

Gimnazija Andrije Mohorovičića

Web Crawler

Python projekt

Napravio: Petar Dušević, 2.5



2020.

Sadržaj:

1 Uvod	2
1.1 Što je webcrawler?	2
1.2 Kako funkcioniraju?	2
1.3 Efikasnost webcrawlera i povijest	2
1.4 Kako ja mogu napraviti jedan webcralwer?	3
2 Upute za korištenje	4
2.1 Izvor	4
2.1.1 Animacije	4
2.1.2 Odabir načina rada	4
2.1.3 Obabir ključa.....	4
2.1.4 Početni link	4
2.1.5 Kopanje	5
2.1.6 Konzola	5
2.1.7 Help tab	6
2.2 Format_maker	6
3 Opis izrade programa	7
3.1 Skromni početci	7
3.2 Odrastanje programa	7
3.3 Izrada autmoatskog načina rada	7
3.4 Izrada manualnog moda	7
3.5 Interaktivnost programa	7
3.6 Ni a ni b, već cijeli file format	8
3.7 Pretraživanje.....	8
3.8 Uljepšavanje programa	8
3.9 I šlag na kraju	9
4 Tehnički podatci	10
5 Postavke konzole	10
6 LITERATURA (slike):	11

1 Uvod

1.1 Što je webcrawler?

Webcrawler je program koji služi za pretraživanje interneta, najčešće u svrhu indeksiranja web stranica i automatiziranog skupljanja podataka sa interneta. Program pronade podatke, zapiše gdje ih je našao i ode na drugu stranicu tražiti podatke. Velike tražilice poput Googlea, Binga, Yahooa itd. koriste webcrawlere kako bi updateale ili prikupljale baze podataka za svoje usluge. Webcrawleri također pronalaze sve relevantne linkove sa stranice kako bi znali na koji bi link trebali skočiti dalje. Ovisno o zahtjevima programera i/ili kompanije, webcrawlere sa stranica mogu skupljati bilo što od određenih riječi, do naslova, cijelih paragrafa ili čak slika.



1.2 Kako funkcioniraju?

Webcrawler krene pretraživanje interneta sa listom linkova sa kojima će krenuti izvlačiti podatke, webcrawler oda na svaki od tih linkova, te skine sve podatke koji mu trebaju, to se

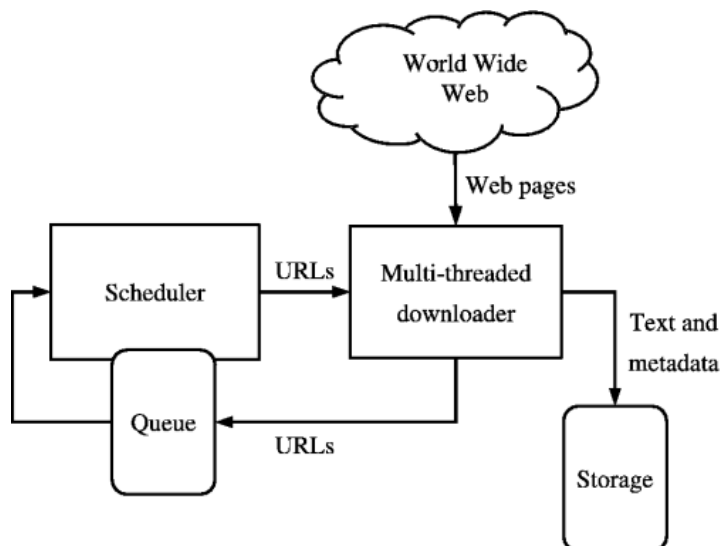


obično sastoji ili od skidanja cijeloga HTML koda, ili izoliranja samo bitnih podataka pa brisanja ostatka stranice, svaki webcrawler iz HTML koda izvlači linkove koji se nalaze u njoj. Nakon toga od svih tih linkova program mora odabrati koji su mu linkovi bitni, a koji nisu previše bitni ili bi mogli biti slijepe ulice. To je bitno zato što program treba znati izbjegavati

stranice koje nisu previše bitne kako bi mogao biti efikasniji. Program također treba znati koje je stranice posjetio, kako bi ih mogao izbjegavati da se nebi vrtio u krug.

