

1) (8 poena) Napisati M-fajl *zad1.m* sa funkcijom  $P = \text{zad1}(f, \text{alfa}, n)$  koja određuje i kao rezultat vraća polinom  $P$  stepena  $n$  takav da je  $\int_{-1}^1 (1-x^2)^{\alpha-1/2} [f(x) - P_n(x)]^2 dx$ ,  $\alpha = \text{const}$ , minimalno. Za bazis koristiti familiju Gegenbauerovih polinoma koji su ortogonalni na  $[-1, 1]$  u odnosu na težinsku funkciju  $w(x) = (1-x^2)^{\alpha-1/2}$  i koji se mogu odrediti korišćenjem rekurentne formule:

$$G_n(x) = \frac{1}{n} [2x(n + \alpha - 1) G_{n-1}(x) - (n + 2\alpha - 2) G_{n-2}(x)], \quad G_0(x) = 1, \quad G_1(x) = 2\alpha x.$$

2) (a) (7 poena) Napisati M-fajl *zad2.m* sa funkcijom  $H = \text{zad2}(a, b, n)$  koja najpre tabelira funkciju  $f(x) = 3x\sqrt{x-1}$  na intervalu  $[a, b]$  sa  $n$  čvorova, a zatim formira i kao rezultat vraća Hermiteov interpolacioni polinom formiran na osnovu vrednosti funkcije i vrednosti prvog izvoda funkcije u svih  $n$  tačaka.

(b) (2 poena) Napisati M-fajl *zad2b.m* sa funkcijom  $N = \text{zad2b}(a, b, x, \text{tol})$  koja određuje najmanji broj čvorova interpolacije  $N$  potrebnih da greška interpolacije u tački  $x$  ne bude veća od  $\text{tol}$ . Grešku interpolacije računati kao razliku tačne vrednosti funkcije  $f(x)$  i vrednosti formiranog polinoma  $H(x)$  u traženoj tački.

3) (6 poena) Napisati M-fajl *zad3.m* sa funkcijom  $H = \text{zad3}(x, a)$  koja za zadati vektor  $x$  određuje i kao rezultat vraća Householder-ovu matricu  $H$  koja će vektor  $x$  preslikati u vektor  $[a, 0, \dots, 0]^T$ .

4) (7 poena) Napisati M-fajl *zad4.m* sa funkcijom *zad4()* koja sa tačnošću  $10^{-4}$  pronalazi sva rešenja datog sistema nelinearnih jednačina

$$xe^x - y = 2, \quad y^2 - x^2 = 0.5.$$

Potrebno je u funkciji najpre grafički lokalizovati sva rešenja, a zatim za svako rešenje odrediti broj iteracija potreban Njutnovoj i modifikovanoj Njutnovoj metodi za dostizanje tog rešenja polazeći od iste aproksimacije za početno rešenje. Za svako rešenje funkcija ispisuje tekst oblika:

Resenje je

x= ...(izracunata vrednost)

y= ...(izracunata vrednost)

Broj potrebnih iteracija za dostizanje ovog resenja Njutnovom metodom je ...

Broj potrebnih iteracija za dostizanje ovog resenja modifikovanom Njutnovom metodom je ...

#### TEST PRIMER:

```
>> P=zad1(@cos,1,3)
```

P =

```
-0.3652 -0.1249 1.0059 1.0104
```

```
>> H=zad2(2.5,4.5,4)
```

H =

```
0.0001 -0.0016 0.0175 -0.1056 0.3339 0.2272 3.3318 -3.0219
```

```
>> N=zad2b(2.5,4.5,2.8,1e-10)
```

N =

```
7
```

```
>> H=zad3([3 5 -1]',11)
```

H =

```
0.9429 1.5714 -0.3143
1.5714 -0.9781 -0.1762
-0.3143 -0.1762 -1.8241
```

%provera:

```
>> H*[3 5 -1]'
```

ans =

```
11.0000
0.0000
-0.0000
```

```
>> zad4()
Resenje je
x=
  -2.1382
```

```
y=
  -2.2520
```

```
Broj potrebnih iteracija za dostizanje ovog resenja Njutnovom metodom je :
  3
```

```
Broj potrebnih iteracija za dostizanje ovog resenja modifikovanom Njutnovom metodom je:
  4
```

```
Resenje je
x=
  0.5942
```

```
y=
  -0.9237
```

```
Broj potrebnih iteracija za dostizanje ovog resenja Njutnovom metodom je :
  6
```

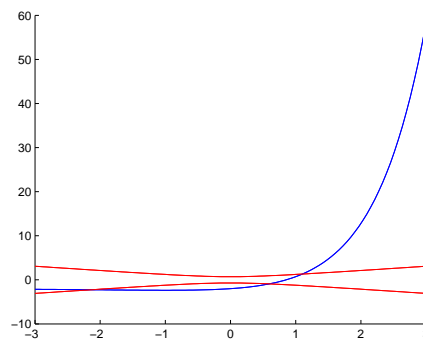
```
Broj potrebnih iteracija za dostizanje ovog resenja :
  5
```

```
Resenje je
x=
  1.1006
```

```
y=
  1.3081
```

```
Broj potrebnih iteracija za dostizanje ovog resenja :
  6
```

```
Broj potrebnih iteracija za dostizanje ovog resenja :
  30
```



Grafik za zad4.m, lokalizacija resenja