Elektrotehnički fakultet
Univerzitet u Beogradu

Odabrana poglavlja iz numeričke analize

Prvi projektni zadatak

Za dati broj x i opseg n i m, formirati niz razlomaka $\frac{p}{q}$ gde imenilac q ispunjava $n \leq q \leq m$. Brojilac p je zaokruzen na najbliži ceo broj vrednosti $x \cdot q$. Potom u prethodnom nizu razlomaka formirati:

- Racionalne aproksimacije I vrste
- Racionalne aproksimacije II vrste
- Razlomke $\frac{p}{q}$ sortirati po minimalnosti apsolutne greške $|x - \frac{p}{q}|$.

Formirati i zapisati u veriznom zapisu sve dobijene aproksimacije.

Jovana Savić 2020/3423

1. Racionalne aproksimacije i verižni zapis

1.1. Racionalne aproksimacije I i II vrste

Definicija 1. Racionalan broj $\frac{p}{q}$ jeste najbolja racionalna aproksimacija realnog broja α prve vrste ako i samo ako važi nejednakost:

$$|\alpha - \frac{p}{q}| < |\alpha - \frac{r}{s}|$$

za sve razlomke $\frac{r}{s} \neq \frac{p}{q}$ takve da je $0 < s \leq q$.

Definicija 2. Racionalan broj $\frac{p}{q}$ jeste najbolja racionalna aproksimacija realnog broja α druge vrste ako i samo ako važi nejednakost:

$$|q\alpha - p| < |s\alpha - r|$$

za sve razlomke $\frac{r}{s} \neq \frac{p}{q}$ takve da je $0 < s \leq q$.

Pored pomenutih definicija navodimo i jedno svojstvo koje će takođe biti korišćeno u rešenju.

Svojstvo 1. Najbolja racionalna aproksimacija druge vrste jeste i najbolja racionalna aproksimacija prve vrste. Obratno ne mora da važi.

1.2. Verižni zapis

U matematici, verižni razlomak je izraz oblika

$$x = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}}$$

gde je $a_0 \in Z$ i $a_i \in N, i \neq 0$.

Prethodni izraz skraćeno zapisujemo

$$x = [a_0; a_1; a_2; ...]$$

U konkretnom slučaju radimo sa konačnim verižnim razlomcima koji su određeni u obliku razlomka:

$$\alpha = \frac{p_n}{q_n} = [a_0; a_1; a_2; \dots ; a_n]$$

Posmatrajmo razlomke $\frac{p_k}{q_k} = [a_0, a_1, \dots a_k]$ za $k \in \{0, 1, \dots n\}$. Tada na osnovu $p_0 = a_0$, $q_0 = 1$ i $p_1 = a_0 a_1 + 1$, $q_1 = a_1$ korišćenjem rekurencija

$$p_k = a_k p_{k-1} + p_{k-2}$$
$$q_k = a_k q_{k-1} + q_{k-2}$$

koje važe za $k \geq 2$ dobijamo niz razlomaka $\frac{p_k}{q_k}$. Ove razlomke nazivamo konvergentama. Racionalne brojeve razmatrajmo kao redukovane prave razlomke.

Tada, svaki racionalni broj $\alpha = \frac{p}{q}$ možemo zapisati u obliku konačnog verižnog razlomka tako što verižne decimale određujemo pomoću Euklidovog algoritma:

$$\begin{aligned} p &= a_0 q + r_1 \\ q &= a_1 r_1 + r_2 \\ r_1 &= a_2 r_2 + r_3 \\ &\dots \\ r_{n-2} &= a_{n-1} r_{n-1} + r_n \\ r_{n-1} &= a_n r_n \end{aligned}$$

pri tom $r_n < r_{n-1} < \dots < r_2 < r_1 < q$. Verižne decimale su jedinstveno određene i poslednja verižna decimala ne može biti 1.

Dalje navodimo neka svojstva.

Svojstvo 2. Svaka najbolja racionalna aproksimacija prve vrste broja α je konvergenta ili neka sekundarna konvergenta tog broja.

Svojstvo 3. Svaka najbolja aproksimacija druge vrste broja α je konvergenta. Obrnuto, svaka konvergenta je najbolja racionalna aproksimacija druge vrste broja α; izuzev izolovanog slučaja $\alpha = a_0 + \frac{1}{2}$ i $\frac{p_0}{q_0} = \frac{a_0}{1}$.

U nizu najboljih aproksimacija prve vrste nalaze se sve konvergente posmatranog broja i te konvergente nazivamo verižnim aproksimacijama. U nizu najboljih aproksimacija prve vrste su i neke sekundarne konvergente posmatranog broja i sve takve sekundarne konvergente nazivamo međuverižnim aproksimacijama.

