

Python – vježbe 4

24. maj 2021.

Rad sa stringovima

1. Napišite program koji od korisnika traži da unese string. Program zatim treba da štampa sljedeće:
 - Ukupan broj karaktera u stringu
 - Sting ponovljen 10 puta
 - Prvi karakter u stringu
 - Prva tri karaktera iz stringa
 - Poslednja tri karaktera iz stringa
 - String sa karakterima u obrnutom redosledu (palindrom)
 - Sedmi karakter u stringu ako string ima dovoljnu dužinu, a u suprotnom štampati poruku o grešci
 - String kod koga su izostavljeni prvi i poslednji karakter
 - String sa svim velikim slovima
 - String u kojem je svako slovo a zamijenjeno sa e
 - String u kojem je svako slovo zamijenjeno razmakom (bjelinom)
2. Jedan jednostavan način da se procijeni broj riječi u nekom stringu je da se prebroji broj razmaka (space) u stringu. Napišite program koji traži od korisnika da unese tekst, a onda mu vraća procijenjeni broj riječi u tom stringu.
3. Napišite program koji traži od korisnika da unese formulu i štampa poruku da li formula ima isti broj lijevih i desnih zagrada.
4. Napišite program koji traži od korisnika da unese riječ i koji štampa da li ta riječ sadrži samoglasnike ili ne.
5. Napišite program koji od korisnika traži da unese string. Program treba da kreira novi string od unijetog stringa tako da se drugi karakter unijetog stringa zamijeni sa zvjezdicom i da se na kraj stringa dodaju tri znaka uzvika. Na kraju treba štampati novi string. Tipičan izlaz može da izgleda ovako:
Unesite string: Australija
A*stralijs!!!
6. Napišite program koji traži od korisnika da unese string **s** kojem sva velika slova konvertuje u mala, i iz kojeg uklanja sve tačke i zareze. Program na kraju štampa rezultat.
7. Napišite program koji traži od korisnika da unese string i koji zatim određuje da li je unijeti string palindrom ili ne. Palindrom je string koji se čita jednako slijeva na desno kao i zdesna na lijevo (na primjer: **Ana voli Milovana**).

8. Na fakultetu se e-mail adrese studenata završavaju sa @student.ac.me, dok se adrese profesora završavaju sa @prof.ac.me. Napišite program koji najprije traži od korisnika koliko će email adresa unijeti, a zatim korisnik unosi te adrese. Nakon što su sve adrese unijete, program treba da štampa poruku da li su sve adrese bile studentske ili je bilo i nekih profesorskih adresa.

9. Napisati program koji od korisnika traži da unese broj, a zatim štampa, po sljedećem šablonu, spisak brojeva u kojem je uneseni broj posljednji:

```
1
 2
   3
    4
```

10. Napišite program koji traži od korisnika da unese string, a zatim štampa svako slovo iz stringa duplirano (ponovljeno) i u posebnom redu. Na primjer, ako je korisnik unio POZ, izlaz treba da bude ovakav:

```
PP
OO
ZZ
```

11. Napišite program koji traži od korisnika da unese string koji sadrži slovo a. Program treba da štampa sljedeće dve linije: u prvoj liniji treba da štampa dio stringa do prvog pojavljivanja slova a uključujući i slovo a, a u drugoj liniji treba da štampa ostatak stringa. Na primjer:

```
Unesite string: digitalna obrada signala
digita
lna obrada signala
```

12. Napišite program koji traži od korisnika da unese riječ a onda konvertuje u veliko slovo svako drugo slovo iz riječi. Na primjer, ako korisnik unese riječ **sOfisticiran**, program štampa: **sOfIsTiCiRaN**.

13. Napišite program koji traži od korisnika da unese dva stringa iste dužine. Program treba da provjeri da li su unijeti stringovi iste dužine. Ako nijesu, program štampa odgovarajuću poruku i završava rad. Ako jesu iste dužine program treba da štampa alternativno slova iz oba stringa. Na primjer, ako je korisnik unio stringove **tabcde** i **ABCDE**, program će štampati **AaBbCcDdEe**.

