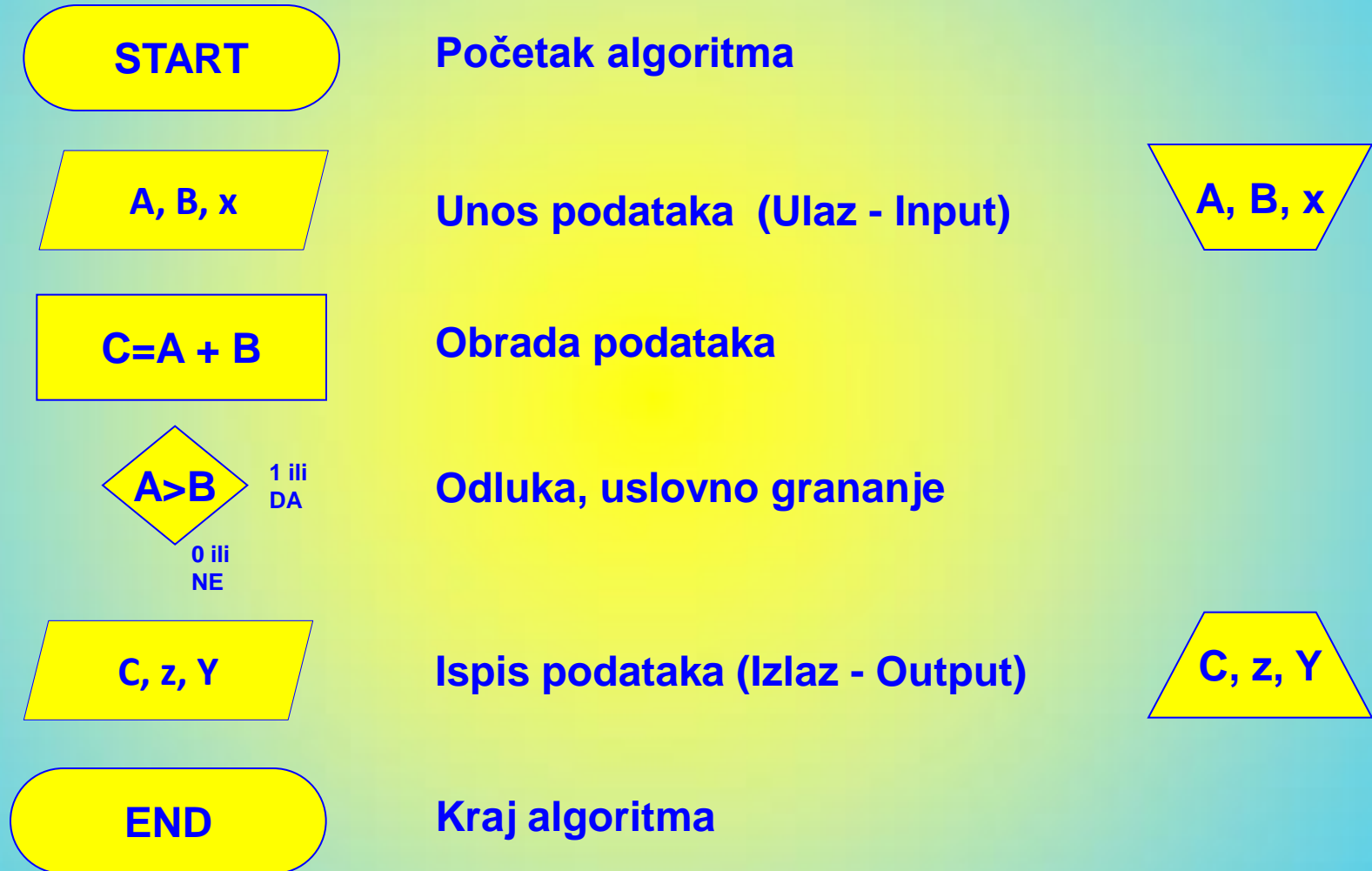
Pseudo kod

1. Zagrejati vodu
2. Staviti kesicu čaja
3. Sačekati 5 minuta
4. Poslužiti

Vežba 1 – Vrste aloritamskih šema

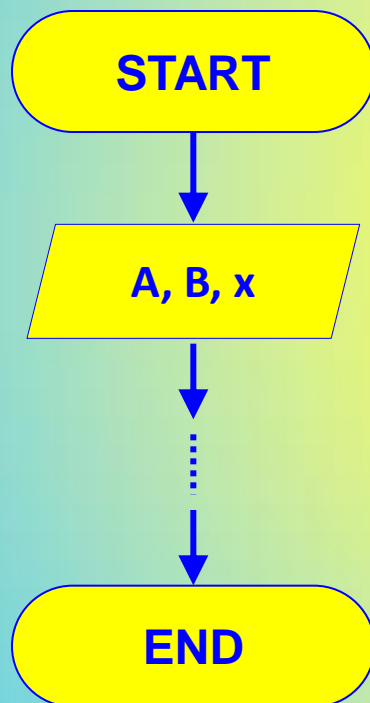


Vežba 1 - Osnovni simboli