Svojstvo 4. Niz imenilaca $q_1, q_2, \ldots q_n, \ldots$ verižnih aproksimacija jeste strogo rastući niz brojeva. Apsolutne razlike između verižnih aproksimacija $\frac{p_n}{q_n}$ i $\frac{p_{n-1}}{q_{n-1}}$, određene sa $|\Delta_n| = |\frac{p_n}{q_n} - \frac{p_{n-1}}{q_{n-1}}| = \frac{1}{q_{n-1}q_n}$ monotono teže ka 0.

Svojstvo 5. Racionalan broj jeste najbolja aproksimacija druge vrste ako i samo ako jeste verižna aproksimacija za dati realan broj.

Neka su za realan broj α određene redom verižne aproksimacije: $\frac{p_{n-2}}{q_{n-2}} = [a_0, a_1, \ldots, a_{n-2}]$, $\frac{p_{n-1}}{q_{n-1}} = [a_0, a_1, \ldots, a_{n-1}]$ i $\frac{p_n}{q_n} = [a_0, a_1, \ldots, a_n]$. Niz razlomaka oblika $\frac{p_{n-2}+j\dot{p}_{n-1}}{q_{n-2}+j\dot{q}_{n-1}}$ određuje međuverižni niz za tražene razlomke $\frac{p_{n-2}}{q_{n-2}}$ i $\frac{p_n}{q_n}$.

Svojstvo 6. Racionalan broj je najbolja aproksimacija prve vrste ako i samo ako jeste verižna aproksimacija ili razlomak iz međuverižnog niza za dati broj α.

Svojstvo 7. Neka su za realan broj α date dve uzastopne aproksimacije $\frac{p_{n-1}}{q_{n-1}} = [a_0, a_1, \ldots, a_{n-1}]$ i $\frac{p_n}{q_n} = [a_0, a_1, \ldots, a_n]$. Ako postoji prva međuverižna aproksimacija prve vrste $\frac{p_n'}{q_n'} = [a_0, a_1, \ldots a_{n-1}, a_n']$ sa međuverižnom cifrom $0 < a_n' < a_n$, tada su razlomci $\frac{p_n''}{q_n''} = [a_0, a_1, \ldots a_{n-1}, a_n'']$ redom za $a_n' < a_n'' < a_n$ sve bolje međuverižne aproksimacije prve vrste datog broja α.

2. Algoritam i implementacija u programskom jeziku Python

Naš zadatak je da odredimo sve racionalne aproksimacije prve i druge vrste, takve da su imenioci u datom opsegu. Na osnovu datih svojstava dobija se algoritam $[1]$ koji na osnovu verižne aproksimacije datog broja formira njemu odgovarajuće verižne i međuverižne aproksimacije.

Za precizna izračunavanja se koristi `mpmath` biblioteka $[2]$. U pitanju je besplatna biblioteka otvorenog koda koja je razvijena od strane Fredrik Johansson-a koja omogućava rad sa realnim i kompleksnim brojevima sa proizvoljnom preciznošću. Bibilioteka je samostalna i veoma efikasno izvršava računske operacije sa izabranom preciznošću, a omogućava i veliki broj veoma naprednih alata poput onih za rad sa matricama. Osim toga, u ovoj biblioteci su predefinisane i najbitnije matematičke konstante, pa su one dodate u program kao gotova opcija.

Realni brojevi su ovoj biblioteci predstavljeni kao uređena četvorka (znak, mantisa, eksponent, broj bitova) gde je broj bitova broj bitova mantise. Broj je predstavljen na standardan način kao proizvod mantise i eksponenta dvojke.

Preciznost koja se koristi se podešava kao globalna promenljiva, mada je moguće definisati je i posebno za konkretno izračunavanje.

Algoritamski deo programa je smešten u fajl `verige.py` i može se koristiti i samostalno. Dalje dajemo objašnjenja i kod koji izvršava suštinski deo programa.

