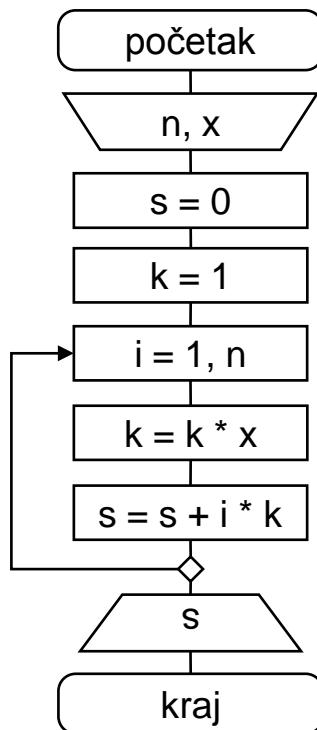


PETLJE

- Napisati algoritam i program koji za uneti ceo broj n i realan broj x izračunava S na sledeći način

$$S = \sum_{i=1}^n ix^i$$

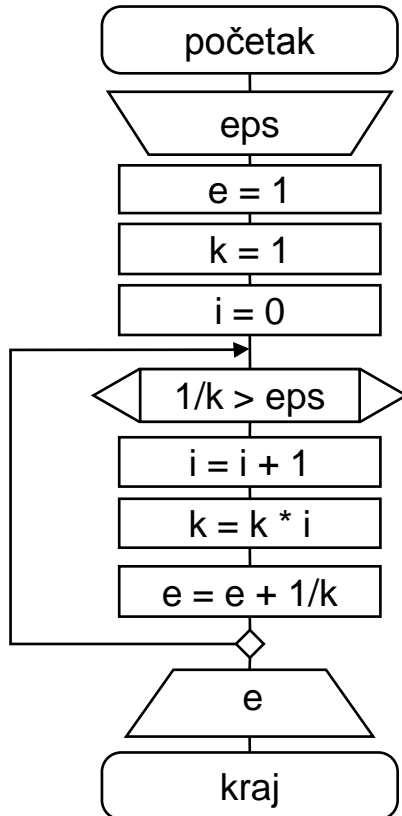


```
Program OP6_1;  
Var i,n:integer;  
    x,s,k:real;  
Begin  
    readln(n,x);  
    s:=0;  
    k:=1;  
    for i:=1 to n do begin  
        k:=k*x;  
        s:=s+i*k  
    end;  
    writeln(s:10:6)  
End.
```

PETLJE

- Napisati algoritam i program koji za zadatu tačnost **eps** izračunava Ojlerov broj **e**

$$e = \sum_{i=0}^{\infty} \frac{1}{i!}$$

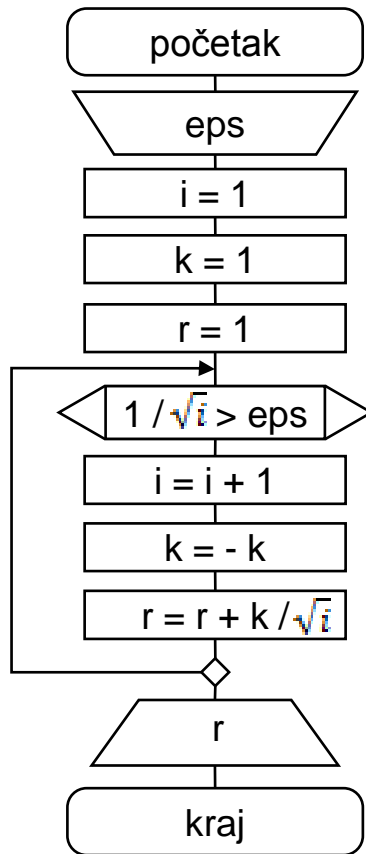


```
Program OP6_2;  
Var i:integer;  
    k,e,eps:real;  
Begin  
    readln(eps);  
    e:=1;  
    k:=1;  
    i:=0;  
    while 1/k>eps do begin  
        i:=i+1;  
        k:=k*i;  
        e:=e+1/k  
    end;  
    writeln(e:10:6)  
End.
```

