**SVEUČILIŠTE U RIJECI  
ODJEL ZA INFORMATIKU**

**Preddiplomski studij informatike**

**Seminarski rad iz kolegija  
OSNOVE INFORMATIKE**

**TEHNIČKA DOKUMENTACIJA  
PYTHON – ZADATAK 1**

Autor: Dino AhmičićMentori: prof. dr. sc. Sanda Martinčić-Ipšić  
 dr. sc. Slobodan Beliga

U Rijeci, Siječanj 2021.

Sadržaj

[1. Funkcija za prebrojavanje riječi 2](#_Toc61632538)

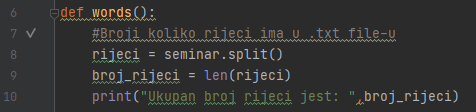
[2. Funkcija za prebrojavanje praznih znakova (razmaka) 3](#_Toc61632539)

[3. Funkcija za prebrojavanje brojeva svih vrsta 4](#_Toc61632540)

[4. Funkcija za prebrojavanje količinu riječi koje počinju sa slovom D 6](#_Toc61632541)

[5. Funkcija za prebrojavanje količinu riječi koje završavaju sa slovom A 8](#_Toc61632542)

# Funkcija za prebrojavanje riječi



Slika . Funkcija prebrojavanja riječi

U prvom zadatku ovoga rada tražilo se napraviti funkciju koja će prebrojiti ukupan broj riječi u seminarskom radu za grafičke kartice. Da bi definirali funkciju potrebno je napisati liniju koda koju možemo vidjeti u šestom retku. Funkciju definiramo varijablom “def” a zatim napišemo ime te funkcije, u ovom slučaju neka to bude “words()”.

U retku 8 definirali smo varijablu “rijeci” kojoj smo pridodali jednakost “seminar.split()”. “.split()” nam omogućava da nam čitavi string podijeli na liste, gdje je svaka riječ jedna stavka te liste.

U retku 9 definirali smo još jednu varijabu pod imenom “broj\_rijeci” te joj pridodali vrijednost “len(rijeci)”. Funkcija “len()” nam vraća vrijednost ukupne količine stavki u nekoj listi. U ovom slučaju, funkcija “len(rijeci)” nam vraća broj riječi u datoteci, te taj broj pridodaje varijabli “broj\_rijeci”.

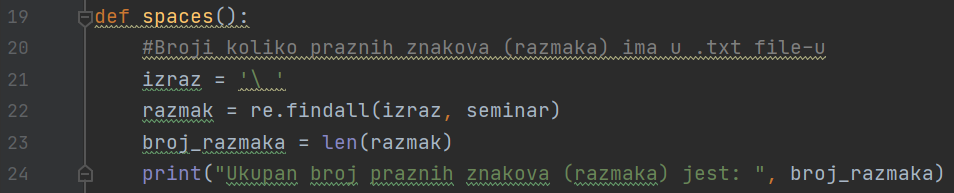
Na kraju ove funkcije koristimo “print()” s kojim na ekran ispisujemo ženjeni tekst i varijablu, u ovom slučaju ispisujemo tekst “Ukupan broj riječi jest: ” i varijablu “broj\_rijeci”.



Slika . Rezultat funkcije

Nakon poziva funkcije, na ekranu nam dolazi poruka koju smo ispisali funkcijom “print()”, te ona glasi “Ukupan broj riječi jest: 2295”.

# Funkcija za prebrojavanje praznih znakova (razmaka)



Slika . Funkcija prebrojavanja razmaka

U drugom zadatku zadatku morali smo napraviti funckiju koja nam prebrojava prazne znakove, odnosno razmake. Na isti način kao i prvu funkciju počinjemo sa “def”, te imenom funkcije, neka ovdje to bude “spaces()”.

U 21. liniji koda definirati ćemo varijablu izraz kojoj ćemo pridodati vrijednost regularnog izraza za razmak ‘\ ’. Sa ovime mi definiramo točno što tražimo u datoteci.

Nakon toga, definirali smo varijablu “razmak” te joj pridodali “re.finall(\*predefinirani reg. izraz\*, \*datoteka koju čitamo\*)” koja nam služi za pretraživanje prethodno definiranog regularnog izraza, u ovome slučaju je to razmak ‘ ’.

U liniji koda ispod smo definirali varijablu “broj\_razmaka” kojoj sa funkcijom “len(razmak)” pridodajemo broj pojavljivanja znaka razmaka.