Ispod su definisane konstante koje se nalaze u biblioteci `mpmath` koja je uvedena pod alijasom `mp` i koje su korisniku date kao jedna od opcija.

```
mp.dps = 1000

math_constants = {
"pi" : mp.pi,
"e" : mp.e,
"phi" : mp.phi,
"euler" : mp.euler,
"catalan" : mp.catalan,
"apery" : mp.apery,
"khinchin" : mp.khinchin,
"glaisher" : mp.glaisher,
"mertens" : mp.mertens,
```

```
"twin prime" : mp.twinprime
}
```

U toku rada se često javlja potreba za nalaženjem apsolutne greške, pa je napravljena funkcija koja objedinjuje njeno izračunavanje.

```
    def difference(num, p, q):
            return mp.fabs(mp.fsub(num, mp.fdiv(p, q)))
```

Dole dajemo funkciju koja računa verižni zapis i na osnovu njega verižne aproksimacije. Funkcija računa prvih 1000 verižnih aproksimacija, odnosno staje kada je došla do razlomka čiji je imenilac veći od granice. Pošto je za nalaženje verižnih aproksimacija potrebno krenuti od početka, to jest od razlomka čiji je imenilac jedinica, leva granica se ignoriše.

```
def verige(num, denominator_limit):
   Z = num
   C = []
   P = []
   Q = []

   C.append(int(mp.floor(num)))
   Z = mp.fdiv(1, mp.frac(Z))
   C.append(int(mp.floor(Z)))

   P.append(C[0])
   P.append(C[0]*C[1] + 1)
   Q.append(1)
   Q.append(C[1])

       for k in range(2, 1000):
               Z = mp.fdiv(1, mp.frac(Z))
           C.append(int(mp.floor(Z)))
           if Q[-1] > denominator_limit:
                   break
           P.append(C[k] * P[k-1] + P[k-2])
               Q.append(C[k] * Q[k-1] + Q[k-2])


    return C, P, Q
```

Dalje je definisana funkcija za nalaženje verižnog zapisa razlomka. Ova funkcija za razlomak koji se prosleđuje kao lista od dva elementa nalazi verižni zapis. Ova funkcija nije neophodna. Naime, verižni zapis razlomaka druge vrste već imamo, a na osnovu svojstva 7 možemo da odredimo i međuverižne zapise. Ipak, u konkretnom slučaju ta ušteda u vremenu je jako mala jer svakako treba da formiramo verižni zapis svakog razlomka za ispis, pa je izabrano ovo rešenje koje daje jednostavniji kod.

```
def find_continued_fraction(fraction):
        assert len(fraction)==2

        [p, q] = fraction
        a0 = p // q
        r1 = p % q
        res = [a0]
        if r1 == 0:
            return res

    a1 = q // r1
        r2 = q % r1
        res.append(a1)

        while  not r2 == 0:
                res.append(r1 // r2)
                r3 = r1 % r2
                r1, r2 = r2, r3

        return res
```

Konačno, dajemo funkciju koja nalazi sve racionalne aproksimacije i ujedno ih generiše tako da je svaki sledeći sa manjom apsolutnom greškom. Ova funkcija je urađena na osnovu one date u $[1]$ samo prilagođena programskom jeziku Python. Ova funkcija vraća listu razlomaka, oznaku koja govori da li je u pitanju međuverižni ili verižni zapis kao i grešku.