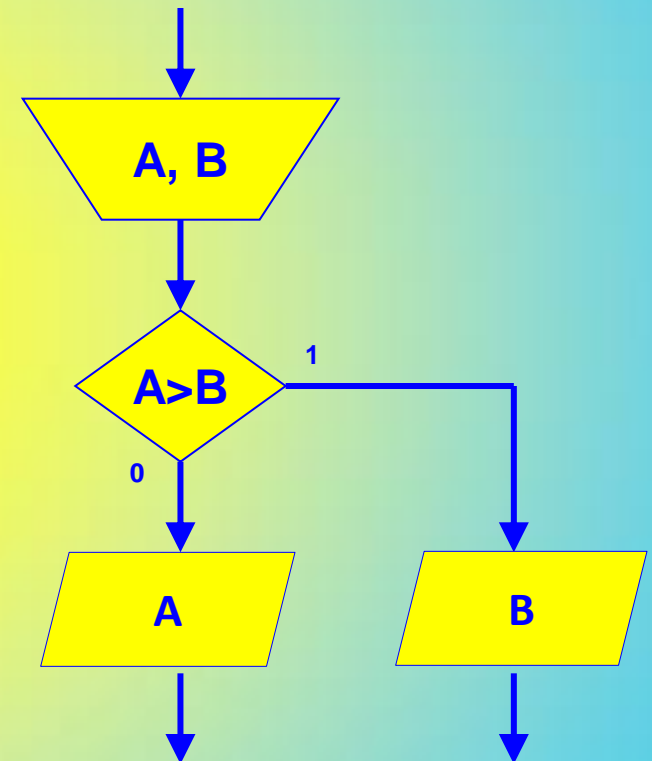


Vežba 1 – Spajanje simbola

Vrši se pomoću STRELICE



ili



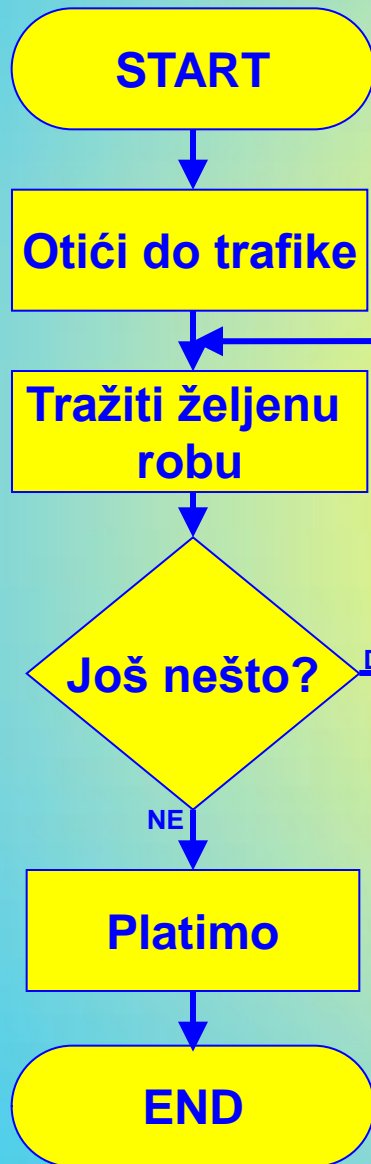
Vežba 1 – Primer iz svakodnevnog života

Zadatak: Kako najlakše kupiti novine?