14. Napišite program koji od korisnika traži da unese svoje ime i prezime malim slovima, i koji prva slova imena i prezimena pretvara u velika.

15. Napraviti generator priče. Korisnik unosi nekoliko riječi, koje se zatim ubacuju u neku priču. Riječi su obično iz nekoliko kategorija, na primjer, mjesta, životinje itd. Program pravite tako što napravite priču iz koje izostavite neke riječi. Program zatim traži od korisnika da unese te riječi iz saopštenih kategorija. Zatim se štampa priča sa dodatnim riječima. Na primjer:

```
Enter a college class: CALCULUS
Enter an adjective: HAPPY
Enter an activity: PLAY BASKETBALL
CALCULUS class was really HAPPY today. We learned how to
PLAY BASKETBALL today in class. I can't wait for tomorrow's class!
```

16. Napisati program koji traži unos imena i prezimena a zatim generiše reklamu:

```
Enter name: George Washington
```

```
Dear George Washington,
```

```
I am pleased to offer you our new Platinum Plus Rewards
card at a special introductory APR of 47.99%. George,
```

an offer like this does not come along every day, so I urge you to call now toll-free at 1-800-314-1592. We cannot offer such a low rate for long, George, so call right away.

17. Napisati program koji generiše tekst od 26 linija (koliko je slova abecede), čiji je dio prikazan ispod:

```
abcdefghijklmnopqrstuvwxyz  
bcdefghijklmnopqrstuvwxyz  
cdefghijklmnopqrstuvwxyzab  
itd.  
yzabcdefghijklmnopqrstuvwx  
zabcdefghijklmnopqrstuvwxy
```

18. Sljedeće zadatke riješiti bez operatora `in`, `count` i `index` metoda, već korišćenjem `for` i `if`:

- Bez korišćenja `in` operatora, napišite program koji traži od korisnika da unese string i jedan znak, pa da štampa da li se taj znak pojavljuje u stringu.
- Bez korišćenja `count` metode, napišite program koji traži od korisnika da unese string i jedan znak, pa da prebroji koliko se puta znak pojavljuje u stringu.
- Bez upotrebe `index` metoda, napišite program koji traži od korisnika da unese string, koji zatim štampa indeks prvog pojavljivanja znaka u stringu. Ako se znak ne pojavljuje u stringu, štampati odgovarajuću poruku.

19. Napišite program koji traži od korisnika da unese jedan veliki cio broj, i koji zatim dodaje separatore hiljada – zareze, što je uobičajeno ukoliko se broj odnosi na novac. Ako korisnik, na primjer, unese 1000000, izlaz će biti 1,000,000.

20. Anagram neke riječi je riječ koja se dobija kada se ispremještaju slova te riječi. Dva anagrama riječi *idle* su *deli* i *lied*. Napišite program koji traži od korisnika da unese string pa štampa jedan slučajan anagram, odnosno, slučajan raspored karaktera iz originalnog stringa.

21. Jedan jednostavan način za enkripciju poruka je premještanje slova. Jedan način premještanja je da se prvo uzmu sva slova na parnim pozicijama a zatim sva slova na neparnim pozicijama. Na primjer, string *message* bi bio kodiran kao *msaesg* jer su slova na parnim pozicijama *m*, *s*, *a*, *e* (sa indeksima 0, 2, 4, and 6) a slova sa neparnim pozicijama *e*, *s*, *g* (sa indeksima 1, 3, and 5).

- Napišite program koji traži od korisnika da unese string, pa korišćenjem ovog metoda štampa šifrovanu verziju stringa.
- Napravite program koji dešifruje string koji je bio šifrovan na gore opisan način.