PETLJE

- Napisati algoritam i program koji za zadatu tačnost **eps** izračunava broj **R**

$$R = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{\sqrt{i}}$$

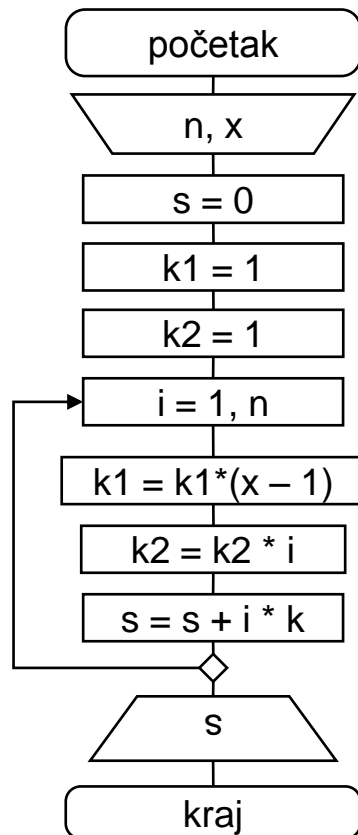


```
Program OP6_3;  
Var r,eps,i,k:real;  
Begin  
  readln(eps);  
  i:=1;  
  k:=1;  
  r:=1;  
  while 1/sqrt(i) > eps do begin  
    i:=i+1;  
    k:=-k;  
    r:=r+k/sqrt(i)  
  end;  
  writeln(r:10:6)  
End.
```

PETLJE

- Napisati algoritam i program koji za uneti ceo broj n ($n > 0$) i realan broj x izračunava S na sledeći način

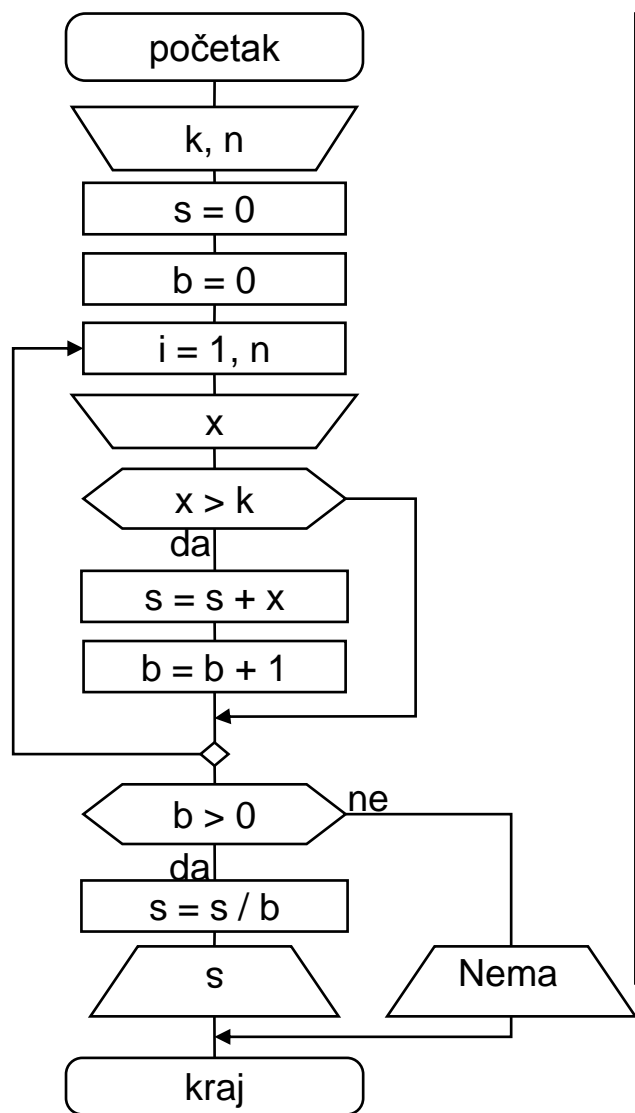
$$S = \sum_{i=1}^n \frac{(x-1)^i}{i!}$$



```
Program OP6_4;  
Var i,n:integer;  
    s,k1,k2,x:real;  
Begin  
    readln(n,x);  
    s:=0;  
    k1:=1;  
    k2:=1;  
    for i:=1 to n do begin  
        k1:=k1*(x-1);  
        k2:=k2*i;  
        s:=s+k1/k2  
    end;  
    writeln(s:10:6)  
End.
```