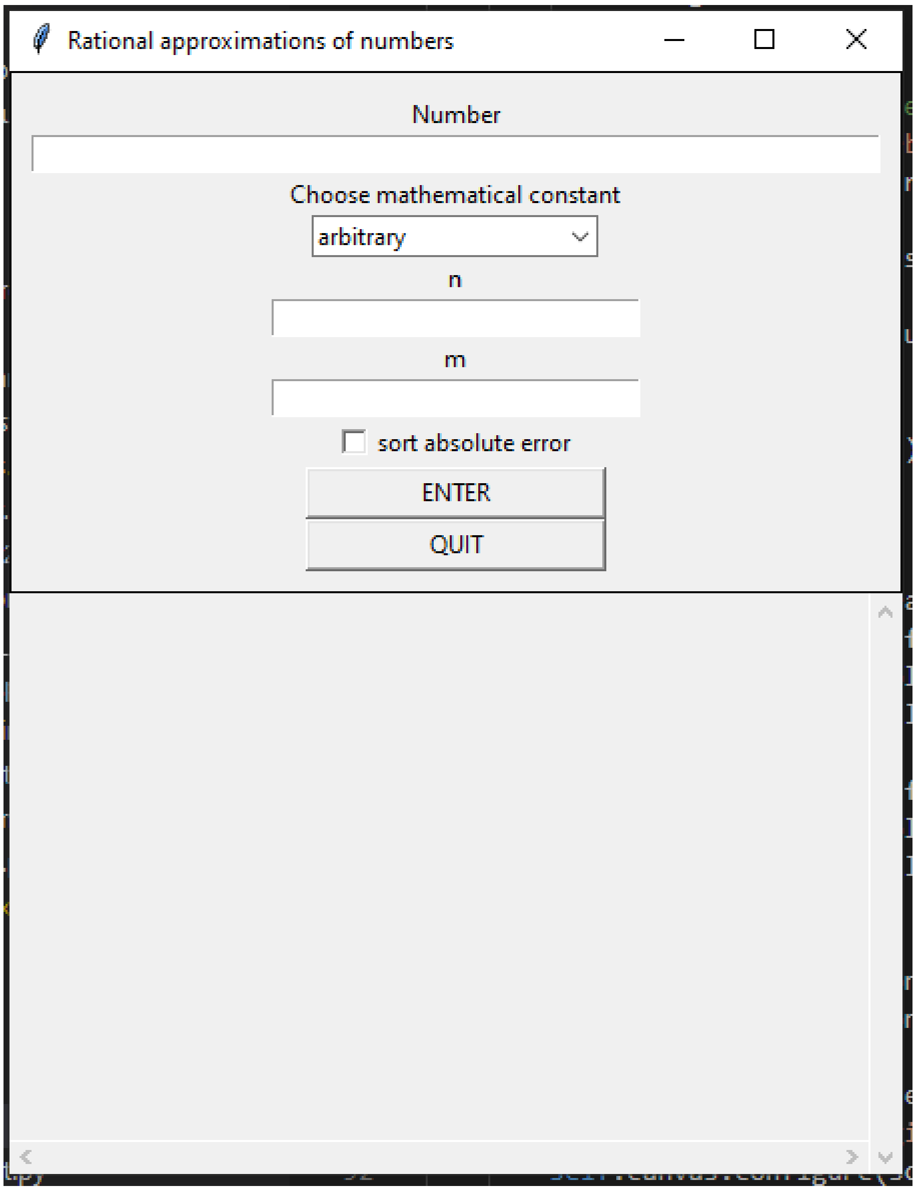
```
def find_all_approx(num, N, M):
    C, P, Q = verige(num, M)
        res = []
        for i in range(1, len(C)):
                m = C[i]
                a1 = P[i-1]
                b1 = Q[i-1]
                a0 = 1
                b0 = 0
                if i > 1:
                        a0 = P[i-2]
```

```
        b0 = Q[i-2]
d1 = difference(num, a1, b1)
if b1 >= N and b1 <= M:
        res.append([a1, b1, 1, mp.nstr(d1)])
if b1 > M:
        break
for j in range(1, m):
        a2 = a0 + j*a1
        b2 = b0 + j*b1
        d2 = difference(num, a2, b2)
        if d1 < d2:
                continue
        d1 = d2
        if b2 >= N and b2 <= M:
                res.append([a2, b2, 0, mp.nstr(d2)])
        if b2 > M:
                break
return res
```

3. Rezultati programa

3.1. Grafički interfejs

Grafički interfejs je napravljen korišćenjem `tkinter` biblioteke. Korišćeno okruženje je Visual Studio Code. Program se sastoji iz dela u kome su komande i dela koji služi za prikaz rezultata. Na slici 1. je prikazan interfejs programa.



Slika 1.

Prvo polje služi za unošenje broja. Korisnik može i da izabere neku od konstanti koje su već navedene. Ukoliko se selektuje opcija *arbitrary* koristi se broj koji je dat u polju *Number*. Polja *n* i *m* se koriste za unos granica imenilaca. Ove vrednosti moraju da budu nenegativni celi brojevi. Klikom na ENTER se prikazuju tražene aproksimacije, a klikom na QUIT se gasi program. Ako je *sort absolute error* selektovano program izbacuje niz aproksimacija tako što ih sortira u opadajući niz u na osnovu apsolutne greške.

Na slici 2. su prikazani rezultati za broj π. Kao što se vidi sa slike, verižne aproksimacije su podebljane i označene zvezdicom. Aproksimacije su date u obliku razlomka, pored je njihov verižni zapis, a potom i greška koja se pravi kada se koristi ta aproksimacija.

3.2. Rezultati programa za broj π

312689/99532	**[3, 7, 15, 1, 292, 1, 2]***	**2.91434e-11**
208341/66317	**[3, 7, 15, 1, 292, 2]***	**1.22356e-10**
104348/33215	**[3, 7, 15, 1, 293]***	**3.31628e-10**
103993/33102	**[3, 7, 15, 1, 292]***	**5.77891e-10**
103638/32989	[3, 7, 15, 1, 291]	1.49364e-9
103283/32876	[3, 7, 15, 1, 290]	2.41568e-9
102928/32763	[3, 7, 15, 1, 289]	3.34409e-9
102573/32650	[3, 7, 15, 1, 288]	4.27892e-9
102218/32537	[3, 7, 15, 1, 287]	5.22024e-9
101863/32424	[3, 7, 15, 1, 286]	6.16813e-9
101508/32311	[3, 7, 15, 1, 285]	7.12265e-9
101153/32198	[3, 7, 15, 1, 284]	8.08386e-9
100798/32085	[3, 7, 15, 1, 283]	9.05185e-9
100443/31972	[3, 7, 15, 1, 282]	1.00267e-8
100088/31859	[3, 7, 15, 1, 281]	1.10084e-8
99733/31746	[3, 7, 15, 1, 280]	1.19972e-8
99378/31633	[3, 7, 15, 1, 279]	1.29929e-8
99023/31520	[3, 7, 15, 1, 278]	1.39959e-8
98668/31407	[3, 7, 15, 1, 277]	1.5006e-8

Slika 2.