Radi se korak po korak:

1. Otići do trafike
2. Tražiti željenu robu
3. Prodavačica pita da li želimo još nešto?
4. Ako želimo - tražimo robu, a ako ne pitamo koliko smo dužni
5. Platimo

Vežba 1 – Primer iz svakodnevnog života



Početak algoritma

Obrada, tj. naredba da se ode do trafike

Obrada, tj. naredba da se traži roba

Odluka, da li tražiti još nešto ili ne. Ako treba još nešto, vraćamo se korak više i tražimo robu, a ako ne, onda idemo na sledeći korak.

Obrada, tj. naredba da se plati

Kraj algoritma

Vežba 1

Tipovi podataka

int	ceo broj, 2 ili 4 bajta
char	znak, jedan bajt, jedan bajt, sadrzi jedan znak
float	realan broj, jednostruka tacnost
double	realan broj dvostruke tacnosti
String	niz znakova

Primer:

```
#include <stdio.h>

main() {
    printf("hello world!");
    printf("\n");
}
```

Funkcije printf i scanf

```
scanf("%d", broj1);
printf("%d\t%d\n", broj1, broj2);
```

uvek je prvi argument izmedju " "

%d ceo broj

\t tab izmedju

\n novi red

Svaka % konstrukcija je u paru sa argumentom koji sledi.

Pravila:

%d štampaj kao ceo broj

%6d štampaj kao ceo broj širok najviše 6 znakova

%f štampaj kao realan broj

%6f štampaj kao realan broj širok najviše 6 znakova

%.2f štampaj kao realan broj sa dve decimale

%6.2f štampaj kao realan broj širok najviše 6 znakova a od toga 2 iza decimalne tačke

%c karakter

%s string

%x heksadecimalni broj

%% je procenat

Vežba 1 – primer 1

Sledeći primer prikazuje unos celog broja koristeći `scanf("%d", &x)`

```
#include <stdio.h>
```

```
main() {  
    int x;  
    printf("Unesi ceo broj : ");  
    /* Obratiti paznju na znak & (operator uzimanja adrese)  
    pre imena promenljive u funkciji scanf */  
    scanf("%d",&x);  
    /* U funkciji printf nije potrebno stavljati & */  
    printf("Uneli ste broj %d\n", x);  
}
```


Vežba 1 – primer 2

Program sabira dva uneta cela broja

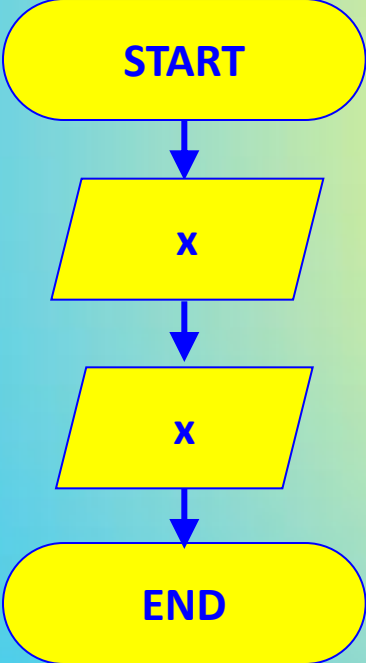
```
#include <stdio.h>
```

```
main() {  
    int a, b, c;  
    printf("Unesi prvi broj : ");  
    scanf("%d", &a);  
    printf("Unesi drugi broj : ");  
    scanf("%d", &b);  
    c = a + b;  
    printf("%d + %d = %d\n", a, b, c);  
}
```

Vežba 1

Zadatak 1.

