

pisanje funkcije biopython

May 12, 2025

```
[2]: seznam = [1, "nekaj", 200, "se nekaj", ["a", 3, "nekaj", 4], "karkoli"]
```

```
[3]: seznam[0]
```

```
[3]: 1
```

```
[4]: seznam[-1]
```

```
[4]: 'karkoli'
```

```
[5]: seznam[-2]
```

```
[5]: ['a', 3, 'nekaj', 4]
```

```
[6]: len(seznam)
```

```
[6]: 6
```

```
[7]: seznam[6]
```

```
-----  
IndexError                                Traceback (most recent call last)  
Cell In[7], line 1  
----> 1 seznam[6]  
  
IndexError: list index out of range
```

```
[8]: seznam[5]
```

```
[8]: 'karkoli'
```

1 Izračunaj povprečno maso oseb v datoteki podatki.txt

```
[11]: vsota_teze = 0
      stevilo_oseb = 0
      for vrstica in open("podatki.txt"):
          ime, teza, visina = vrstica.split()
          teza = int(teza)

          vsota_teze += teza # vsota_teze = vsota_teze + teza
          stevilo_oseb += 1

      povprecje = vsota_teze / stevilo_oseb
      print(povprecje)
```

65.75

```
[13]: f = open("povprecje.txt", "w")
```

```
[14]: f.write(povprecje)
```

```
-----
TypeError                                Traceback (most recent call last)
Cell In[14], line 1
----> 1 f.write(povprecje)

TypeError: write() argument must be str, not float
```

```
[15]: str(povprecje)
```

```
[15]: '65.75'
```

```
[16]: f.write(str(povprecje))
```

```
[16]: 5
```

```
[17]: f.close()
```

```
[21]: with open("povprecje.txt", "w") as f:
      f.write('nekaj\n')
      f.write('še nekaj\n')
```

2 Izračunaj povprečno maso in zapisi rezultat v povprecje.txt

```
[27]: vsota_teze = 0
      stevilo_oseb = 0
      for vrstica in open("podatki.txt"):
          ime, teza, visina = vrstica.split()
          teza = int(teza)

          vsota_teze += teza # vsota_teze = vsota_teze + teza
          stevilo_oseb += 1

      povprecje = vsota_teze / stevilo_oseb

      # print(povprecje)
      with open("povprecje.txt", "w") as f:
          f.write(str(povprecje)+"\n")
```

3 Izračunaj povprečno maso, gradi seznam mas

```
[32]: vse_teze = []
      for vrstica in open("podatki.txt"):
          ime, teza, visina = vrstica.split()
          teza = int(teza)

          vse_teze.append(teza)

      povprecje = sum(vse_teze) / len(vse_teze)
      print(povprecje)
```

65.75

```
[30]: vse_teze
```

```
[30]: [56, 62, 78, 67]
```

```
[31]: sum(vse_teze)
```

```
[31]: 263
```

```
[33]: min(vse_teze)
```

```
[33]: 56
```

```
[34]: max(vse_teze)
```

```
[34]: 78
```