54293/17282	[3, 7, 15, 1, 152]	2.45304e-7
53938/17169	[3, 7, 15, 1, 151]	2.48674e-7
53583/17056	[3, 7, 15, 1, 150]	2.52089e-7
53228/16943	[3, 7, 15, 1, 149]	2.55549e-7
52873/16830	[3, 7, 15, 1, 148]	2.59056e-7
52518/16717	[3, 7, 15, 1, 147]	2.62611e-7
52163/16604	[3, 7, 15, 1, 146]	2.66213e-7
355/113	**[3, 7, 16]***	**2.66764e-7**
333/106	**[3, 7, 15]***	**8.32196e-5**
311/99	[3, 7, 14]	0.000178512
289/92	[3, 7, 13]	0.000288306
267/85	[3, 7, 12]	0.000416183
245/78	[3, 7, 11]	0.000567013
223/71	[3, 7, 10]	0.000747583
201/64	[3, 7, 9]	0.000967654
179/57	[3, 7, 8]	0.00124178
22/7	**[3, 7]***	**0.00126449**
19/6	[3, 6]	0.025074
16/5	[3, 5]	0.0584073
13/4	[3, 4]	0.108407
3/1	**[3]***	**0.141593**

Slika 3.

5.3. Rezultati programa za broj e

Na slici 4. su prikazane dobijene aproksimacije broja e.

271801/99990	[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 8, 1, 1, 5]	2.76227e-10
49171/18089	**[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 8, 2]***	**2.76651e-10**
25946/9545	**[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 9]***	**5.5151e-9**
23225/8544	**[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 8]***	**6.74695e-9**
20504/7543	[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 7]	2.22635e-8
17783/6542	[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 6]	4.25284e-8
15062/5541	[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 5]	7.01152e-8
12341/4540	[2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 4]	1.09867e-7
2721/1001	**[2, 1, 2, 1, 1, 4, 1, 1, 6, 2]***	**1.10177e-7**
1457/536	**[2, 1, 2, 1, 1, 4, 1, 1, 7]***	**1.75363e-6**
1264/465	**[2, 1, 2, 1, 1, 4, 1, 1, 6]***	**2.25857e-6**
1071/394	[2, 1, 2, 1, 1, 4, 1, 1, 5]	7.71678e-6
878/323	[2, 1, 2, 1, 1, 4, 1, 1, 4]	1.55746e-5
685/252	[2, 1, 2, 1, 1, 4, 1, 1, 3]	2.78602e-5
193/71	**[2, 1, 2, 1, 1, 4, 2]***	**2.80307e-5**
106/39	**[2, 1, 2, 1, 1, 5]***	**0.000333111**
87/32	**[2, 1, 2, 1, 1, 4]***	**0.000468172**
68/25	[2, 1, 2, 1, 1, 3]	0.00171817
49/18	[2, 1, 2, 1, 1, 2]	0.00394039
19/7	**[2, 1, 2, 2]***	**0.00399611**
11/4	**[2, 1, 3]***	**0.0317182**
8/3	**[2, 1, 2]***	**0.0516152**
5/2	[2, 2]	0.218282
3/1	**[3]***	**0.281718**
2/1	**[2]***	**0.718282**

Slika 4.

5.4. Rezultati programa za broj ϕ

Slika 5. prikazuje aproksimacije koje nalazi program za konstantu zlatnog preseka. Dobijamo poznat verižni zapis.