22. Opštija verzija gornje tehnike šifriranja je takozvana „cik-cak” šifra, u kojoj se umjesto razdvajanja parnih i neparnih pozicija, razdvajanje vrši na tri, četiri ili više grupa. Na primjer, u slučaju razdvajanja na tri grupe, string „secret message” bi bila podijeljena u tri grupe. Prva grupa bi bila „sr sg“, odnosno slova sa indeksima 0, 3, 6, 9 i 12. Druga grupa „eemse”, od slova sa indeksima 1, 4, 7, 10, i 13. Poslednja, treća grupa „ctea“ od slova sa indeksima 2, 5, 8, i 11. Šifrovana poruka bi bila „sr sgeemsectea“.

- Napišite program koji traži od korisnika da unese string, pa korišćenjem „cik-cak” metode šifrira uneti tekst podelom na tri grupe.
- Napišite program za dešifrovanje podjele na tri grupe.
- Napišite program koji traži od korisnika da unese string i jedan cio broj kojim se kazuje na koliko grupa će biti podijeljena poruka (tri, četiri, ili bilo koji drugi broj). Šifrovati string „cik-cak” metodom.
- Napišite program koji dešifruje „cik-cak” u opštem slučaju.

23. U matematičkoj analizi, izvod od x^5 je $5x^4$. Izvod od x^N je Nx^{N-1} . Napišite program koji traži od korisnika da unese izraze kao što su x^3 ili x^5 i koji štampa njihov izvod.
24. U algebarskim izrazima se simbol za množenje često izostavlja, kao kod $3x + 4y$ ili $3(x + 5)$. U programskim jezicima neophodna je upotreba simbola za množenje, $3 * x + 4 * y$ ili $3 * (x + 5)$. Napišite program koji traži od korisnika da unese algebarski izraz i koji zatim umeće simbol za množenje tamo gdje je potrebno.
25. Napišite program koji traži od korisnika da unese naziv varijable, i koji provjerava i štampa da li je zadati naziv varijable u skladu sa Python pravilima imenovanja varijabli.

Rad sa stringovima – predlog rješenja

1. Napišite program koji od korisnika traži da unese string. Program zatim treba da štampa sljedeće:

- Ukupan broj karaktera u stringu
- Sting ponovljen 10 puta
- Prvi karakter u stringu
- Prva tri karaktera iz stringa
- Poslednja tri karaktera iz stringa
- String sa karakterima u obrnutom redosledu (palindrom)
- Sedmi karakter u stringu ako string ima dovoljnu dužinu, a u suprotnom štampati poruku o grešci
- String kod koga su izostavljeni prvi i poslednji karakter
- String sa svim velikim slovima
- String u kojem je svako slovo a zamijenjeno sa e
- String u kojem je svako slovo zamijenjeno razmakom (bjelinom)

```
s=input('Unesite string: ')

print('Ukupno je karaktera', len(s))
print('Ponovljeni string je:', s*10)
print('Prvi karakter stringa je: ', s[0])
print('Prva tri karaktera stringa su:', s[0:3])
print('Poslednja tri karaktera stringa su: ', s[-3:])
print('Obrnuti string je: ', s[::-1])
if len(s)>=7:
    print('Sedmi karakter je: ', s[6])
else:
    print('Sedmi karakter se ne moze stampati')
print('String sa izostavljenim karakterima: ', s[1:-1])
print(s.upper())
print(s.replace('a', 'e'))
s1=''
for i in range(len(s)):
    if s[i].isalpha():
        s1=s1+' '
    else:
        s1=s1+s[i]
print('String sa eliminisanim slovima: ', s1)
```

2. Jedan jednostavan način da se procijeni broj riječi u nekom stringu je da se prebroji broj razmaka (space) u stringu. Napišite program koji traži od korisnika da unese tekst, a onda mu vraća procijenjeni broj riječi u tom stringu.

```

s=input('Unesite string: ')
brojac=0
for kar in s:
    if kar==' ':
        brojac=brojac+1
print('Rijeci je:',brojac+1)

