



PRAKTIKUM IZ PROGRAMIRANJA 1

VEŽBE 6

Milica Vasović, Jana Brzaković, Ana Vidosavljević



1. String je niz karaktera ograničen jednostrukim ili dvostrukim navodnicima
2. String mora početi i završiti se istim navodnicima
3. String se može štampati u više linija
 - Ako se započne sa tri navodnika (jednostruka ili dvostruka) i tako i završi
 - Ako se u string umetne specijalni karakter `\n`
4. Dva ili više stringova se spajaju u jedan korišćenjem operatora `+`
5. String se može pomnožiti celobrojnomo vrednošću
6. Delovima stringa može se pristupiti preko indeksa navedenog u zagradama `[]`



- Napisati program koji će ispisati sve podreči date reči.

Ulaz:	Izlaz:
abc	a
	ab
	abc
	b
	bc
	c



```
s = input()
for i in range(len(s)):
    for j in range(i, len(s)):
        print(s[i:j+1])
```



- Napisati program kojim se za dve date reči proverava da li je druga sadržana u prvoj, ako jeste odrediti prvu poziciju na kojoj se druga reč pojavljuje u prvoj.

```
s1 = input()
s2 = input()

indeks = s1.find(s2)

if(indeks != -1):
    print(indeks)
else:
    print('Rec "%s" ne sadrzi rec "%s"' %(s1, s2))
    //print('Rec "' + s1 + '" ne sadrzi rec "' + s2 + '"')
```



- Napišite program koji traži od korisnika da unese tekst, a onda mu vraća procenjeni broj reči u tom stringu.
- Napišite program koji traži od korisnika da unese string. Program treba da kreira novi string `novi_string` od unetog stringa tako da se drugi znak unetog stringa zameni sa zvezdicom i da se na kraj stringa dodaju tri znaka uzvika. Na kraju treba štampati `novi_string`. Tipičan izlaz može da izgleda ovako:

```
Unesite string: Qbert  
Q*ert!!!
```

- Napisati program koji učitava jedan string koji predstavlja korisničko ime i lozinku korisnika razdvojenih `_`. Ukoliko je dužina korisničkog imena manja od 3 karaktera, potrebno je izvršiti modifikaciju korisničkog imena i „duplirati“ ga.
- Učitavati stringove sa tastature sve dok se ne učitava prazan string. Za svaki string štampati da li je palindrom.



- Napisati program koji određuje koliko puta se data niska javlja kao podniska druge niske. Više pojavljivanja podniske se mogu preklapati.
- Reči se u rečnicima ređaju po abecedi. Napisati program koji upoređuje dve reči na osnovu leksikografskog poretka.
 - a) Sa razlikom između velikih i malih slova
 - b) Bez razlike između velikih i malih slova

Na standardni izlaz ispisati -1 ako je prva u rečniku ispred druge, 1 ako je druga ispred prve, tj. 0 ako su reči jednake.



- Napišite program koji traži od korisnika da unese tekst, a onda mu vraća procenjeni broj reči u tom stringu.



```
x = input("Uneti text: ")  
  
print ("Procenjeni broj reci u tekstu je: %d"  
%(x.count(' ')+1))
```

ili

```
print ("Procenjeni broj reci u tekstu je: ",  
x.count(' ')+1)
```



- Napišite program koji traži od korisnika da unese string. Program treba da kreira novi string novi_string od unetog stringa tako da se drugi znak unetog stringa zameni sa zvezdicom i da se na kraj stringa dodaju tri znaka uzvika. Na kraju treba štampati novi_string. Tipičan izlaz može da izgleda ovako:

Unesite string: Qbert
Q*ert!!!



```
x = input("Uneti text: ")
novi = x.replace(x[1], "*", 1)
novi_string = novi + "!!!"
print(novi_string)
```



- Napisati program koji učitava jedan string koji predstavlja korisničko ime i lozinku korisnika razdvojenih `_`. Ukoliko je dužina korisničkog imena manja od 3 karaktera, potrebno je izvršiti modifikaciju korisničkog imena i „duplirati“ ga.



```
x = input()
ime = ""
lozinka = ""
for i in range(0, len(x)):
    if x[i] == '_':
        ime = x[0:i]
        lozinka = x[i + 1: len(x)]
        break
if len(ime) < 3:
    ime = ime * 2
print(ime + " " + lozinka);
```



- Učitavati stringove sa tastature sve dok se ne učitava prazan string. Za svaki string štampati da li je palindrom.



```
def daLiJePalindrom(s):  
    p = True  
    for i in range(0, len(s)//2):  
        if s[i] != s[-(i+1)]:  
            p = False  
            break  
    return p  
  
def Stringovi():  
    s = input("Unesi rec: ")  
    while s != "":  
        if(daLiJePalindrom(s)):  
            print("Jeste palindrom")  
        else:  
            print("Nije palindrom")  
        s = input("Unesi rec: ")
```

Stringovi()



- Napisati program koji određuje koliko puta se data niska javlja kao podniska druge niske. Više pojavljivanja podniske se mogu preklapati.



```
podniska = input()
niska = input()
broj_pojavljivanja = 0

for p in range(len(niska) - len(podniska) + 1):
    if podniska == niska[p : p + len(podniska)]:
        broj_pojavljivanja += 1

print(broj_pojavljivanja)
```



```
podniska = input()
niska = input()

broj_pojavljivanja = 0
p = niskafind(podniska)

while p != -1:
    broj_pojavljivanja += 1
    p = niskafind(podniska, p + 1)

print(broj_pojavljivanja)
```



- Reči se u rečnicima ređaju po abecedi. Napisati program koji upoređuje dve reči na osnovu leksikografskog poretka.

a) Sa razlikom između velikih i malih slova

b) Bez razlike između velikih i malih slova

Na standardni izlaz ispisati -1 ako je prva u rečniku ispred druge, 1 ako je druga ispred prve, tj. 0 ako su reči jednake.

Primer za a):

Ulaz:	Izlaz:
abcde	-1
abcefg	

Primer za b):

Ulaz:	Izlaz:
aBcdE	-1
abCEfg	



```
def poredi(s1, s2):  
    for i in range(min(len(s1), len(s2))):  
        if s1[i] != s2[i]:  
            return ord(s1[i]) - ord(s2[i])  
  
    return len(s1) - len(s2)  
  
rec1 = input()  
rec2 = input()  
  
p = poredi(rec1, rec2)  
  
if p < 0:  
    print(-1)  
elif p > 0:  
    print(1)  
else:  
    print(0)
```



```
rec1 = input()
rec2 = input()

if rec1 < rec2:
    print(-1)
elif rec1 > rec2:
    print(1)
else:
    print(0)
```



```
def poredi(s1, s2):  
    for i in range(min(len(s1), len(s2))):  
        if s1[i].upper() != s2[i].upper():  
            return ord(s1[i].upper()) - ord(s2[i].upper())  
  
    return len(s1) - len(s2)  
  
rec1 = input()  
rec2 = input()  
  
p = poredi(rec1, rec2)  
  
if p < 0:  
    print(-1)  
elif p > 0:  
    print(1)  
else:  
    print(0)
```



```
rec1 = input()
rec2 = input()

rec1 = rec1.upper()
rec2 = rec2.upper()

if rec1 < rec2:
    print(-1)
elif rec1 > rec2:
    print(1)
else:
    print(0)
```