4 Preberi zaporedja v datoteki query-dna.txt in poročaj za vsak gen, število pojavitev baze t

```
[36]: for vrstica in open("datoteke/query-dna.txt"):
      gen, sekvenca = vrstica.split()
      print(gen)
      print(sekvenca)
```

```
ASDF13
gcaactgttggacggctacagtgcggttggtagaactgagtcggtttaaggactcacacatcgcggggtctgcaaagtgt
aatctacaaggagagcccgag
SDFG14
cgaagggcaatcggaagttgaggttcgtcatattaagtttggggaacgccgacatctaaatcttttaggtgataaatgcc
taaatcagattcaatgtatt
DFGH15
cgacctcgtaaaatgacaaactgtcgtggagcagatttcggtcatgccgccgagccctaccaatcgagttcaactat
cgctaactcgcgatgagcct
FGHJ16
tcgcggttagccacagccgggcctgattacagaggggtgaattcgatgcttgatgcggattcctggtaagctccgccgt
gcgaccgacaactctcgact
GHJK17
tggtgatgtgtacatctttgaaaggctcacctgaacaaaagtgtattacaatcaacgagcccaggagctgatccctc
aacaagggcacccaagaagt
HJKL18
tacagacatatcgctcccgtagctggaggatttcacatgatctaagcaaagccgtagtgggagttcctatggcaataag
cgaccttctataaccgagag
ZXCV19
tcatgcatgttaggttacatctaggctatgcctgtcccagtcagcaggtgggcctagattaagaaatgccgggttaggca
acacaacaccggtgtccttt
XCVB20
cgctacgtatgtccctaataagggtcatggtgctagccagggtcggggctagtttttaaggattttctgcccccaaca
aggagccagataggcccctt
CVBN21
acttggccactttactcctgcaatcttttagtcctggggggagtttaaaatcattccagctgggatgggtctctatcctct
ccctgtaataacaacaacg
VBNM22
tggtctttaagattaactgctcattagatctgtctccaaactgttaccgccggaatcacaggagaatcagtcacct
aagttgcgtaggcatatcc
```

```
[49]: zap = ↵
      ↪ "cgctacgtatgtccctaataagggtcatggtgctagccagggtcggggctagtttttaaggattttctgcccccaacaaggagccaga"
```

```
[50]: zap.count('t')
```

```
[50]: 25
```

```
[42]: stevci = {}  
      for crka in zaporedje:  
          if crka not in stevci:  
              stevci[crka] = 0  
          stevci[crka] += 1
```

```
[43]: stevci
```

```
[43]: {'c': 27, 'g': 27, 't': 25, 'a': 21}
```

5 funkcije

```
[46]: def stevilo_nukleotidov(zaporedje):  
      stevci = {} # lokalna spremenljivka  
      for crka in zaporedje:  
          if crka not in stevci:  
              stevci[crka] = 0  
          stevci[crka] += 1  
      return stevci
```

```
[51]: stevilo_nukleotidov(zap)
```

```
[51]: {'c': 27, 'g': 27, 't': 25, 'a': 21}
```

```
[48]: stevilo_nukleotidov("tctcagagatt")
```

```
[48]: {'t': 4, 'c': 2, 'a': 3, 'g': 2}
```

```
[58]: for vrstica in open("datoteke/qwerty-dna.txt"):  
      gen, sekvenca = vrstica.split()  
      print(gen)  
      #print(stevilo_nukleotidov(sekvenca)) # to ni ok, želimo le t  
  
      #frekvence = stevilo_nukleotidov(sekvenca)  
      #stevilo_tjev = frekvence["t"]  
      #print(stevilo_tjev)  
  
      # bolj "pythonovsko"  
      print(stevilo_nukleotidov(sekvenca)["t"])
```

```
ASDF13  
21  
SDFG14  
30  
DFGH15  
21
```

FGHJ16
21
GHJK17
20
HJKL18
23
ZXCV19
26
XCVB20
25
CVBN21
30
VBNM22
27

```
[54]: frekvence = stevilo_nukleotidov("tctcagagatt")
```

```
[55]: frekvence
```

```
[55]: {'t': 4, 'c': 2, 'a': 3, 'g': 2}
```

```
[56]: frekvence["t"]
```

```
[56]: 4
```

