II kolokvijum iz Numeričkih metoda

1 Odrediti donju granicu broja značajnih cifara prilikom izračunavanja funkcije `exp`. Napisati script file u `Matlabu` koji izračunava broj značajnih cifara prilikom izračunavanja funkcije `exp` na intervalu $[-100, 100]$, sa korakom .1, pod pretpostavkom da su argumenti funkcije sa gornjom granicom apsolutne greške 10^{-5}. Nacrtati eksperimentom i teorijski dobijenu zavisnost broja značajnih cifara u funkciji argumenta izračunavanja.

Rešenje. Relativna greška prilikom izračunavanja funkcije `exp` je odredjena sa

$$r_f = \frac{\Delta_e xp}{\exp(x)} = \frac{|\exp(x') - \exp(x)|}{\exp(x)} \leq \frac{\exp(x)}{\exp(x)}\Delta_x = |x' - x| \leq 10^{-5}, \quad z_f \approx -\log_{10} r_f \geq -\log_{10} 10^{-5} =$$

Script file može izgledati ovako

```
x=(-100:.1:100)';
xPrim=x+10^(-5)*(2*rand(length(x),1)-1);
y=exp(x);
yPrim=exp(xPrim);
z_fe=-log10(abs(1-yPrim./y));
z_ft=5+0*x;
plot(x,z_fe,'-',x,z_ft,'+');
grid on;
xlabel('x');
ylabel('broj znacajnih cifara');
legend('eksperiment','teorija');
```

Dobijamo sliku 1.

2. Dokaziti da alternativni red

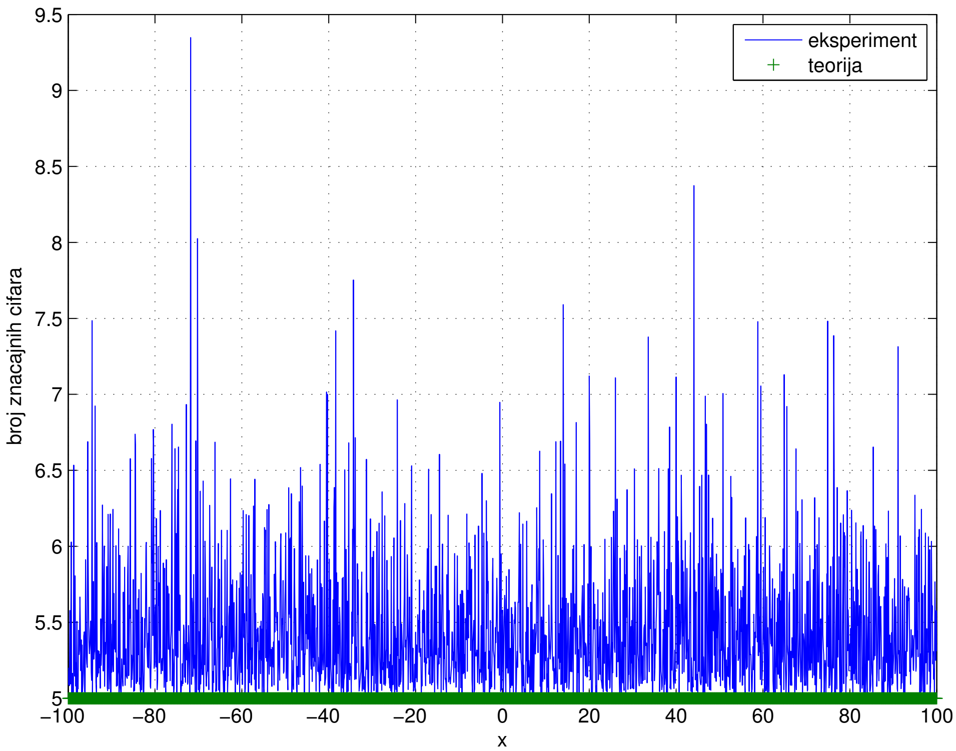
$$\sum_{k=0}^{+\infty} (-1)^k \frac{20^k}{k!}$$

konvergira. Sumirati dati red. Napisati pogram u `Matlabu` koji izračunava sumu reda (suma reda ne može biti `NaN`) i odrediti relativnu grešku i broj značajnih cifara u dobijenom rezultatu. Koristeći model formata `double` sa šesnaest dekadnih cifara mantise objasniti dobijene rezutlate.