121393/75025	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	7.94518e-11
75025/46368	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	2.08007e-10
46368/28657	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	5.4457e-10
28657/17711	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	1.4257e-9
17711/10946	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	3.73254e-9
10946/6765	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	9.77191e-9
6765/4181	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	2.55832e-8
4181/2584	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	6.69777e-8
2584/1597	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	1.7535e-7
1597/987	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	4.59072e-7
987/610	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	1.20186e-6
610/377	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	3.14653e-6
377/233	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	8.23768e-6
233/144	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	2.15668e-5
144/89	[1, 1, 1, 1, 1, 1, 1, 1, 1, 2]*	5.64607e-5
89/55	[1, 1, 1, 1, 1, 1, 1, 1, 2]*	0.000147829
55/34	[1, 1, 1, 1, 1, 1, 1, 2]*	0.00038693
34/21	[1, 1, 1, 1, 1, 1, 2]*	0.00101363
21/13	[1, 1, 1, 1, 1, 2]*	0.00264937
13/8	[1, 1, 1, 1, 2]*	0.00696601
8/5	[1, 1, 1, 2]*	0.018034
5/3	[1, 1, 2]*	0.0486327
3/2	[1, 2]*	0.118034
2/1	[2]*	0.381966
1/1	[1]*	0.618034

Slika 5.

5.4. Rezultati programa za Hinčinovu konstantu

Aleksandr Yakovlevich Khinchin je dokazao da za skoro svaki realan broj x verižne cifre a_i imaju konačnu geometrijsku sredinu koja je nezavisna od broja x. To jest, za

$$x = a_0 + \cfrac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}}$$

skoro uvek važi:

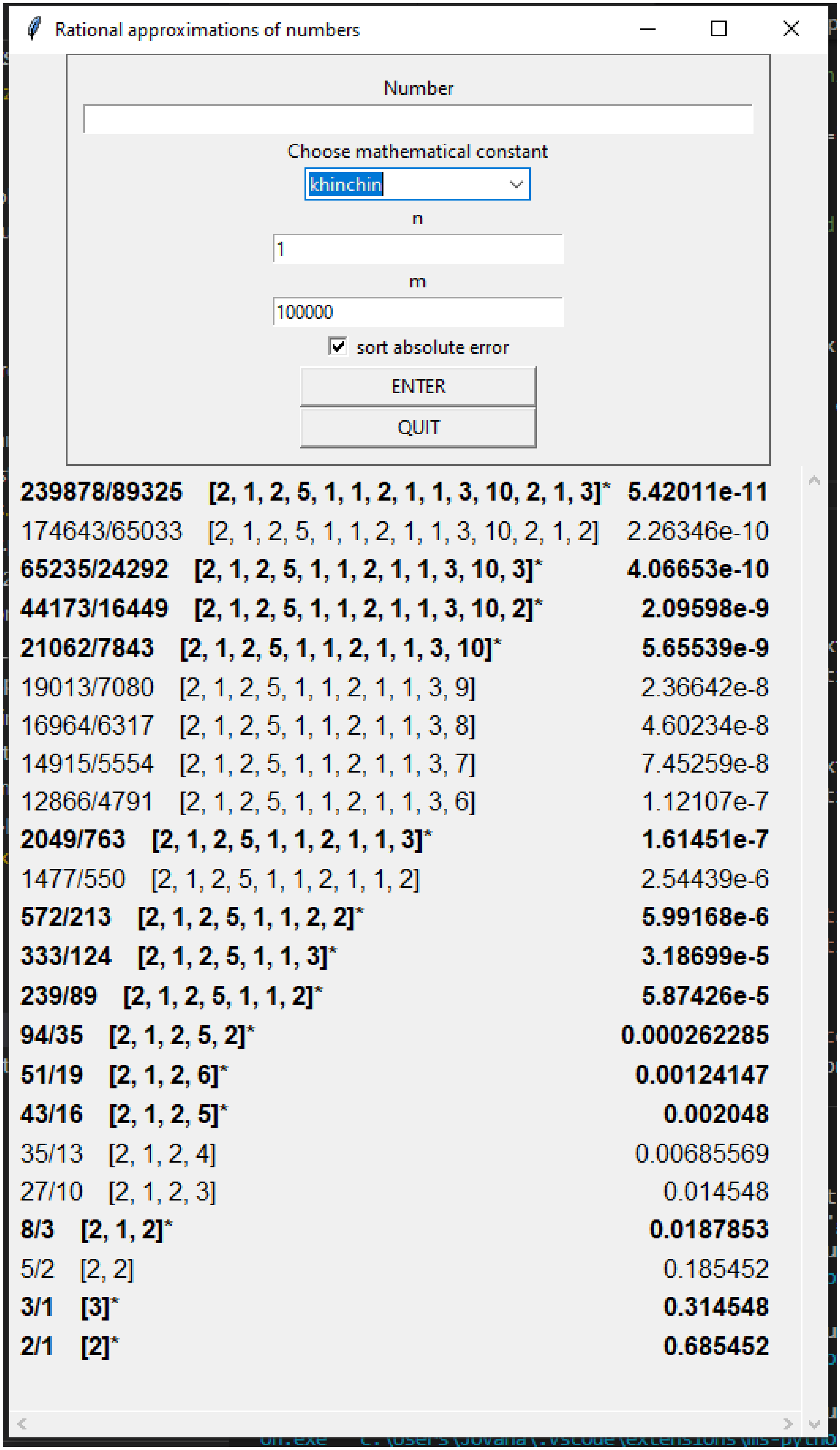
$$\lim_{n \to \infty} (a_1 a_2 \dots a_n)^{\frac{1}{n}} = K_0$$

gde je K_0 Hičinova konstanta.