6 Za dano zaporedje preštej pojavitve 3-merov (3-terk)

```
[60]: zap = "acttgccactttactcctgcaatcttttagtcctg"
```

```
[62]: zap[0:3]
```

```
[62]: 'act'
```

```
[63]: zap[1:4]
```

```
[63]: 'ctt'
```

```
[64]: zap[2:5]
```

```
[64]: 'ttg'
```

```
[75]: stevci = {}  
for i in range(0, len(zap)-3+1):  
    terka = zap[i:i+3]  
    if terka not in stevci:  
        stevci[terka] = 0
```

```
stevci[terka] += 1
```

```
[76]: stevci
```

```
[76]: {'act': 3,
      'ctt': 3,
      'ttg': 1,
      'tgg': 1,
      'ggc': 1,
      'gcc': 1,
      'cca': 1,
      'cac': 1,
      'ttt': 2,
      'tta': 2,
      'tac': 1,
      'ctc': 1,
      'tcc': 2,
      'cct': 2,
      'ctg': 2,
      'tgc': 1,
      'gca': 1,
      'caa': 1,
      'aat': 1,
      'atc': 1,
      'tct': 1,
      'tag': 1,
      'agt': 1,
      'gtc': 1}
```

```
[67]: zap
```

```
[67]: 'acttggccactttactcctgcaatcttttagtcctg'
```

```
[84]: def stevilo_kterk(zap, k=3):
      stevci = {}
      #k = 3
      for i in range(0, len(zap)-k+1):
          terka = zap[i:i+k]
          if terka not in stevci:
              stevci[terka] = 0
          stevci[terka] += 1
      return stevci
```

```
[78]: stevilo_kterk("atgatgggc", 3)
```

```
[78]: {'atg': 2, 'tga': 1, 'gat': 1, 'tgg': 1, 'ggg': 1, 'ggc': 1}
```

```
[79]: stevilo_kterk("atgatgggc", 2)
```

```
[79]: {'at': 2, 'tg': 2, 'ga': 1, 'gg': 2, 'gc': 1}
```

```
[80]: stevilo_kterk("atgatgggc", 1)
```

```
[80]: {'a': 2, 't': 2, 'g': 4, 'c': 1}
```

```
[81]: stevilo_kterk("atgatgggc", 5)
```

```
[81]: {'atgat': 1, 'tgatg': 1, 'gatgg': 1, 'atggg': 1, 'tgggc': 1}
```

```
[85]: stevilo_kterk("atgatgggc")
```

```
[85]: {'atg': 2, 'tga': 1, 'gat': 1, 'tgg': 1, 'ggg': 1, 'ggc': 1}
```

```
[86]: stevilo_kterk("atgatgggc", 5)
```

```
[86]: {'atgat': 1, 'tgatg': 1, 'gatgg': 1, 'atggg': 1, 'tgggc': 1}
```

7 Preberi query-dna.txt in vrni gen z največ pojavitvami terke agt

```
[113]: najvec_pojavitev = 0
najvec_gen = ""
ciljna_kterka = 'tgatcc'
for vrstica in open("datoteke/qwerty-dna.txt"):
    gen, sekvenca = vrstica.split()
    #print(gen)
    frekvence = stevilo_kterk(sekvenca, len(ciljna_kterka))
    if ciljna_kterka in frekvence:
        #print(frekvence['atg'])
        if frekvence[ciljna_kterka] > najvec_pojavitev:
            najvec_pojavitev = frekvence[ciljna_kterka]
            najvec_gen = gen

print(najvec_gen, najvec_pojavitev)
```

GHJK17 1

```
[89]: imenik = {'Ana': '041777222', 'Berta': '031535323'}
```

```
[90]: imenik['Ana']
```

```
[90]: '041777222'
```

```
[92]: imenik['Katarina']
```



```
-----  
KeyError                                Traceback (most recent call last)  
Cell In[92], line 1  
----> 1 imenik['Katarina']  
  
KeyError: 'Katarina'
```

```
[93]: if 'Katarina' in imenik:  
      print(imenik['Katarina'])
```

```
[94]: if 'Ana' in imenik:  
      print(imenik['Ana'])
```