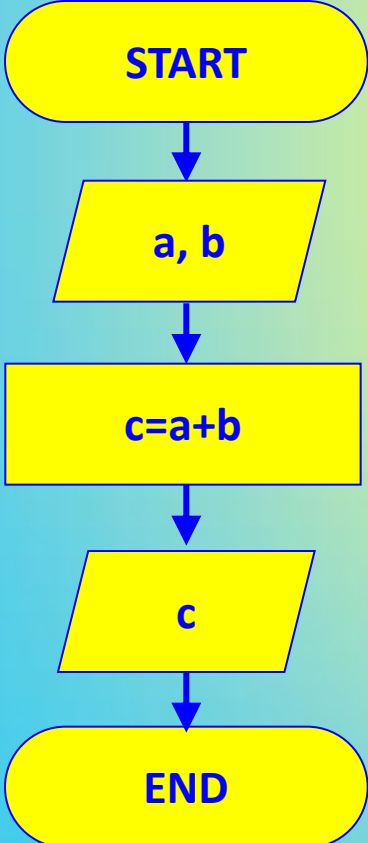
Napisati algoritam za učitavanje i ispis broja.

Dijagram toka	Tekstualni algoritam (pseudo kod)	Programski kod
 <pre>graph TD; START([START]) --> I1[/x/]; I1 --> O1[/x/]; O1 --> END([END]);</pre>	<ol style="list-style-type: none">1. Ulaz-učitati broj x2. Izlaz-ispisati broj x	<pre>#include <stdio.h> main() { int x = 5; printf(„Ispis broja: %d”, x); } main() { int x; printf("Unesite broj x: "); scanf("%d", &x); printf(„Ispis unetog broja: %d”, x); }</pre>

Vežba 1

Zadatak 2.

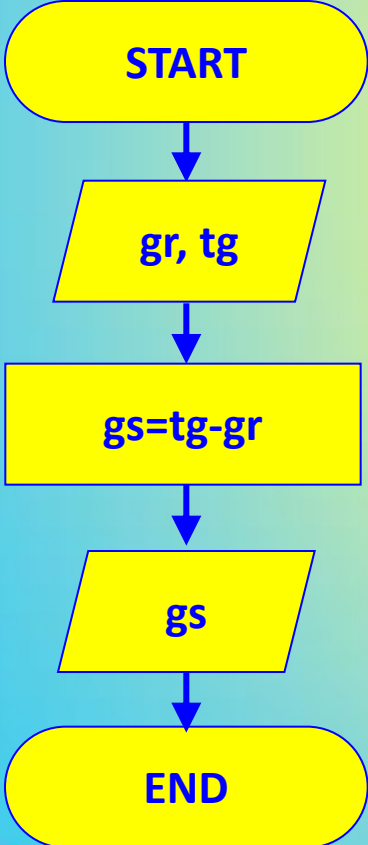
Napisati algoritam i program koji sabira dva uneta broja.

Dijagram toka	Tekstualni algoritam (pseudo kod)	Programski kod
 <pre>graph TD; START([START]) --> Input[/a, b/]; Input --> Process[c=a+b]; Process --> Output[/c/]; Output --> END([END]);</pre>	<ol style="list-style-type: none">1. Ulaz-učitati dva broja2. Obrada-izračunati zbir dva uneta broja3. Izlaz-ispisati zbir dva uneta broja	<pre>#include <stdio.h> main() { int a, b, c; printf("Unesite broj A: "); scanf("%d", &a); printf("\nUnesite broj B: "); scanf("%d", &b); c=a+b; printf("\nZbir brojeva A i B je: %d", c); }</pre>

Vežba 1

Zadatak 3.

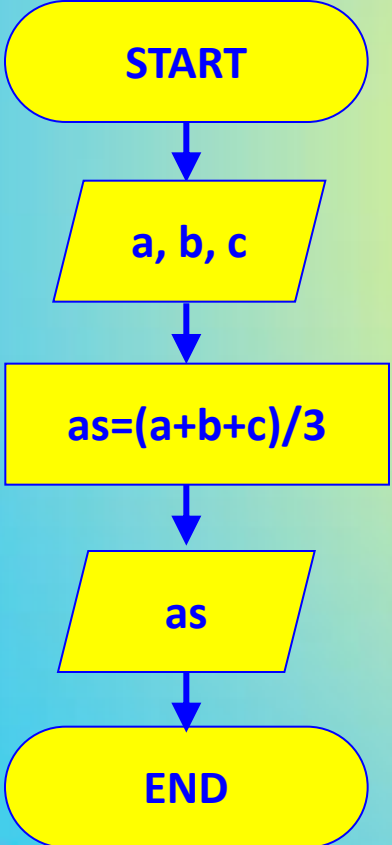
Napisati algoritam i program koji računa godine starosti.

