

Zadatak 22

type [Zadatak22/tablica.m](#)

```
% tablica.m
X=0:0.1:0.5;
F=exp(X)-2*(X-1).^2;
%Test: [I,Y,x]=nula(0,0.5,1e-5,15) %zadovoljena tacnost tol
%Test: [I,Y,x]=nula(0,0.5,1e-5,4) %zadovoljen max br iteracija iterM
%Test: [I,Y,x]=nula(0,0.5,1e-5,6) %zadovoljena je tacnost i dostignut max
%br iteracija iterM

% drugi primer
% X=0.5:0.1:1;
% F=exp(-X)-X.^3+0.2;
%Test: [I,Y,x]=nula(0.5,1,1e-4,10) %zadovoljena tacnost tol
%Test: [I,Y,x]=nula(0.5,1,1e-4,3) %zadovoljen max br iteracija iterM
%Test: [I,Y,x]=nula(0.5,1,1e-4,5) %zadovoljena je tacnost i dostignut max
%br iteracija iterM
```

type [Zadatak22/Njutn1.m](#)

```
function koef=Njutn1()
%Funkcija koef=Njutn1() vraca koeficijente koef I Njutnovog
%interpolacionog polinoma po promenljivoj q konstruisanog koriscenjem
%svih vrednosti iz tablica.m

tablica;
n=length(X);

KR=zeros(n,n-1);%tablica konacnih razlika
for i=1:n-1
    KR(i,1)=F(i+1)-F(i);
end

for j=2:n-1
    for i=1:n-j
        KR(i,j)=KR(i+1,j-1)-KR(i,j-1);
    end
end

end
%disp([X' Y' KR]);

% I Njutnov interpolacioni polinom po promenljivoj q:
yk=F(1);
qk=[1 0];
for j=1:n-1
    yk=[0 yk]+qk*KR(1,j)/factorial(j);
    qk=conv(qk,[1 -j]);
end
koef=yk;
```

type [Zadatak22/nula.m](#)

```
function [I,Y,x]=nula(x0,xF,tol,iterM)
%Funkcija [I,Y,x]=nula(x0,xF,tol,iterM) odredjuje nulu tablicno zadate
%funkcije u tablica.m koriscenjem metode regula-falsi.
%
%Ulazni argumenti su:
%    x0 - pocetna vrednost iterativnog niza
%    xF - fiksirana tacka u metodi regula-falsi
%    tol - tacnost
```

```

% iterM - maksimalni dozvoljeni broj iteracija
%Izlazni argumenti su:
% I - vektor [1:iter] gde je iter broj iteracija
% dok se ne zadovolji tacnost tol ili iterM
% Y - vrednosti tablicno zadate funkcije u tackama iterativnog niza
% izracunata pomocu I Njutnovog interpolacionog polinoma
% x - nula tablicno zadate funkcije
%
tablica;
h=X(2)-X(1);
koef=Njutn1(); %Koeficijenti I Njutnovog int.pol. po q
f=@(x)polyval(koef,(x-X(1))/h); %Anonimna f-ja koja racuna vrednost
%I Njutnovog polinoma u x (q=(x-X(1))/h)
Y=zeros(1,iterM);% Y ima najvise iterM elemenata
iter=0;
while iter<iterM
    x1=x0-(f(x0)/(f(xF)-f(x0)))*(xF-x0); %regula-falsi
    iter=iter+1;
    Y(iter)=f(x1);
    greska=abs(x1-x0);
    if greska<tol
        break;
    end
    x0=x1;
end
I=1:iter;
Y=Y(1:iter);
% Izasli smo iz petlje ako je iter==iterM ili greska< tol
% znaci mora da vazi po izlasku iz petlje da je iter==iterM ili greska<tol
if iter==iterM && greska<tol
    disp('Zadovoljena je tacnost i dostignut max br iteracija.');
```

```
[I,Y,x]=nula(0,0.5,1e-4,15) % zadovoljena je tacnost
```

```

Zadovoljena je tacnost tol
I = 1x5
    1    2    3    4    5
Y = 1x5
    0.0845   -0.0080    0.0008   -0.0001    0.0000
x =
0.2133
```

```
[I,Y,x]=nula(0,0.5,1e-4,4) % nije zadovoljena tacnost
```

```

Zadovoljen je maksimalan broj iteracija.
I = 1x4
    1    2    3    4
Y = 1x4
    0.0845   -0.0080    0.0008   -0.0001
x =
0.2133
```

```
[I,Y,x]=nula(0,0.5,1e-4,6) % u petoj iteraciji je zadovoljena tacnost
```

```

Zadovoljena je tacnost tol
I = 1x5
    1    2    3    4    5
```

```

Y = 1×5
    0.0845    -0.0080     0.0008    -0.0001     0.0000
x =
    0.2133

```

Zadatak 20

```
type Zadatak20/tablica.m
```

```

% ----- I test -----
%X = linspace(0,4,5);
X=linspace(-1,0.5,5);
F = X-1;
% ----- II test -----
% X = linspace(0,2,6);
% F = X.^3-exp(X);
% %F = exp(X).*sin(X)-cos(X);

```

```
type Zadatak20/funk.m
```

```

% I test
% y = funk(0.15)
% dobija se da je y = -0.8500

% II test
%

function y = funk(x)

tablica;
n = length(X);
pr = zeros(n,n-1);

for i=1:n-1
    pr(i,1)=(F(i+1)-F(i))/(X(i+1)-X(i));
end

for j=2:n-1
    for i = 1: n-j
        pr(i,j) = (pr(i+1,j-1)-pr(i,j-1))/(X(i+j)-X(i));
    end
end
%[X' F' pr]
polinom = [ F(1)];
r = [1];

for i=1:n-1
    r = conv(r,[1 -X(i)]);
    polinom = [0 polinom] + r*pr(1,i);
end
y = polyval(polinom, x);
end

```

```
type Zadatak20/polov.m
```

```

function x = polov(tol)
tablica;
a=X(1);
b=X(end);
if (funk(a)*funk(b)>0) % nisu ispunjeni uslovi za primenu metode,
    % prekida se rad programa i ispisuje greska ('Funkcija...')
    error('Funkcija ne menja znak na posmatranom segmentu')
end

```

```

end
if funk(a)==0
    x=a;
else
    if funk(b)==0
        x=b;
    end
end
% ako ni jedna grana nije tru onda je funk(a)*funk(b)<0
%if (funk(a)~=0 && funk(b)~=0)
    iter=0;
    while(abs(a-b)>tol)
        x=(a+b)/2;
        iter=iter+1;
        if funk(x)==0
            break
        else
            if funk(a)*funk(x)<0
                b=x;
            else % u suprotnom funkcija menja znak na intervalu [x,b]
                a=x;
            end
        end
    end
end
x=(a+b)/2; % izašli smo iz petlje jer je |a-b|<tol a nula je bas u tom intervalu
% pa za procenu mozemo uzeti x=(a+b)/2
end

```

```
x=polov(1e-4) % za drugi test kada je X=linspace(-1,0.5,5)
```

Error using polov (line 7)
Funkcija ne menja znak na posmatranom segmentu

```
x=polov(1e-4)
```

```
x =
1
```