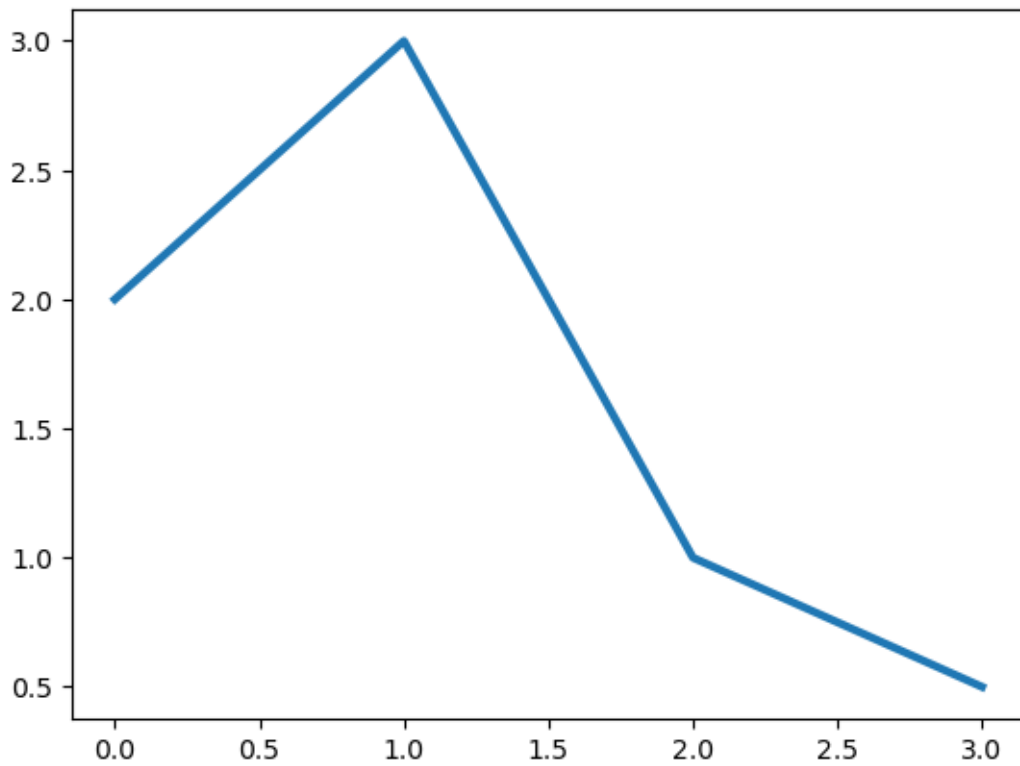
041777222

```
[ ]:
```

```
[114]: import matplotlib.pyplot as plt
```

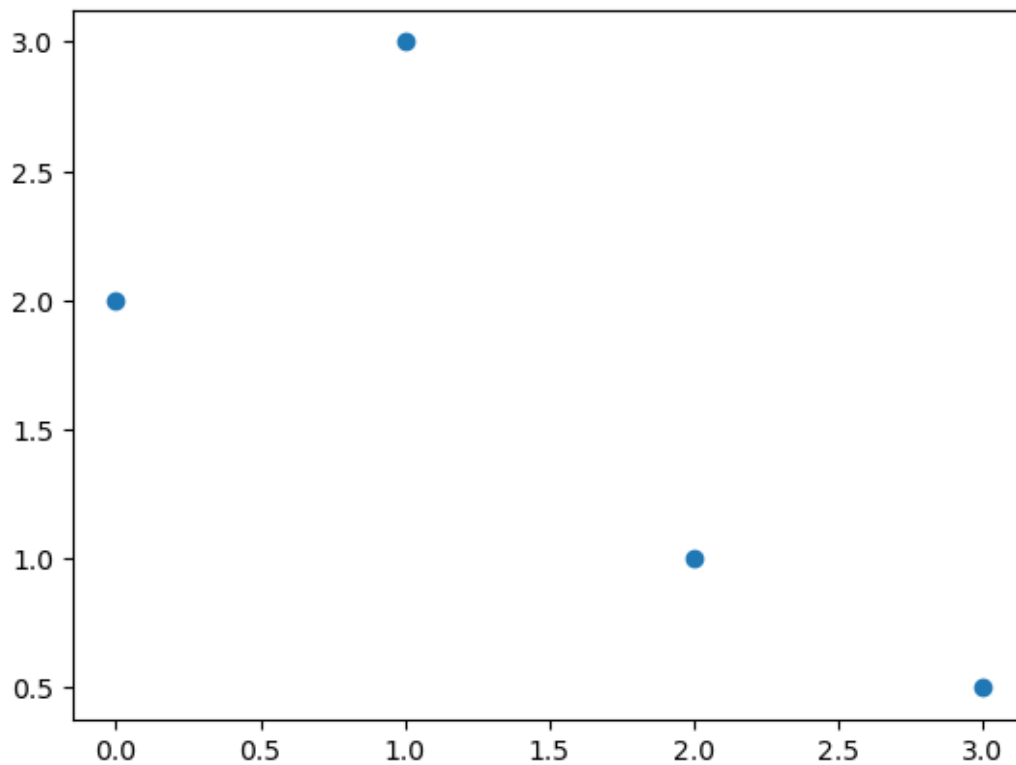
```
[115]: plt.plot([2,3,1,0.5], lw=3)
```

```
[115]: [<matplotlib.lines.Line2D at 0x7f77e536f410>]
```