PETLJE

- Napisati algoritam i program koji za uneti broj **k** i **n** unetih celih brojeva, izračunava srednju vrednost onih brojeva koji su veći od broja **k**.



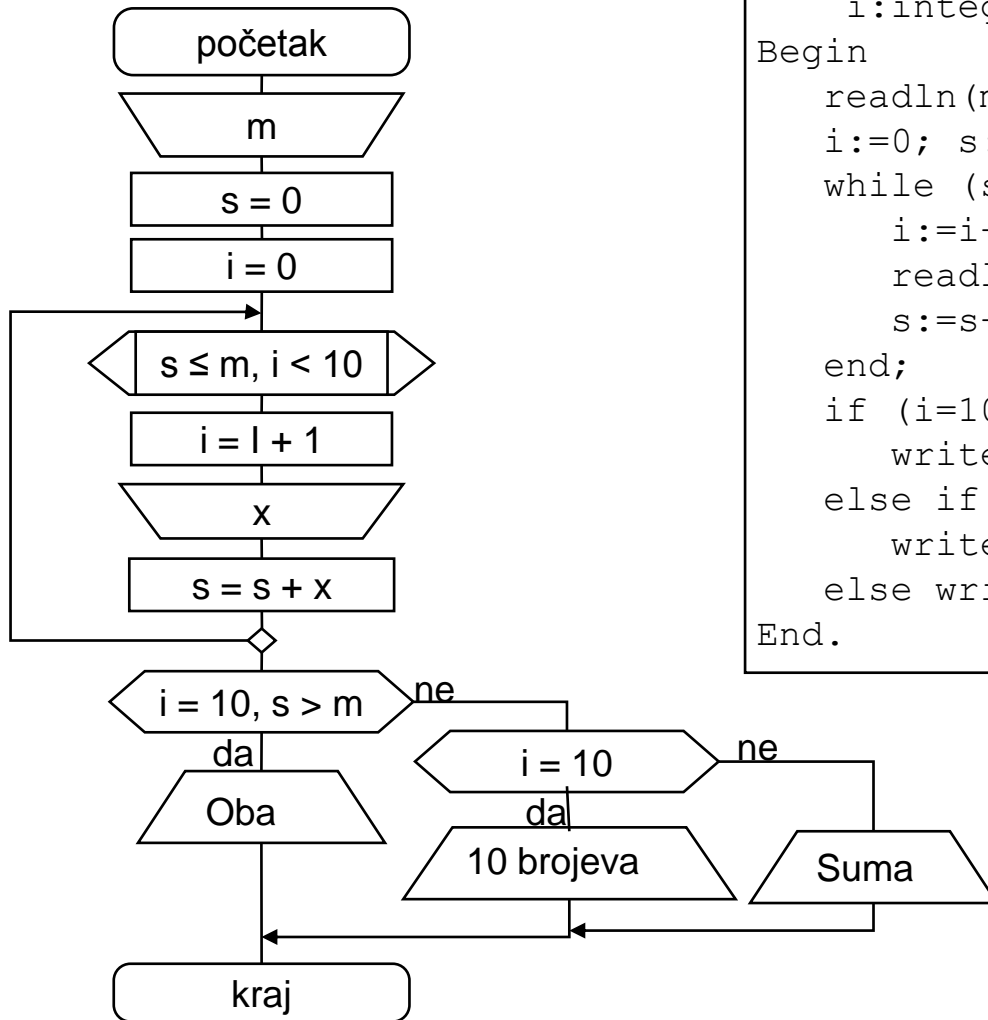
```

Program OP6_5;
Var i,n,b:integer;
    k,s,x:real;
Begin
  readln(k,n);
  s:=0;
  b:=0;
  for i:=1 to n do begin
    readln(x);
    if x>k then begin
      s:=s+x;
      b:=b+1;
    end;
  end;
  if b>0 then begin
    s:=s/b;
    writeln(s:6:2)
  end
  else
    writeln('Nema brojeva vecih od ',k:6:2)
End.