```

3. Napišite program koji traži od korisnika da unese formulu i štampa poruku da li formula ima isti broj lijevih i desnih zagrada.

```

s=input('Unesite string koji predstavlja formulu: ')
bl=0
bd=0
for i in range(len(s)):
    if s[i]=='(':
        bl=bl+1
    elif s[i]==')':
        bd=bd+1
if bl==bd:
    print('Isti je broj zagrada')
else:
    print('Formula nije korektna')
#moze se prosiriti za druge vrste zagrada

```

4. Napišite program koji traži od korisnika da unese riječ i koji štampa da li ta riječ sadrži samoglasnike ili ne.

```

s=input('Unesite string: ')
ind=0
for kar in s:
    if kar in 'aeou':
        ind=1
if ind==0:
    print('Nema samoglasnika')
else:
    print('Ima samoglasnika')

```

5. Napišite program koji od korisnika traži da unese string. Program treba da kreira novi string od unijetog stringa tako da se drugi karakter unijetog stringa zamijeni sa zvjezdicom i da se na kraj stringa dodaju tri znaka uzvika. Na kraju treba štampati novi string. Tipičan izlaz može da izgleda ovako:

```

Unesite string: Australija
A*stralija!!!

```

```

s=input('Unesite string: ')
s=s[0]+'*'+s[2:]+''''
print(s)

```

6. Napišite program koji traži od korisnika da unese string **s** kojem sva velika slova konvertuje u mala, i iz kojeg uklanja sve tačke i zareze. Program na kraju štampa rezultat.

```

s=input('Unesite string: ')
s=s.lower()
s=s.replace(' ','')
s=s.replace('.', '')
print(s)

```

7. Napišite program koji traži od korisnika da unese string i koji zatim određuje da li je unijeti string palindrom ili ne. Palindrom je string koji se čita jednako slijeva na desno kao i zdesna na lijevo (na primjer: **Ana voli Milovana**).

```
s=input('Unesite string: ')
s=s.lower()
s=s.replace(' ', '')
ind=0
s_obrnuti=s[::-1]
for i in range(len(s)):
    if s[i]!=s_obrnuti[i]:
        ind=1
if ind==0:
    print('U pitanju je palindrom')
else:
    print('Nije u pitanju palindrom')
```

8. Na fakultetu se e-mail adrese studenata završavaju sa @student.ac.me, dok se adrese profesora završavaju sa @prof.ac.me. Napišite program koji najprije traži od korisnika koliko će email adresa unijeti, a zatim korisnik unosi te adrese. Nakon što su sve adrese unijete, program treba da štampa poruku da li su sve adrese bile studentske ili je bilo i nekih profesorskih adresa.

```
broj=eval(input('Unesite broj adresa: '))
bp=0
bs=0
print('Unesite adrese: ')
for i in range(broj):
    adr=input()
    if adr[-10:]=='prof.ac.me':
        bp=bp+1
    elif adr[-13:]=='student.ac.me':
        bs=bs+1
print('Studentskih adresa je: ',bs)
print('Profesorskih adresa je: ',bp)
```

9. Napisati program koji od korisnika traži da unese broj, a zatim štampa, po sljedećem šablonu, spisak brojeva u kojem je uneseni broj posljednji:

```
1
 2
  3
   4
```

```
broj=eval(input('Unesite broj: '))
for i in range(broj):
    print(' '*i, broj)
```

10. Napišite program koji traži od korisnika da unese string, a zatim štampa svako slovo iz stringa duplirano (ponovljeno) i u posebnom redu. Na primjer, ako je korisnik unio POZ, izlaz treba da bude ovakav:

```
PP
OO
ZZ
```

```
s=input('Unesite string: ')
for i in range(len(s)):
    print(s[i]*2)
```

11. Napišite program koji traži od korisnika da unese string koji sadrži slovo a. Program treba da štampa sljedeće dve linije: u prvoj liniji treba da štampa dio stringa do prvog pojavljivanja slova a uključujući i slovo a, a u drugoj liniji treba da štampa ostatak stringa. Na primjer:

```
Unesite string: digitalna obrada signala
digita
lna obrada signala
```