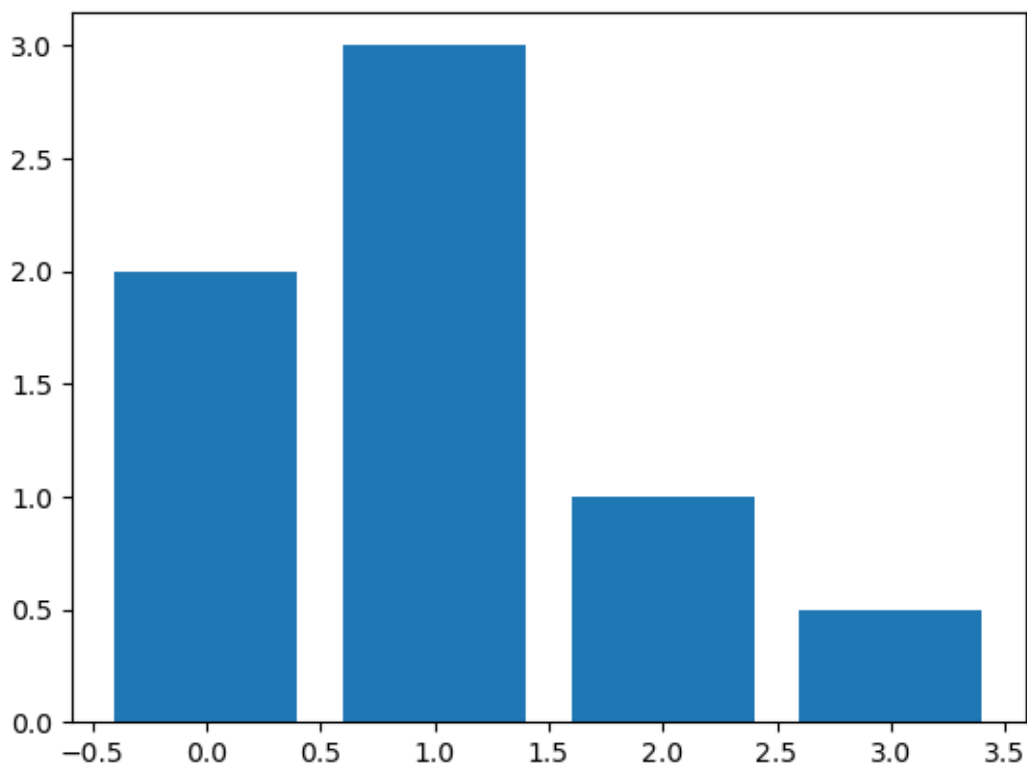
```
[116]: plt.plot([2,3,1,0.5], 'o')
```

```
[116]: [<matplotlib.lines.Line2D at 0x7f77e5203b90>]
```



```
[117]: plt.bar([0,1,2,3], [2,3,1,0.5])
```

```
[117]: <BarContainer object of 4 artists>
```



```
[118]: zap = "gcaactgttggacggctacagtgacggttggttagaactgagtcggtttaaggactcacacatcgcggtctgcaaagtgtaatctacaag"
```