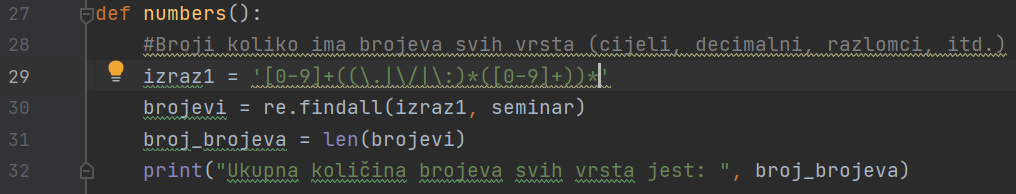
Na kraju, funkcijom print() na ekran ispisujemo željeni tekst te varijablu “broj\_razmaka”.



Slika . Rezultat funkcije

Nakon poziva funkcije, na ekran nam dolazi poruka pisana funkcijom “print()” koja glasi “Ukupan broj praznih znakova (razmaka) jest: 2072”.

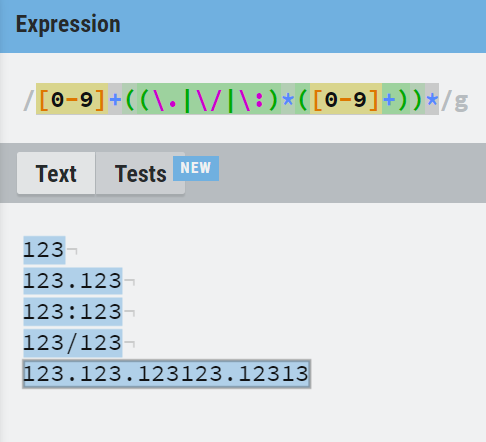
# Funkcija za prebrojavanje brojeva svih vrsta



Slika . Funkcija prebrojavanja brojeva

Kao treći zadatak morali smo napraviti funkciju koja će brojati količinu brojeva zapisanih u datoteku seminara. Funkciju smo nazvali “numbers()”.

Isto kao i prethodni zadatak, koristiti ćemo regularni izraz. Definirali smo varijablu “izraz1” te smo joj dodijelili regularni izraz koji glasi ' [0-9]+((\.|\/|\:)\*([0-9]+))\*'  
[0-9]+ izraz nam naznačuje da nam uhvati sve brojeve u rasponu od 0 do 9, te znakom ‘+’ naznačimo da se ti pojedinačni brojevi mogu ponavljati i više od jedanput (npr. 123).  
Izraz (\.|\/|\:)\* nam omogućava da uhvatimo sve brojeve nakon kojih slijedi jedan od znakova ‘.’ ‘/’ ‘:’ (točka, kosa crta, dvotočka), te znakom ‘\*’ na kraju zagrade naznačimo da se ti znakovi nemoraju nužno pojavljivati, no mogu i više puta (npr. 123. ili 123/ ili 123:).  
Izrazom ([0-9]+) možemo uhvatiti sve brojeve nakon prijašnjih znakova, koji mogu biti jednoznamenkasti ili višeznamenkasti. Na kraju izraza, poslije zadnje zagrade smo stavili znak ‘\*’ kojim naznačujemo da se svi spomenuti znakovi i brojevi poslije tih znakova nemoraju nužno pojavljivati.



Slika . regexr.com

Nakon što smo definirali izraz, funkcijom “re.findall(\*predefinirani izraz\*, \*datoteka iz koje čitamo\*)” čitamo datoteku, te nakon svakom uspješnog pronalaska broja, vrijednost zapisujemo u novu varijablu zvanu “brojevi”.

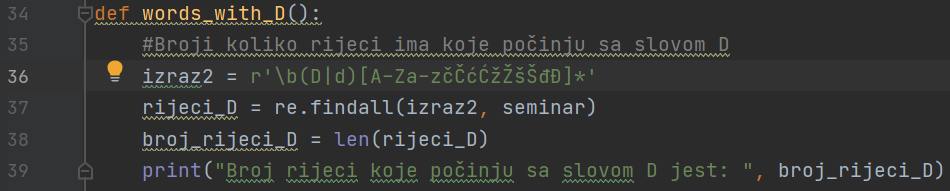
U 31. liniji koda novoj varijabli zvanoj “broj\_brojeva” funkcijom “len(brojevi)” prepisujemo frekvenciju pojavljivanja brojeva svih oblika.

Na kraju, funkcijom “print()” ispisujemo poruku na ekran.



Slika . Rezultat funkcije

# Funkcija za prebrojavanje količinu riječi koje počinju sa slovom D



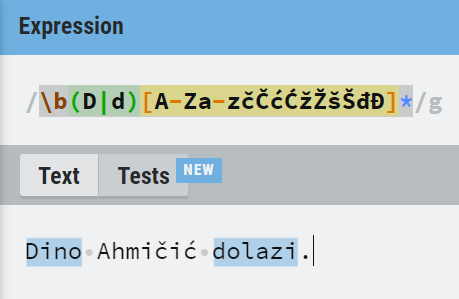
Slika . Funkcija prebrojavanja riječi koje počinju sa slovom D

Kao sljedeći zadatak morali smo naopraviti funkciju koja nam prebrojava riječi koje počinju sa prvim slovom moga imena, slovom D. Funkciju smo definirali i nazvali kao “words\_with\_D()”.

