**FUNKCIJE**

1. Napisati funkciju krug koja izracunava obim i povrsinu kruga. Ulazni argument je poluprecnik kruga.

function [O, P] = krug(r)

P = pi\*r.^2;

O = 2\*pi\*r;

end

2. Napisati funkciju vektpr koja izracunava vektorski proizvod dva vektora koji su ulazni argumenti.

function vektorski = vektpr(x,y);

%Za ulazne vektore racuna njihov skalarni i vektorski proizvod

vektorski(1)=x(2)\*y(3)-y(2)\*x(3);

vektorski(2)=-x(1)\*y(3)+x(3)\*y(1);

vektorski(3)=x(1)\*y(2)-y(1)\*x(2);

end

Sa raznim komentarima uz naredbu error moze se “popraviti” po zelji.

3. Napisati funkciju polar koja Dekartove koordinate prebacuje u polarne koordinate.

function [r,t] = polar(x,y)

r=sqrt(x^2+y^2);

t=atan(y/x);

if (x<0 & y>0) || (x<0 & y <0)

t = t + pi;

end

4. Napisati funkciju po imenu fact1 koja racuna faktorijel prirodnog broja n koristeci for petlju. Ulazni argument je n, a izlaz je n!

Zatim napisati funkciju fact2 koja racuna faktorijel pomocu rekurzije.

function y = fact1(n)

%Racuna faktorijel bez rekurzije

y = 1;

for k=1:n

y = y\*k;

end

end

function y = fact2(n)

if n == 1

y=1;

else

y = n \* fact2(n-1);

end

5. Napisati funkciju matrica1 koja ima ulazne argumente m i n, a kao izlazni argument matricu Q formata 2n x 2m koja se sastoji od 4 n x m podmatrice.

Elementi podmatrice u gornjem levom uglu su svi jednaki 0, elementi podmatrice u gornjem desnom uglu su svi jednaki 1, donji levi su jednaki 2 i donji desni su

jednaki 3.

function Q = matrica1(n,m)

Q = zeros(n,m);

Q = [Q Q+1; Q+2 Q+3];

end

6. Napisati funkciju mojatabela1 koja nema ni ulaznih ni izlaznih argumenata a treba da napravi tabelu od 4 kolone a, b, c i d na sledeci nacin:

-kolona a su kvadrati prvih 10 prirodnih brojeva

-kolona b su koreni prvih 10 prirodnih brojeva

-kolona c je aritmeticka sredina kolona a i b

-kolona d je razlika kolone b i kolone a, d=b-a.

function [] = mojatabela1()

fprintf('======================================\n');

fprintf(' a b c d \n')

fprintf('======================================\n');

for j = 1:10

a = j^2;

b = sqrt(j);

c = (a+b)/2;

d = b - a;

fprintf('%4i %6.5f %6.5f %+6.5f \n',a,b,c,d);

end

fprintf('======================================\n');

7. Napisati funkciju mnozenje\_matrica koja kao ulazni argument ima dve matrice A i B, a kao izlazni argument daje C=A\*B (ne koristeci Matlab-ovu operaciju \*)

ukoliko je mnozenje moguce ili adekvatno obavestenje ukoliko nije moguce pomnoziti matrice A i B.

function C = mnozenjematrica (A,B)

[nrA,ncA] = size(A);

[nrB,ncB] = size(B);

if ncA ~= nrB

disp('Matrice A i B ne mogu se pomnoziti'); % alternative komandi error

return;

end

C = zeros(nrA,ncB);

for i = 1:nrA

for j = 1:ncB

for k = 1:ncA

C(i,j) = C(i,j) + A(i,k)\*B(k,j);

end

end

end

end

8. Napisati funkciju Kramer koja resava sistem linearnih jednacina AX=B primenom Kramerovog pravila. Ulazni argumenti su A i B, a izlazni argument resenje X ukoliko postoji, ili adekvatno obavestenje u suprotnom.

function X = Kramer(A,B);

[m,n]=size(A);

if m ~= n

error('Matrica nije kvadratna')

end

if det(A) == 0

error('Matrica je singularna'),

end

for j = 1:n

C=A;

C(:,j) = B;

X(j)=det(C)/det(A);

end

9. Napisati funkciju nadji\_nulu (f,x1,x2) koja nalazi nulu funkcije f na intervalu [x1, x2] sa tacnoscu 1e-10 koristeci Bolcano-Kosijevu teoremu. Na granicama intervala [x1,x2] funkcija f mora imati suprotan znak.

function x = nadji\_nulu (f, x1,x2)

x = (x1+x2)/2;

while abs(f(x)) > 1e-10

if f(x1)\*f(x)>0

x1 = x;

else

x2 = x;

end

x = (x1+x2)/2;

end

end