Rešenje. Ako označimo $a_k = 20^k/k!$, $k \in \mathbb{N}_0$, nalazimo da je

$$a_{k+1} = \frac{20}{k+1} a_k < a_k, \quad k < 20.$$

Takodje, $\lim a_k = 0$. Na osnovu Liebnitzovog kriterijuma nalazimo da red konvergira. Korsiteći razvoj funkcije exp u potencijalni red nalazimo da je suma reda e^{-20}. Script file koji sumira red može biti recimo



Slika 1: Broj značajnih cifara u funkciji argumenta funkcije exp.

```
nN=100000;
suma=1;
a=1;
for k=1:nN
        a=a*(-20/k);
        suma = suma+a;
end
r=abs(exp(-20)-suma)/exp(-20)
z=-log10(r)
```

Izvršenjem dobijamo

```
r =
   1.982583036019132
z =
  -0.297231385815472
```

Ništa se bitno ne menja i ako za `nN` uzmemo i znatno veću vrednost.

Za objašenje razloga ovako malog broja značjanih cifara, uz činjenicu da se maksimalna vrednost izraza $20^k/k!$, $k \in \mathbb{N}_0$, postiže za $k = 20$, i iznosi $\approx 10^7$, dovoljno je setiti se da sva izračunavanja na poziciji $10^{7-16} = 10^{-9}$ ne mogu biti tačna. (Videti sličan primer u knjizi.)

3. Koristeći `Matlab` formirati donje trougaonu matricu

$$A_k = \begin{bmatrix} 10^{-k} & 0 & 0 & \dots & 0 \\ 1 & 10^{-k} & 0 & \dots & 0 \\ 1 & 1 & 10^{-k} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 10^{-k} \end{bmatrix}.$$

reda 10. Izračunati vrednost vektora $b_k = A_k x$, gde je x vektor čije su sve koordinate jednake 1.

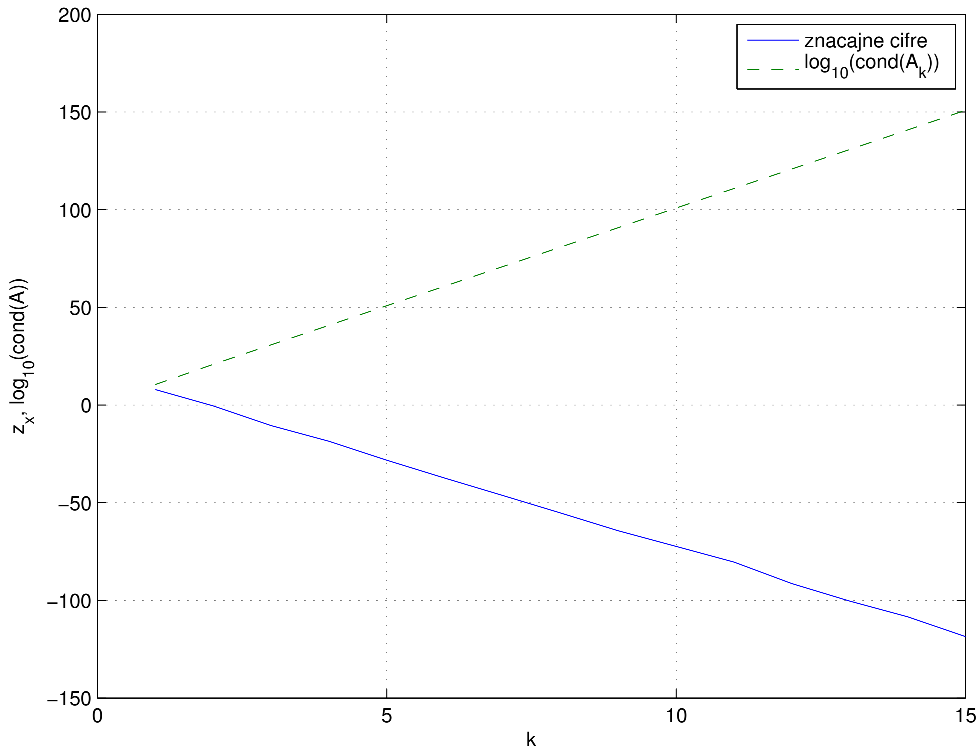
Za $k = 1, \dots, 15$, koristeći funkciju `linsolve` rešiti sistem linearnih jednačina $A_k x_k^G = b_k$. Nacrtati zavisnost broja značajnih cifara rešenja x_k^G i zavisnost faktora uslovljenosti matrica A_k u funkciji k. Interpretirati rezultate sa grafika.