```

PETLJE

- Napisati algoritam i program u kome se najpre unosi realan broj **m**, a zatim se unose realni brojevi sve dok zbir unetih brojeva ne postane veći od **m** ili dok se ne unese 10 brojeva. Na izlazu ispisati koji je kriterijum za izlaz iz petlje ispunjen.



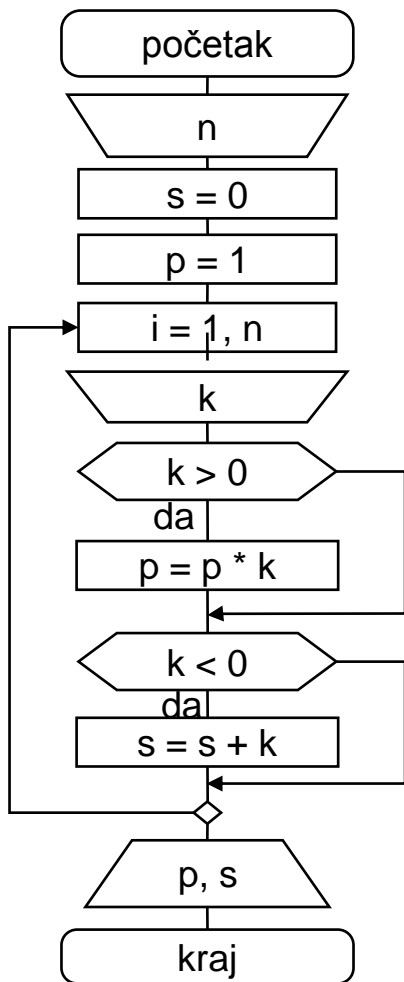
```

Program OP6_6;
Var m,x,s:real;
    i:integer;
Begin
  readln(m);
  i:=0; s:=0;
  while (s<=m) and (i<10) do begin
    i:=i+1;
    readln(x);
    s:=s+x
  end;
  if (i=10) and (s>m) then
    writeln('Ispunjena su oba uslova')
  else if i=10 then
    writeln('Uneto je 10 brojeva')
  else writeln('Suma je veca od ',m:6:2)
End.

```


PETLJE

- Napisati algoritam i program koji za **n** unetih celih brojeva računa proizvod pozitivnih i zbir negativnih.



```

Program OP6_7;
Var i,n,s,k,p:integer;
Begin
  readln(n);
  s:=0;
  p:=1;
  for i:=1 to n do begin
    readln(k);
    if k>0 then p:=p*k;
    if k<0 then s:=s+k;
  end;
  writeln('Proizvod pozitivnih je ',p);
  writeln('Zbir negativnih je ',s)
End.

```

DOMAĆI 5

- Napisati algoritam i program kojim se određuje n -ti član Fibonačijevog niza: 1, 1, 2, 3, 5, 8, 13, ...
- Napisati algoritam i program u kome se unose celi brojevi i računa njihov zbir sve dok se ne unese 0. Štampati dobijeni zbir i koliko je brojeva uneto.
- Dve tačke realne prave nazivamo bliskim ako rastojanje između njih ne prelazi 10. Napisati program kojim se učitavaju koordinate tačaka, dok su sve međusobno bliske. Program treba da ispiše broj učitanih tačaka.

Primer: ulaz: 3 -4 5 0 7 izlaz: 5