Rješenje:

```
s=input('Unesite string: ')
prvo=s.index('a')
print(s[:prvo+1])
print(s[prvo+1:])
```

12. Napišite program koji traži od korisnika da unese riječ a onda konvertuje u veliko slovo svako drugo slovo iz riječi. Na primjer, ako korisnik unese riječ **sofisticiran**, program štampa: **sOfIsTiCiRaN**.

```
s=input('Unesite string: ')
s1=''
for i in range(0,len(s)):
    if i%2==1:
        s1=s1+s[i].upper()
    else:
        s1=s1+s[i]
print(s1)
```

13. Napišite program koji traži od korisnika da unese dva stringa iste dužine. Program treba da provjeri da li su unijeti stringovi iste dužine. Ako nijesu, program štampa odgovarajuću poruku i završava rad. Ako jesu iste dužine program treba da štampa alternativno slova iz oba stringa. Na primjer, ako je korisnik unio stringove **abcde** i **ABCDE**, program će štampati **AaBbCcDdEe**.

```
s1=input('Unesite prvi string: ')
s2=input('Unesite drugi string: ')
if len(s1)!=len(s2):
    print('Morate unijeti stringove iste duzine!')
else:
    for i in range(len(s1)):
        print(s2[i],s1[i],sep='',end='')
```

14. Napišite program koji od korisnika traži da unese svoje ime i prezime malim slovima, i koji prva slova imena i prezimena pretvara u velika.

```
s=input('Unesite ime i prezime sa malim slovima: ')
poz=s.index(' ')
s1=s[0].upper()+s[1:poz+1]+s[poz+1].upper()+s[poz+2:]
print(s1)
```

15. Napraviti generator priče. Korisnik unosi nekoliko riječi, koje se zatim ubacuju u neku priču. Riječi su obično iz nekoliko kategorija, na primjer, mjesta, životinje itd. Program pravite tako što napravite priču iz koje izostavite neke riječi. Program zatim traži od korisnika da unese te riječi iz saopštenih kategorija. Zatim se štampa priča sa dodatnim riječima. Na primjer:

```
Enter a college class: CALCULUS
Enter an adjective: HAPPY
Enter an activity: PLAY BASKETBALL
CALCULUS class was really HAPPY today. We learned how to
PLAY BASKETBALL today in class. I can't wait for tomorrow's class!
```

16. Napisati program koji traži unos imena i prezimena a zatim generiše reklamu:

```
Enter name: George Washington
```

```
Dear George Washington,
```

```
I am pleased to offer you our new Platinum Plus Rewards
card at a special introductory APR of 47.99%. George,
an offer like this does not come along every day, so I
urge you to call now toll-free at 1-800-314-1592. We
cannot offer such a low rate for long, George, so call
right away.
```

17. Napisati program koji generiše tekst od 26 linija (koliko je slova abecede), čiji je dio prikazan ispod:

```
abcdefghijklmnopqrstuvwxy
zabcdefghijklmnopqrstuvwxya
cdefghijklmnopqrstuvwxyzab
itd.
yzabcdefghijklmnopqrstuvwx
zabcdefghijklmnopqrstuvwxy
```

Rješenje:

```
s='abcdefghijklmnopqrstuvwxy
z'
for i in range(len(s)):
    print(s[i:], s[:i], sep='')
```

18. Sljedeće zadatke riješiti bez operatora `in`, `count` i `index` metoda, već korišćenjem `for` i `if`:

- Bez korišćenja `in` operatora, napišite program koji traži od korisnika da unese string i jedan znak, pa da štampa da li se taj znak pojavljuje u stringu.
- Bez korišćenja `count` metode, napišite program koji traži od korisnika da unese string i jedan znak, pa da prebroji koliko se puta znak pojavljuje u stringu.
- Bez upotrebe `index` metoda, napišite program koji traži od korisnika da unese string, koji zatim štampa indeks prvog pojavljivanja znaka u stringu. Ako se znak ne pojavljuje u stringu, štampati odgovarajuću poruku.