Rešenje. Script file koji crta potreban grafik može biti sledeći

```
A=tril(ones(10,10),-1);
x=ones(10,1);
err=[];
c=[];
ops.LT=true;
for k=1:15
        A(1:11:end)=10^(-k);
        b=A*x;
        xG=linsolve(A,b,ops);
        err=[err norm(xG-x)/norm(x)];
        c=[c norm(A)*norm(inv(A))];
end
plot(1:15,-log10(err),'-',1:15,log10(c),'--');
grid on;
xlabel('k');
legend('znacajne cifre','faktor uslovljenoti');
```

Dobijamo sliku 2. Kako znamo ocena relatine greške rešenja sistema lienarnih jednačina može se dati koristeći izraz

$$\frac{||\Delta x||}{||x||} \leq \frac{2||A||\,||A^{-1}||}{1-r} \max\{\frac{||\Delta b||}{||b||}, \frac{||\Delta A||}{||A||},$$



Slika 2: Značajne cifre i faktor uslovljenosti u funkciji k.

uz aproksimaciju $r_b = r_A = 2^{-53}$ i $r = ||\Delta A||\,||A^{-1}|| < 1$, dobijamo

$$z_x \approx -\log_{10}\frac{||\Delta x||}{||x||} \geq -\log_{10}\frac{2||A||\,||A^{-1}||}{1-r} + 16 \approx 16 - \log_{10}\mathrm{cond}(A),$$

što se otprilike i vidi na priloženim graficima, jer su prave očigledno samo istim nagibima suprotnog znaka.

4. Posmatrajmo iterativni metod

$$x^{k+1} = x^k + \frac{20}{7}(b - Ax^k), \quad k \in \mathbb{N}_0, \quad A = \begin{bmatrix} \frac{23}{180} & \frac{17}{180} & \frac{11}{180} \\ -\frac{3}{20} & \frac{7}{20} & \frac{3}{20} \\ -\frac{11}{45} & \frac{4}{45} & \frac{47}{90} \end{bmatrix}, \quad b = \begin{bmatrix} \frac{17}{60} \\ \frac{7}{20} \\ \frac{11}{30} \end{bmatrix}.$$

Pokazati da niz x^k, $k \in \mathbb{N}_0$ konvergira ka rešenju sistema linearnih jednačina $Ax = b$. Odrediti prinos broja značajnih cifara po iteraciji. Odrediti približno potreban broj iteracija za dostizanje mašinske tačnosti. Implementirati metod u script fileu `Matlaba` i prikazati grafik zavisnosti broja značajnih cifara u funkciji broja iteracija. Uporediti dobijene rezultate sa teorijskim ocenama.

Rešenje. Iterativni proces možemo posmatrati u formi

$$x^{k+1} = Bx^k + \beta, \quad B = I - \frac{20}{7}A, \quad \beta = \frac{20}{7}b.$$

Konvergenija niza je obezbedjena ako je spektralni radijus matrice B manji od jedan. Uz pomoć sledećeg script filea

```
A=[23/180, 17/180, 11/180;
-3/20, 7/20, 3/20;
-11/45, 4/45, 47/90];
b=[17/60; 7/20; 11/30];
lambda=20/7;
B=eye(3)-lambda*A;
beta=lambda*b;
rho=max(abs(eig(B)))
```

Dobijamo

```
rho =
   0.428571428571429
```

odakle zaključujemo da je $\rho(B) \approx .42 < 1$, pa je proces konvergentan.