1.3 Efikasnost webcrawlera i povijest

Danas webcrawleri koji služe za osvježavanje web stranice google moraju biti vrlo efikasni, kada se neka stranica promjeni, očekuje se da se više manje instantno promjene prikazuju na googleu. Nekoć davno, takvo je nešto bilo nemoguće za napraviti, kad je google tek počeo rad na svojoj tražilici tijekom početka stoljeća, programu je trebalo nekoliko mjeseci da prođe kroz cijeli internet, zato što bi webcrawler postupno prolazio kroz sve stranice dok nebi



završio, te onda opet ispočetka. Sa vremenom je google napravio veliku arhivu linkova uz pomoć koje su mogli odvojiti bitne linkove od onih manje bitnih linkova te su određene web stranice počeli osvježavati češće. Google je sa vremenom usavršio svoj internet crawling sistem i trenutno funkcionira po poprilično kompliciranim procesima koji mu omogućuju skoro pa maksimalnu

efikasnost. Danas je ogromna efikasnost bitna radi jako velike količine informacija na internetu i stalnih promjena koje se događaju na njemu.

1.4 Kako ja mogu napraviti jedan webcralwer?

Za napraviti webcrawler ne treba previše programerskog znanja, ja sam svoj webcrawler odlučio napraviti u programu Python 3.8 zato što je Python vrlo pogodan za ove stvari (a i takav je bio zadatak projekta). Webcrawler kojeg sam ja odlučio napraviti nije trebao biti previše kompliciran, a ni baš ludo efikasan radi ograničenja kompjutera na kojemu radim.



Program sam zamislio tako da ću u njega unijeti link i on će ići po internetu krećući od tog linka i tražiti neki tekstualni podatak koji mu dam (riječ, fraza, rečenica, slovo, itd.). Zatim bi mi program trebao izbaciti sve linkove koje je posjetio i reći kojiko je zadanih riječi našao na kojem linku. Program sam isplanirao da će raditi kao konzolna aplikacija u retro stilu.

2 Upute za korištenje

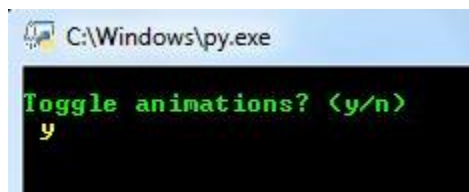
2.1 Izvor

Dakle kako bi otvorili program trebate samo dvaput kliknuti na datoteku `izvor.py`



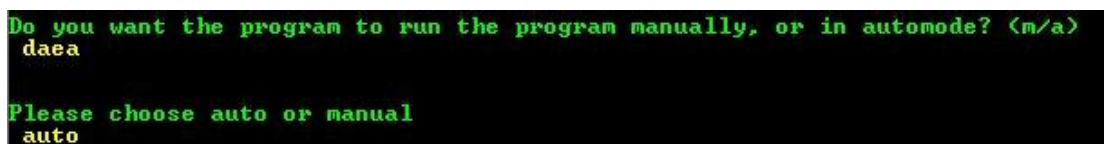
2.1.1 Animacije

Prva stvar koju će vas ovaj program pitati je želite li koristiti animacije, na to pitanje trebate odgovoriti sa 'yes' ili 'no', ako odaberete 'yes' program će prikazivati animacije ali će svi procesi trajati malo sporije, a ako odaberete 'no' onda ćete isključiti sve animacije i program će raditi maksimalnom brzinom koju može dostignuti.



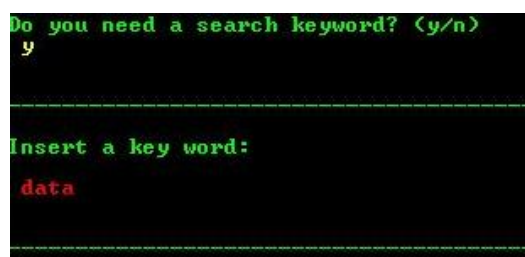
2.1.2 Odabir načina rada

Nakon pokretanja program će vas pitati o načinu rada. Za automode napišete 'automode' ili bilo koju skraćenicu te riječi, a za manual napišete isto bilo koji skraćenicu te riječi, ukoliko programu nije jasno što ste odabrali pitat će vas da ponovo upišete riječ. Ako odaberete automode program će vas pitati koliko linkova želite da posjeti, tu trebate unesti neki broj koji nebi trebao biti prevelik (ne savjetujem više od 20 odjednom). Ukoliko odaberete manual ići ćete direktno na slijedeći dio procesa.



2.1.3 Obabir ključa

Ključ (eng. key) je riječ koju će program tražiti po linkovima preko kojih bude prelazio. Prvo ćete biti upitani treba li vam ključ, ako kažete da vam netreba ćete nastaviti na slijedeći dio procesa, a ako kažete da trebate ćete morati unesti ključ.



