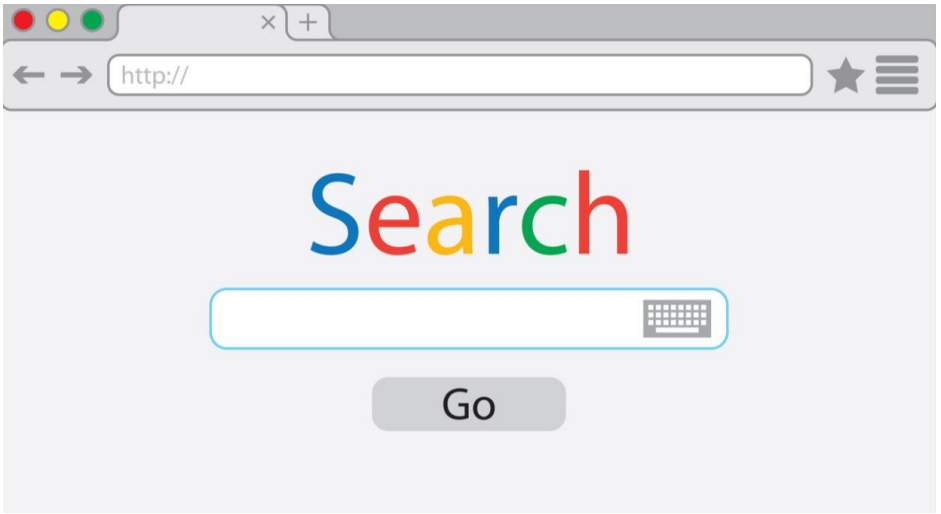
Specifikacija drugog projektnog zadatka

Algoritmi i strukture podataka 2021/2022

Search engine

Potrebno je napraviti mašinu za pretraživanje tekstualnih dokumenata (search engine). Program prilikom startovanja treba da obiđe stablo direktorijuma u fajl sistemu počevši od datog korena, da parsira tekstualne fajlove u njima i da izgradi strukture podataka potrebne za efikasno pretraživanje. Nakon toga, program omogućava korisniku da unosi tekstualne upite koji se sastoje od jedne ili više reči razdvojenih razmakom, pretražuje dokumente koristeći prethodno kreiranu strukturu podataka i korisniku ispisuje rangirane rezultate pretrage.

Test skup i alat za parsiranje možete preuzeti na sledećem linku ili na Teams-u u sekciji Files/Projekat2.



Za maksimalno 15 poena

Implementirati rangiranje rezultata pretrage tako da na rang rezultata utiče broj pojavljivanja traženih reči u dokumentu uz korišćenje proizvoljnih struktura podataka.

Za više od 15 poena

Na rangiranje, osim broja pojavljivanja traženih reči u dokumentu, treba utiče i broj linkova iz drugih dokumenata na pronađeni dokument, kao i broj traženih reči u dokumentima koji sadrže link na traženi dokument. Za organizovanje stranica koristiti graf. Za efikasnu pretragu reči na stranici koristiti strukturu podataka trie. Sve navedene strukture podataka se moraju samostalno implementirati.

Zahtevi

- Pri pokretanju programa potrebno je korisniku dati mogućnost da unese putanju do direktorijuma koji želi da pretraži. Neophodno je omogućiti i promenu direktorijuma za pretraživanje bez prekida rada programa.

- Potrebno je omogućiti više uzastopnih pretraga bez ponovnog kreiranja pomoćnih struktura podataka.

- Potrebno je podržati ispravnu upotrebu logičkih operatora AND, OR i NOT prilikom formiranja upita. Pri čemu upit može imati samo jedan logički operator.
  - Logički operator AND zahteva prisustvo obe navedene reči u stranicama koje su u rezultujućem skupu. Obe reči treba u jednakoj meri da utiču na ukupan rang (npr. python AND sequence).
  - Logički operator OR zahteva prisustvo bar jedne reči u stranicama koje čine rezultujući skup (npr. set OR graph).
  - Logički operator NOT se u ovom slučaju smatra BINARNIM. Dakle, predviđa se upotreba u obliku 'python NOT java' gde se pojavljivanje reči python zahteva u stranicama rezultujućeg skupa, dok se reč java u stranicama rezultujućeg skupa ne sme pojaviti.