Varijabli “izraz1” dodijelili smo regularni izraz '\b(D|d)[A-Za-zčČćĆžŽšŠđĐ]\*'.

Izraz \b(D|d) nam pokupi sve riječi koje imaju slovo D ili d kao početno (\b nam označava početak riječi)

[A-Za-zčČćĆžŽšŠđĐ]\* nam pokupi sva slova u riječi koja slijede poslije slova D ili d.  
A-Za-z nam obuhvaća sva slova engleske abecede, čČćĆžŽšŠđĐ nam obuhvaća dijakritičke znakove u hrvatskoj abecedi.



Slika . regexr.com

Nakon što smo definirali naš regularni izraz, potrebno nam je funkcijom “re.findall(izraz, seminar)” prošetati kroz datoteku te uhvatiti sve riječi koje počinju sa slovom D ili d, te ih zapisati u varijablu “rijeci\_D”.

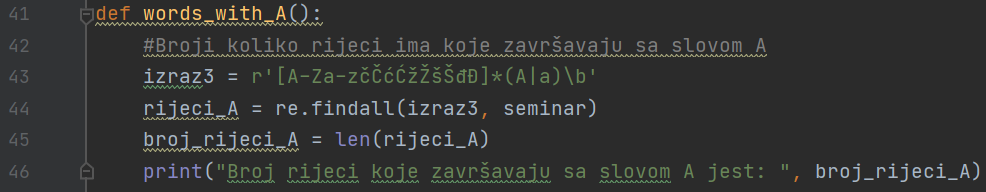
U novu varijablu “broj\_rijeci\_D” spremili smo količinu riječi koje počinju sa slovom D ili d funkcijom “len(rijeci\_D)”.

Na kraju funkcije, opet koristimo “print()” da bi na ekran ispisali željenu poruku.



Slika . Rezultat funkcije

# Funkcija za prebrojavanje količinu riječi koje završavaju sa slovom A



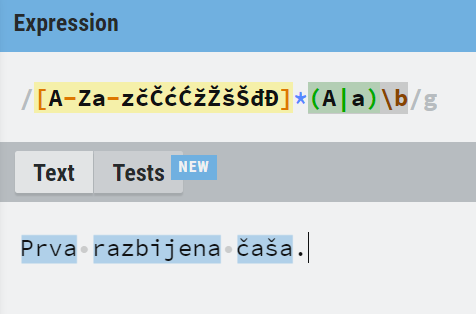
Slika . Funkcija prebrojavanja riječi koje završavaju sa slovom A

Definiramo funkciju i damo joj naziv, neka to bude “words\_with\_A()”.

Kao i prijašnji zadatak moramo definirati regularni izraz, ovdje je to   
‘[A-Za-zčČćĆžŽšŠđĐ]\*(A|a)\b’.

[A-Za-zčČćĆžŽšŠđĐ]\* izraz nam hvata riječi koje započinju sa svim slovima engleske abecede i dijakritičkim znakovima, te znakom ‘\*’ naznačimo da se ta slova mogu ponavljati i više puta.

Izrazom (A|a)\b uhvatimo riječi koje završavaju sa slovima A ili a (\b nam ovdje stoji na kraju izraza, te nam naznačuje na slova A i a na kraju riječi).



Slika . regexr.com

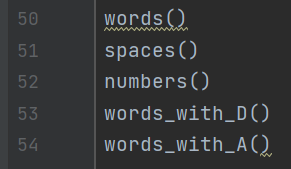
Kao što smo i do sada tako radili, funkcijom “re.findall(izraz3, seminar)” čitamo datoteku i u varijablu “rijeci\_A” zapisujemo takve pronađene riječi.

Funkcijom “len(rijeci\_A)” u varijablu “broj\_rijeci\_A” zapisujemo količinu takvih riječi.

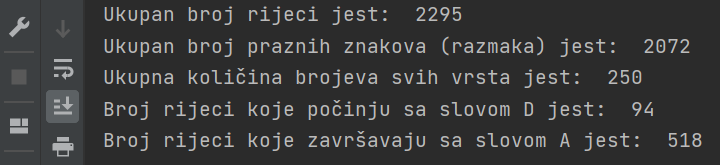
Funkcijom “print(“Broj riječi koje završavaju sa slovom A jest: ”, broj\_rijeci\_A)” ispisujemo poruku na ekran.



Slika . Rezultat funkcije



Slika . Pozivi svih funkcija



Slika . Svi rezultati