8 Za dano zaporedje, izriši graf deleža G%+C% v drsečem oknu dolžine 20

```
[144]: delezi_na_zaporedju = []
for i in range(0, len(zap)-20+1, 1):
    zap_v_oknu = zap[i:i+20]
    frek = stevilo_nukleotidov(zap_v_oknu)

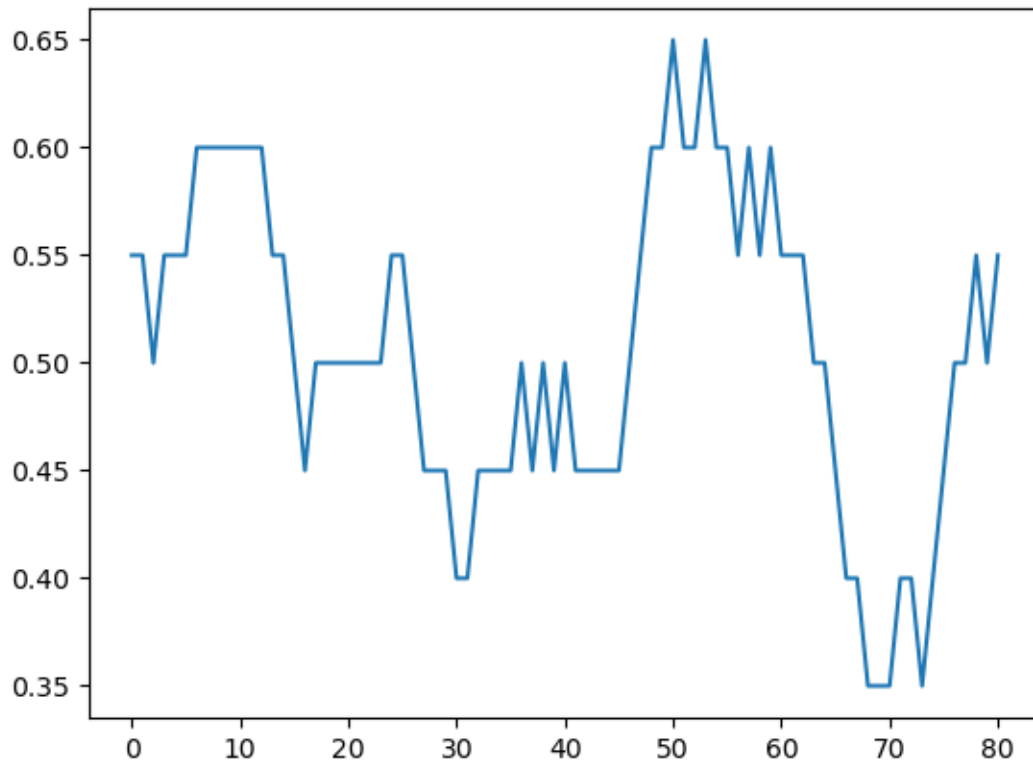
    delez_G = frek.get('g', 0)/20
    delez_C = frek.get('c', 0)/20

    delez_G_ali_C = delez_G + delez_C

    delezi_na_zaporedju.append(delez_G_ali_C)
```

```
[145]: plt.plot(delezi_na_zaporedju)
```

```
[145]: [<matplotlib.lines.Line2D at 0x7f77e4e0b190>]
```



```
[123]: imenik
```

```
[123]: {'Ana': '041777222', 'Berta': '031535323'}
```

```
[124]: imenik['Ana']
```

```
[124]: '041777222'
```

```
[125]: imenik['Katarina']
```

```
-----  
KeyError                                Traceback (most recent call last)  
Cell In[125], line 1  
----> 1 imenik['Katarina']  
  
KeyError: 'Katarina'
```

```
[127]: imenik.get('Katarina', '')
```

```
[127]: ''
```

```
[137]: list(range(0, 10))
```

```
[137]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
[138]: list(range(0, 10, 1))
```

```
[138]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
[139]: list(range(0, 10, 2))
```

```
[139]: [0, 2, 4, 6, 8]
```

```
[140]: list(range(0, 10, 3))
```

```
[140]: [0, 3, 6, 9]
```

```
[141]: list(range(0, 10, 4))
```

```
[141]: [0, 4, 8]
```

```
[ ]:
```