Niz vektora x^k, konvergira ka nekom vektoru x, za koji važi

$$x = x + \frac{20}{7}(b - Ax), \quad Ax = b.$$

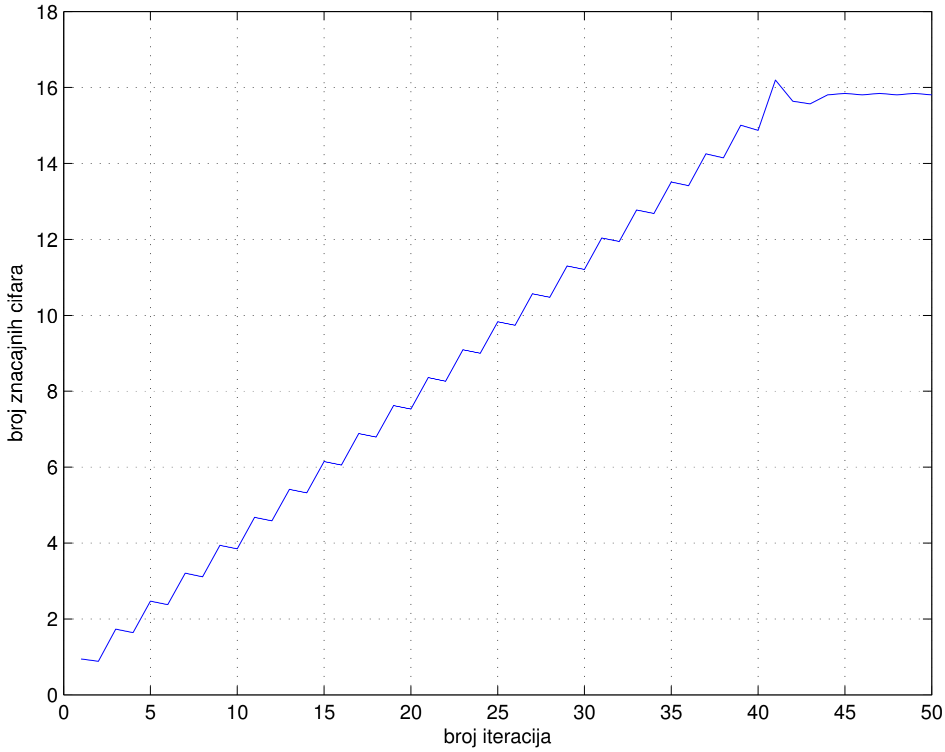
Na osnovu opšte teorije iterativnih procesa za rešavanje sistema lineranih jednačina nalazimo jednostavno da je prinos značajnih cifara o iteraciji $-\log_{10}\rho(B) \approx .368$. Za dostizanje mašinske tačnsoti potrebno nam je $16/.369 \approx 43.5$ iteracija.

Implementacija metoda u script fileu može biti

```
x=zeros(3,1);
A=[23/180, 17/180, 11/180;
-3/20, 7/20, 3/20;
-11/45, 4/45, 47/90];
b=[17/60; 7/20; 11/30];
x=A\\b;
lambda=20/7;
x0=zeros(3,1);
err=[];
for i=1:50
      x0=x0+lambda*(b-A*x0);
```

```
    err=[err norm(x-x0)/norm(x)];
end
plot(1:50,-log10(err));
grid on;
ylabel('broj znacajnih cifara');
xlabel('broj iteracija');
```

Dobijamo sliku 3.



Slika 3: Značajne cifre u funkciji broja iteracija.

Sa slike vidimo da se mašinksa tačnost, 16 značajnih cifara, postiže za otprilike 43 iteracija, što smo dobili i teorijskom ocenom.

5. Napisati funkciju `fS` u Matlabu koja implementira Steffensenov metod

$$x_{k+1} = x_k - \frac{(f(x_k))^2}{f(x_k + f(x_k)) - f(x_k)},$$

pri čemu funkcija `fS` prihvata kao argumente početnu vrednost aproksimacije `x0`, maksimalnu dozvoljenu relativnu grešku `prec`, maksimalnu dozvoljenu apsolutnu grešku `acc`, kao i maksimalni dozvoljeni broj iteracija `maxIter`.

Primeniti funkciju `fS` na rešavanje jednačine $\sin x - \cos x = 0$ sa startnom vrednošću

$x_0 = 1$. Oceniti red konvergencije iterativnog procesa proračunom prinosa značajnih cifara po iteraciji. Uporediti Steffensenov i Newtonov metod.