2.1.4 Početni link

Nakon odabira ključa napisati ćete URL stranice od koje želite da program krene raditi to može biti bilo koja stranica, ali morate unesti točan URL te stranice, dakle <https://www.google.com> je točan link, dok www.google.com ili google.com nebi bili linkove s kojima bi ovaj program mogao rukovati. Naime treba napomenuti da postoje skraćenice. Dok sam radio ovaj program trebao sam ga stalno testirati te zato postoje dvije riječi koje će se ponašati kao linkovi, ukoliko napišete 'google' ili 'youtube' program će otići na google ili youtube, no to su apsolutno jedine dvije skraćenice koje zasada postoje u programu.

2.1.5 Kopanje

Kada program završi sa svojim kopanjem javiti će vam tri stvari:

- Koliko je linkova otkriveno
- Koliko je linkova posjećeno
- Koliko je stringova koji su isti kao ključ pronađeni

Nakon toga će vas pitati gdje da nastavi ići dalje. Na automodeu ćete trebati unijeti broj

```
-----
Insert URL:
https://www.youtube.com

List of links:
1 https://www.youtube.com/t/terms
2 https://www.youtube.com
3 https://www.youtube.com/new
4 https://www.youtube.com/yt/press/hr/
5 https://www.youtube.com/yt/copyright/hr/
6 https://www.youtube.com/t/contact_us
7 https://www.youtube.com/yt/creators/
8 https://www.youtube.com/yt/dev/hr/
9 https://www.youtube.com/yt/about/hr/
10 https://www.youtube.com/yt/advertise/
11 https://www.youtube.com/yt/policyandsafety/hr/
12 https://www.google.com/intl/hr/policies/privacy/

12 scraped link(s)
1 visited link(s)
0 key(s) found

<-h for the help tab>
What link next? <number>
```

linkova koje želite poslijetiti, a na manualu trebate unijeti redni broj linka na koji želite ići. Ukoliko ste na manualui program će vam pokazati listu svih prikupljenih linkova sa koje ćete birati, no ako ste na automodeu program vam neće automatski pokazati listu sakupljenih linkova jer nije bitna. Ukoliko je želite vidjeti trebate napisati '-l'. Postoji mogućnost da vam lista neće

stati u konzolu, u tome slučaju trebate napisati '-lp' te će vam program izbaciti .txt datoteku sa listom prikupljenih linkova te ćete moći birati iz nje.

Ukoliko želite vidjeti linkove na kojima su pronađeni ključevi, trebate napisati '-v' u konzolu.

2.1.6 Konzola

Ovaj program ima posebnu funkciju za unos podataka koja se zove Konzola, ona služi tome kako bi ste pri procesu kopanja mogli unositi komande u program koje vam pomažu u radu. Komande su označene znakom '-' na prvom mjestu.

Komanda	Skraćenica	Funkcija
-help	-h	Otvora help tab
-exit / -halt / -stop	-x	Zaustavlja program
-clear	-cls	"Očisti" konzolu
-switch	-s	Promjeni mode (auto<>manual)
-list	-l	Prikazuje listu prikupljenih linkova

-list reset	-lr	Resetira listu prikupljenim linkova
-list print	-lp	Napravi .txt datoteku sa prikupljenim linkovima
-list save	-ls	Spremi listu sa prikupljenim linkovima
-list load	-ll	Učita listu sa prikupljenim linkovima
-visited	-v	Prikazuje listu posjećenih linkova
-visited reset	-vr	Resetira listu posjećenih linkova
-visited print	-vp	Napravi .txt datoteku sa posjećenim linkovima
-visited save	-vs	Spremi listu sa posjećenim linkovima
-visited load	-vl	Učita listu sa posjećenim linkovima

2.1.7 Help tab

Na help tabu mogu se saznati razno razne informacije o programu. Kako bi ste vidjeli informacije samo trebate upisati jedan od brojeva koje vam help nudi, a kako bi ste izašli iz help taba trebate upisati broj '0'.

```
What link next? <number>
-h

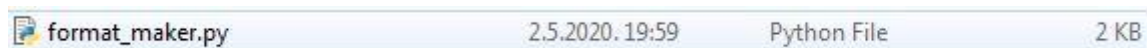
-----

Help tab:
0: Exit help
1: URL inserting help
2: Lists help
3: Console commands
4: Automode and Manual mode explained
5: Keys help

Choose an option:
```

2.2 Format_maker

Ovaj je program služi za konvertiranje .petar datoteka i poprilično ga je jednostavno koristiti. Sve što trebate je staviti ga u mapu sa datotekama koje želite konvertirati, kada pokrenete program pita vas želite li konvertirati .txt u .petar ili obrnuto, nakon što odaberete, trebate upisati ime datoteke koju želite konvertirati i ime datoteke gdje želite konvertirati. Program će završiti i izaći automatski.