Dijagram toka	Tekstualni algoritam (pseudo kod)	Programski kod
 <pre>graph TD; START([START]) --> Input[/gr, tg/]; Input --> Process[gs=tg-gr]; Process --> Output[/gs/]; Output --> END([END]);</pre>	<ol style="list-style-type: none">1. Ulaz-učitati godinu rođenja i trenutnu godinu2. Obrada-izračunati godine starosti3. Izlaz-ispisati godine starosti	<pre>#include <stdio.h> main() { int gr, tg, gs; printf("\nUnesite godinu vaseg rodjenja: "); scanf("%d", &gr); printf("\nUnesite trenutnu godinu: "); scanf("%d", &tg); gs=tg-gr; printf("\nVi imate %d godina \n", gs); }</pre>

Vežba 1

Zadatak 4.

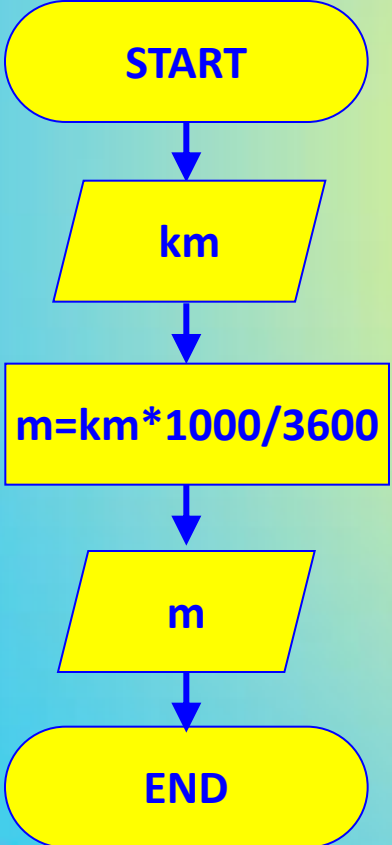
Napisati algoritam i program za pronalaženje aritmetičke sredine tri uneta broja.

Dijagram toka	Tekstualni algoritam (pseudo kod)	Programski kod
 <pre>graph TD; START([START]) --> Input[/a, b, c/]; Input --> Process[as=(a+b+c)/3]; Process --> Output[/as/]; Output --> END([END]);</pre>	<ol style="list-style-type: none">1. Ulaz-učitati tri broja2. Obrada-izračunati aritmetičku sredinu3. Izlaz-ispisati aritmetičku sredinu	<pre>#include <stdio.h> main() { double a, b, c, as ; printf("Unesi broj A: "); scanf("%lf", &a); printf("Unesi broj B: "); scanf("%lf", &b); printf("Unesi broj C: "); scanf("%lf", &c); as=(a+b+c)/3; printf("Aritmetička sredina je: %.2f", as); }</pre>

Vežba 1

Zadatak 5.

Napisati algoritam za pretvaranje km/h u m/s.

Dijagram toka	Tekstualni algoritam (pseudo kod)	Programski kod
 <pre>graph TD; START([START]) --> km[/km/]; km --> process[m=km*1000/3600]; process --> m[/m/]; m --> END([END]);</pre>	<ol style="list-style-type: none">1. Ulaz-učitati brzinu u km/h2. Obrada-izračunati brzinu u m/s3. Izlaz-ispisati brzinu u m/s	<pre>#include <stdio.h> main() { double km, m ; printf("Brzinu u km/h: "); scanf("%lf", &km); m=km*1000/3600; printf("Brzina u m/s je: %.2f", m); }</pre>