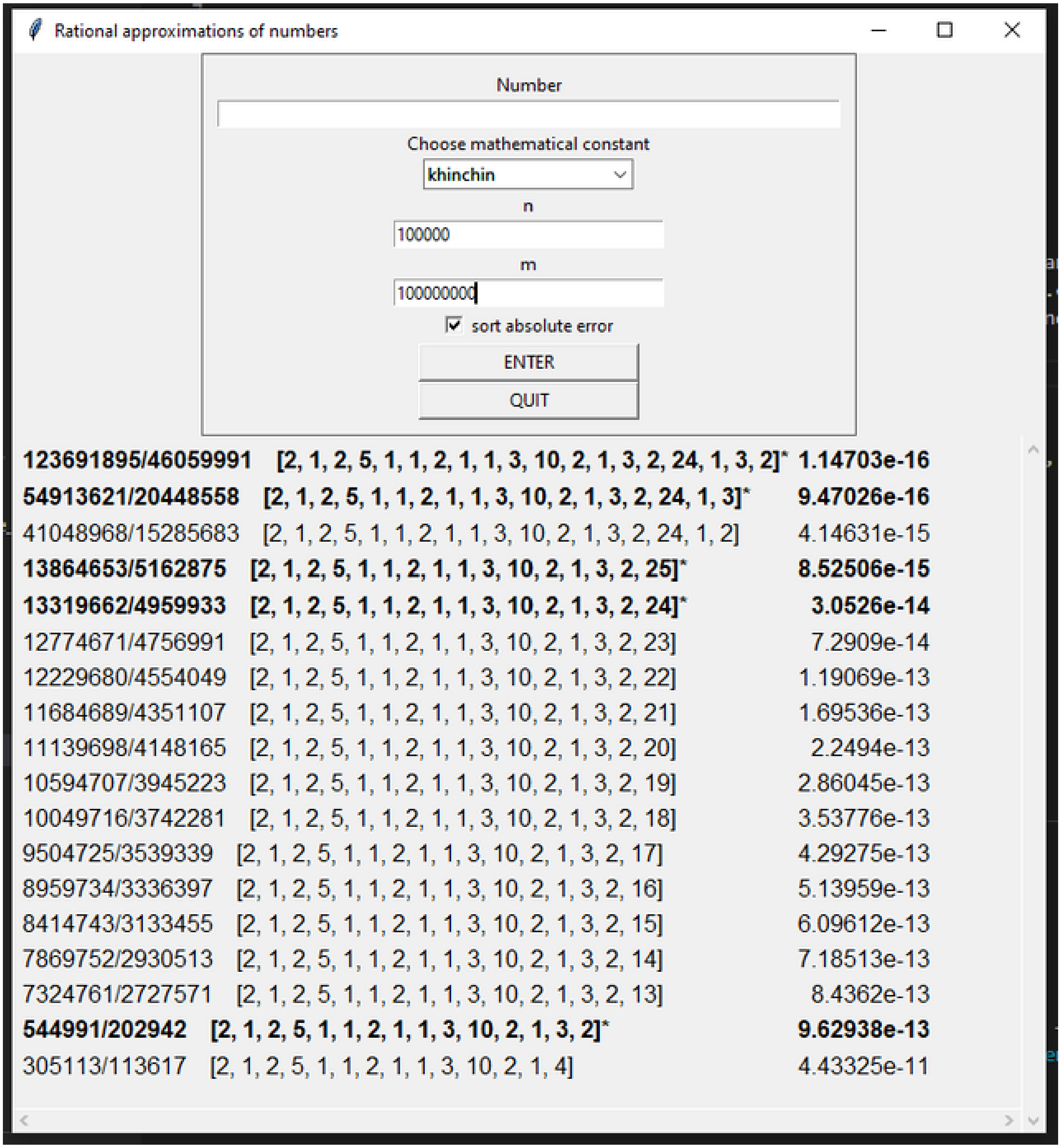
$$K_0 = \prod_{r=1}^{\infty} \left(1 + \frac{1}{r(r+2)}\right)^{\log_2 r} \approx 2.6854520010\dots$$

Gore navedeni limes važi za skoro sve realne brojeve, to jest, ukoliko izaberemo slučajan realan broj, verovatnoća da njegov verižni zapis nema ovu osobinu je jednaka nuli (što ne znači da je nemoguća, na primer, broj e je izuzetak). Zanimljivo je da se ne zna da li je sama Hičinova konstanta takav izuzetak.