3 Opis izrade programa

3.1 Skromni počeci

Kako bi krenuo izradu programa trebao sam istražiti kako bih mogao skidati podatke sa interneta koristeći se pythonom, nakon malo istraživanja naletio sam na python paket koji se zove urllib4 koji služi za interakciju python programa i web stranica na internetu. Koristeći taj paket mogao sam skinuti kod stranice, no moj program i dalje nije mogao razumjeti što ti podatci znače, za to sam koristio jedan drugi paket koji se zvao BeautifulSoup4 (bs4) i koji služi za to da bi program mogao izvlačiti informacije iz HTML koda. Koristeći ta dva paketa sam napravio prvu verziju programa koja je bila u stanju izvući sve linkove sa stranice koju bi ste joj dali.

3.2 Odrastanje programa

Nakon što sam napravi glavni kod programa trebalo ga je automatizirati. Pošto ovaj program nisam radio s namjerom da arhiviram cijeli internet, odlučio sam u program staviti dva načina rada, jedan način bi bio automatski gdje bi program krenuo raditi a jedan bi bio manualni, gdje bi korisnik sam određivao na koji bi link webcrawler išao. Također sam tu odlučio da ću svoj webcrawler nazvati Izvor, zato što su webcrawleri izvori podataka za internet.

3.3 Izrada automatskog načina rada

Učiniti da program može raditi autonomno bio je jedan od najvećih izazova koje sam imao u cijelome programu, trebao sam reorganizirati sve funkcije programa tako da se nebi ispreplitala ili otvarale druge funkcije bez da sebe zatvore kako bi izbjegao rušenje programa. Također sam trebao osmisliti način na koji će Izvor odabirati linkove na koje će ići. Pošto ne treba proći cijeli internet, odlučio sam to napraviti tako da program bira nasumični broj koji označava člana u setu linkova koje je skupio i onda ide na taj link. Također sam morao napraviti da program je također može izbjegavati već posjećenei nedostupne ili nepostojeće linkove.

3.4 Izrada manualnog moda

Ovaj dio nije bio previše težak, sve što je u njemu trebalo je naći način da korisnik može točno odabrati link iz seta na koji će otići, to sam rješio pretvorbom seta u listu i onda ispisivanjem te liste sa rednim brojevima članova








3.5 Interaktivnost programa

Nakon ovoga trebalo je nadodati stvari u program koje bi ga učinile intuitivnijim za korištenje. Želio sam učiniti tako da program navodi korisnika kroz korištenje, te bi čak i osobi koja ne zna kako program funkcionira bilo jasno što treba napraviti da bi došla do željenog rezultata, za to sam dodao razne yes/no opcije, učinio sam da je jako teško slučajno ispasti iz konzole sa pogrešnim unosom te sam čak dodao help opciju gdje bi korisnik mogao pitati program o tome kako funkcionira i što treba učiniti. Jedna od stvari koju je također trebalo dodati bila je konzola, koja je uglavnom služila za to da bilo gdje možete upisivati

komande koje bi vam pomagale pri korištenju programa. Sa konzolom sam dodao opciju da se sprema i učitaju svi posjećeni linkovi, također sam napravio opciju i da se linkovi ispišu u .txt dokument koji bi bio čitak.

3.6 Ni a ni b, već cijeli file format

Kako bi spremao linkove odlučio sam napraviti svoj file format isključivo za ovaj program. File format nazvao sam .petar (jer se zovem Petar). File format sam dodao čisto iz razloga kako nebi nitko osim programa mogao pristupiti tome file formatu i pročitati što u njemu piše (osim ako stvarno želi). Podatke sam u svojim datotekama odlučio zapisivati koristeći se ASCII vrijednostima, svako slovo imalo je ASCII vrijednost podignutu za 5 (jer se zovem Petar) tako da bi tekst bio red znakova koji nebi imao smisla. Također sam u svojem fileu sve

 AI0.petar	25.4.2020. 21:35	PETAR File	1 KB	zapisivao u jednom redu, a nove redove ("\n") sam označavao sa ASCII znakom
 He.petar	7.5.2020. 19:39	PETAR File	1 KB	
 He1.petar	25.4.2020. 21:25	PETAR File	1 KB	
 He2.petar	1.5.2020. 2:10	PETAR File	1 KB	
 He3.petar	30.4.2020. 22:47	PETAR File	1 KB	
 He4.petar	30.4.2020. 21:50	PETAR File	1 KB	
 He5.petar	7.5.2020. 19:37	PETAR File	1 KB	

15 (' '). Unutar programa nadodao sam funkcije za kodiranje i dekodiranje .petar datoteka. Također sam uz program Izvor nadodao jedan program koji se zove format_maker koji omogućuje bilo kome da pretvara .txt datoteke u .petar datoteke i obratno (da bi se znatiželjni ljudi mogli malo zabaviti sa datotekama).