19. Napišite program koji traži od korisnika da unese jedan veliki cio broj, i koji zatim dodaje separatore hiljada – zareze, što je uobičajeno ukoliko se broj odnosi na novac. Ako korisnik, na primjer, unese 1000000, izlaz će biti 1,000,000.

20. Anagram neke riječi je riječ koja se dobija kada se ispremještaju slova te riječi. Dva anagrama riječi idle su deli i lied. Napišite program koji traži od korisnika da unese string pa štampa jedan slučajan anagram, odnosno, slučajan raspored karaktera iz originalnog stringa.

```
from random import randint
anagram=''
rijec=input('Unesite rijec: ')
duzina=len(rijec)
for i in range(duzina):
    poz=randint(0, len(rijec)-1)
    anagram=anagram+rijec[poz]
    rijec=rijec[:poz]+rijec[poz+1:]
print(anagram)
```


21. Jedan jednostavan način za enkripciju poruka je premještanje slova. Jedan način premještanja je da se prvo uzmu sva slova na parnim pozicijama a zatim sva slova na neparnim pozicijama. Na primjer, string *message* bi bio kodiran kao *msaesg* jer su slova na parnim pozicijama *m*, *s*, *a*, *e* (sa indeksima 0, 2, 4, and 6) a slova sa neparnim pozicijama *e*, *s*, *g* (sa indeksima 1, 3, and 5).

(a) Napišite program koji traži od korisnika da unese string, pa korišćenjem ovog metoda štampa šifrovanu verziju stringa.

```
s=input('Unesite string: ')
pom1=''
pom2=''
for i in range(len(s)):
    if i%2==0:
        pom1=pom1+s[i]
    else:
        pom2=pom2+s[i]
poruka=pom1+pom2
print(poruka)
```

(b) Napravite program koji dešifruje string koji je bio šifrovan na gore opisan način.

22. Opštija verzija gornje tehnike šifriranja je takozvana „cik-cak” šifra, u kojoj se umjesto razdvajanja parnih i neparnih pozicija, razdvajanje vrši na tri, četiri ili više grupa. Na primjer, u slučaju razdvajanja na tri grupe, string „secret message” bi bila podijeljena u tri grupe. Prva grupa bi bila „sr sg“, odnosno slova sa indeksima 0, 3, 6, 9 i 12. Druga grupa „eemse”, od slova sa indeksima 1, 4, 7, 10, i 13. Poslednja, treća grupa „ctea“ od slova sa indeksima 2, 5, 8, i 11. Šifrovana poruka bi bila „sr sgeemsectea“.

(a) Napišite program koji traži od korisnika da unese string, pa korišćenjem „cik-cak” metode šifrira uneti tekst podelom na tri grupe.

(b) Napišite program za dešifrovanje podjele na tri grupe.

(c) Napišite program koji traži od korisnika da unese string i jedan cio broj kojim se kazuje na koliko grupa će biti podijeljena poruka (tri, četiri, ili bilo koji drugi broj). Šifrovati string „cik-cak” metodom.

(d) Napišite program koji dešifruje „cik-cak” u opštem slučaju.

23. U matematičkoj analizi, izvod od x^5 je $5x^4$. Izvod od x^{10} je $10x^9$. Napišite program koji traži od korisnika da unese izraze kao što su x^3 ili x^5 i koji štampa njihov izvod.

```
s=input('Unesite izraz:')
N=eval(s[2:])
print(N, '*', s[0:2], N-1, sep='')
```

24. U algebarskim izrazima se simbol za množenje često izostavlja, kao kod $3x + 4y$ ili $3(x + 5)$. U programskim jezicima neophodna je upotreba simbola za množenje, $3 * x + 4 * y$ ili $3 * (x + 5)$. Napišite program koji traži od korisnika da unese algebarski izraz i koji zatim umeće simbol za množenje tamo gdje je potrebno.

25. Napišite program koji traži od korisnika da unese naziv varijable, i koji provjerava i štampa da li je zadati naziv varijable u skladu sa Python pravilima imenovanja varijabli.