Na slici 4. su prikazani rezultati aproksimacije za Hičinovu konstantu, gde imenioci idu do 100 000. Na sledećoj slici su prikazane aproksimacije kada je izabran opseg imenioca 100 000 - 100 000 000. U oba slučaja su razlomci sortirani tako da se prvo prikazuju oni sa najmanjom apsolutnom greškom. Za izabranu tačnost od 1000 decimala je ovo ujedno i gornja granica sa kojom program može da izračuna rezultate dovoljno brzo.


Rational approximations of numbers
Number
Choose mathematical constant
khinchin
n
1
m
100000
sort absolute error
ENTER
QUIT
239878/89325 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3]* 5.42011e-11
174643/65033 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 2] 2.26346e-10
65235/24292 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 3]* 4.06653e-10
44173/16449 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2]* 2.09598e-9
21062/7843 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10]* 5.65539e-9
19013/7080 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 9] 2.36642e-8
16964/6317 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 8] 4.60234e-8
14915/5554 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 7] 7.45259e-8
12866/4791 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 6] 1.12107e-7
2049/763 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3]* 1.61451e-7
1477/550 [2, 1, 2, 5, 1, 1, 2, 1, 1, 2] 2.54439e-6
572/213 [2, 1, 2, 5, 1, 1, 2, 2]* 5.99168e-6
333/124 [2, 1, 2, 5, 1, 1, 3]* 3.18699e-5
239/89 [2, 1, 2, 5, 1, 1, 2]* 5.87426e-5
94/35 [2, 1, 2, 5, 2]* 0.000262285
51/19 [2, 1, 2, 6]* 0.00124147
43/16 [2, 1, 2, 5]* 0.002048
35/13 [2, 1, 2, 4] 0.00685569
27/10 [2, 1, 2, 3] 0.014548
8/3 [2, 1, 2]* 0.0187853
5/2 [2, 2] 0.185452
3/1 [3]* 0.314548
2/1 [2]* 0.685452

Slika 5.


Rational approximations of numbers
Number
Choose mathematical constant
khinchin
n
100000
m
100000000
sort absolute error
ENTER
QUIT
123691895/46059991 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 24, 1, 3, 2]* 1.14703e-16
54913621/20448558 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 24, 1, 3]* 9.47026e-16
41048968/15285683 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 24, 1, 2] 4.14631e-15
13864653/5162875 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 25]* 8.52506e-15
13319662/4959933 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 24]* 3.0526e-14
12774671/4756991 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 23] 7.2909e-14
12229680/4554049 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 22] 1.19069e-13
11684689/4351107 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 21] 1.69536e-13
11139698/4148165 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 20] 2.2494e-13
10594707/3945223 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 19] 2.86045e-13
10049716/3742281 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 18] 3.53776e-13
9504725/3539339 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 17] 4.29275e-13
8959734/3336397 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 16] 5.13959e-13
8414743/3133455 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 15] 6.09612e-13
7869752/2930513 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 14] 7.18513e-13
7324761/2727571 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2, 13] 8.4362e-13
544991/202942 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 3, 2]* 9.62938e-13
305113/113617 [2, 1, 2, 5, 1, 1, 2, 1, 1, 3, 10, 2, 1, 4] 4.43325e-11

Slika 6.

6. Zaključak

U poglavlju 2. je objašnjena veza između racionalnih aproksimacija i verižnih zapisa. Predstavljen je algoritam koji efikasno na osnovu verižnog zapisa broja određuje njegove racionalne aproksimacije prve i druge vrste. Algoritam dat u $[1]$ je izuzetno efikasan i omogućava da nađemo jako veliki broj verižnih i međuverižnih aproksimacija čak i u programskom jeziku kao što je Python (koji je za red veličina sporiji nego Java, C++ i slični jezici) i da uz to imamo i grafički interfejs.

7. Literatura

[1] Branko Malešević: Racionalne aproksimacije realnih brojeva i neke primene, Nastava matematike, XLIII, 3, 1998.
[2] Fredrik Johansson and others. *mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 0.18)*, December 2013. http://mpmath.org/
[3] Branko Malešević, Luka Milinković: Verižni razlomci i primene, Simpozijum MATEMATIKA I PRIMENE, Matematički fakultet, Univerzitet u Beogradu, 2014.