3.7 Pretraživanje

U program je također trebalo ubaciti opciju da se traži neki podatak koji mu vi date, taj podatak sam ja odlučio da će biti string koji će korisnik unesti i za kojim će ovaj program tragati kroz more podataka u internetu i govoriti vam gdje ga je našao. Taj sam kod samo ubacio pored djela za izvlačenje linkova i sve je radilo kako spada.

3.8 Uljepšavanje programa

Sad kad sam imao program koji funkcionira kako treba, trebalo ga je uljepšati, dodao sam fancy konzolne animacije, boje, animacije učitavanja, pa čak i na kraju logo koji se pokaže u obliku slova pri pokretanju programa koristeći se paketom pillow.

4 Tehnički podatci

Veličina programa: 85.6 KB

Veličina bez save fileova: 68.9 KB

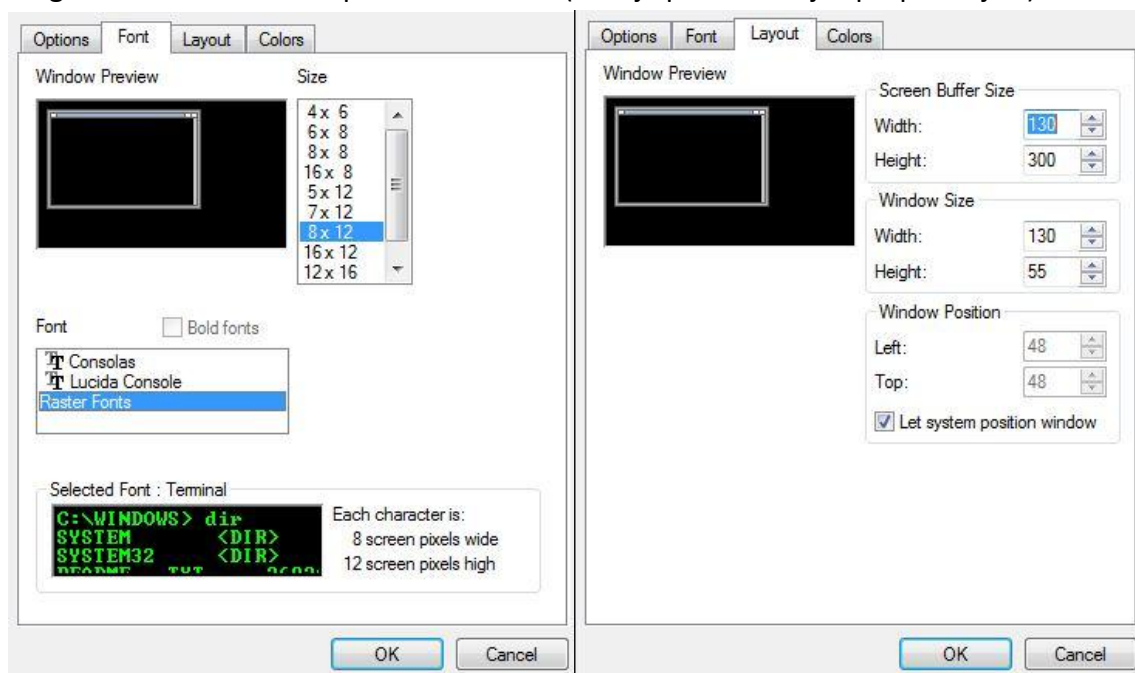
Veličina Izvora: 18.9 KB

Veličina Format_makera: 1.48 KB

Ovo možete pokretati na apsolutno svakome kompjuteru iz 21 stoljeća, stvarno neznam gdje ovako jednostavni program nebi radio

5 Postavke konzole

Za najbolje iskustvo pri korištenju programa klinite desnim klikom na okvir konzole i odaberite opciju "properties", font trebate staviti na "Raster Fonts" a veličinu fonta na 8x12. NA layout tabu trebate namjestiti "screen buffer width" na barem 100, a "screen buffer height" može biti bilo što po vašem izboru (iako je povećana brojka preporučljiva)



6 LITERATURA (slike):

<https://www.simplilearn.com/what-is-a-web-crawler-article>

<https://www.google.com>

<https://medium.com/kariyertech/web-crawling-general-perspective-713971e9c659>