Pokazati analitički da Steffensenov metod ima red konvergencije dva i odrediti asimptotsku konstantu greške.

Rešenje. Steffensenov metod možemo implementirati na sledeći način

```
function [x1,err,iter]=fS(fun,x_0,prec,acc,maxIter)

x1 = x_0;
x0 = inf;
iter = 0;
err = [];
while abs(x1-x0)>max(abs(x1)*prec,acc)
    x0=x1;
    f0 = fun(x0);
    x1 = x0-f0^2/(fun(x0+f0)-f0);
    err = [err abs(x1-x0)/abs(x1)];
    iter = iter+1;
    if(iter > maxIter)
        display('Maksimalni broj iteracija prekoracen');
    end
end
```

Script file koji izvršava funkciju `fS` može biti sledeći

```
fun=@(x)(sin(x)-cos(x));
[x,err,iter]=fS(fun,1,10^(-15),10^(-20),100)
```

Dobijamo izlaz

```
x =
   0.785398163397448
err =
  Column 1
   0.296776034906197
  Column 2
   0.018154979718138
  Column 3
   0.000005067761508
  Column 4
   0.000000000000000
```

```
iter =
     4
```

Broj značajnih cifara, u svakoj iteraciji, možemo odrediti na sledeći način

```
>>-log10(err)
ans =
  Column 1
   0.527571171884292
  Column 2
   1.741004231923522
  Column 3
   5.295183831489573
  Column 4
  15.849679651557175
```

Možemo oceniti da u svakoj iteraciji broj značajnih cifara raste tri puta.

Da bismo mogli uporediti Steffensenov i Newtnov metod moramo oceniti red konvergencije Steffensenovog metoda. Iterativna funkcija Steffensenovog metoda je

$$\phi(x) = x - \frac{f^2(x)}{f(x + f(x)) - f(x)}.$$

Na osnovu razvoja funkcije f u tački x, sa priraštajem $f(x)$, nalazimo

$$\phi(x) = x - \frac{f(x)}{f'(x) + O(f(x))} = x - \frac{f(x)}{f'(x)} + f(x)O(f(x)) = \phi_N(x) + O(f^2(x)),$$

gde smo sa ϕ_N označili iterativnu funkciju Newtnovog metoda.

Pretpostavimo da je $f(\alpha) = 0$. Medjutim, za Newtnov metod važi

$$\phi_N(x) = \alpha + \frac{1}{2}\frac{f''(\alpha)}{f'(\alpha)}(x - \alpha)^2,$$

pa je

$$\phi(x) \approx \alpha + \frac{1}{2}\frac{f''(\alpha)}{f'(\alpha)}(x - \alpha)^2,$$

zaključujemo da je red Steffensenovog metoda barem dva. U konrektnom rešavanju jednačine $\sin x - \cos x = 0$, vidimo da je red metoda jedank tri, jer dobijamo tri puta više značajnih cifara u narednoj iteraciji u odnosu n prethodnu. Zaključujemo da Steffensenov metod ima istu asimptotsku konstantu greške kao i Newtonov metod.

Da prethodno ponašanje nije pravilo, možemo pokazati koristeći jednačinu $x - \cos x = 0$. Izvršavanjem scripta

```
fun=@(x)(x-cos(x));
[x,err,iter]=fS(fun,1,10^(-14),10^(-20),100);
-log10(err)
```

dobijamo

```
ans =
  Column 1
   0.506161913559633
  Column 2
   1.508351364900582
  Column 3
   3.383348428311638
  Column 4
   7.126975155408626
  Column 5
  14.619164254016624
```

Vidimo da broj značajnih cifara raste dva puta u odnosu na prethodnu iteraciju, dakle, red metoda je dva.

Kao što vidimo Newtonov i Steffensenov metod imaju isti red konvergencije, medjutim, Steffensenov metod je superioran u odnosu na Newtonov jer ne koristi informaciju u izvodu funkcije. Steffensenov metod nastaje iz Newtonovog metoda aproksimacijom

$$f'(x) \approx \frac{f(x + f(x)) - f(x)}{f(x)},$$

što je prihvatljivo u okolini tačke α.

prof. dr Aleksandar Cvetković