- Ukoliko korisnik unosi upit sastavljen od više reči, rangiranje stranica po svakoj pojedinačnoj reči utiče na sveukupno rangiranje određene stranice. U ovom slučaju, ne treba insistirati na prisustvu svake od reči u rezultatima.
  Primer:
  - python – potrebno je pronaći sve dokumente koji sadrže reč python
  - python programming language – potrebno je pronaći sve dokumente koji sadrže bar jednu od reči python, programming, odnosno language

- Rezultati upita treba da budu putanje do HTML stranica (iz test skupa) sortirane po izračunatom rangu. Odabir optimalnog algoritma za sortiranje se prepušta studentima, pri čemu studenti samostalno implementiraju odabrani algoritam za sortiranje. Prva stranica po rangu treba da sadrži i kratak kontekst (isečak iz dela stranice) u kome je tražena reč pronađena. Detalji implementacije prikaza konteksta u slučaju prikazivanja rezultata upita sa logičkim operatorima se prepuštaju studentu.

- Broj rezultujućih stranica koje se u jednom trenutku prikazuju prepušta se studentima na izbor, a potrebno je obezbediti mogućnost da se broj dinamički promeni od strane korisnika.

Dodatni bodovi

Druga unapređenja navedenih funkcionalnosti mogu doneti **do 5 dodatnih poena**.

Student može izabrati neki od ponuđenih primera ili može realizovati funkcionalnosti po sopstvenom izboru uz obavezne konsultacije sa asistentom.
Primeri:

- fraze - fraza se navodi pod navodnicima. U rezultatima se prikazuju (uz rangiranje) stranice u kojima se navedeni delovi fraze pojavljuju uzastopno u istom redosledu.
- did you mean - Ukoliko rezultata pretrage nema (nema rangiranih stranica) ili ih ima veoma malo, ponuditi korisniku da zadati upit zameniti sličnim, popularnijim upitom.
- kombinovana pretraga logičkih operatora- obavezna upotreba zagrada i dva logička operatora (dictionary OR list OR set) AND tree

Nefunkcionalni zahtevi

- Mašina za pretraživanje tekstualnih dokumenata treba da bude konzolna aplikacija napisana u programskom jeziku Python 3.
- Posmatraju se samo HTML stranice (datoteke/dokumenti) unutar test skupa.
- Program ni u jednom trenutku ne sme neočekivano da prestane da radi (na primer, za korenski direktorijum se odabere direktorijum koji sadrži slike). Voditi računa da se korisnik obavesti ukoliko unese upit u formatu koji ovom specifikacijom nije definisan kao validan.
- U posebnom tekstualnom fajlu, studenti treba da navedu koje strukture podataka su koristili kao i da objasne svaki od algoritama koji je implementiran. Objašnjenje algoritama treba da bude u vidu kraktog tekstualnog zapisa, prikaza pseudokoda, formule. U istom fajlu navesti i da li je rađena neka od funkcionalnosti za dodatne bodove. (okvirno jedna strana A4)

Opšte informacije

- Zadatak nosi 35 bodova.
- Rok za predaju(slanje): **21.6.2022 u 23:59**
- **Odbrana projekta**: 24.6.2022 učionica NTP-307 u posebnim terminima koji će biti istaknuti do 21.6.

- Prilikom slanja rešenja smestiti sve fajlove u folder pod nazivom ***projekat2_sv_XX_YYYY*** gde se umesto XX_YYYY navodi broj indeksa - broj upisa i godina upisa (primer: projekat2_sv_08_2016). Ubaciti fajl u zip arhivu i nazvati je isto kao i zadatak (*projekat2_sv_XX_YYYY.zip*). Poslati na email asistenta (*tamara.kovacevic@uns.ac.rs*) uz naslov poruke "***projekat2_sv_XX_YYYY***". Nakon prijema mail-a ćete dobiti potvrdu asistenta o pristignutom zadatku.

- Ukoliko studenti imaju svoj laptop, poželjno je da isti ponesu na termin odbrane. Studenti koji nisu u mogućnosti da brane na svojim računarima potrebno je prilikom slanja zadatka napomenuti asistentu i biće im obezbedjen računar za odbranu.

- Pre slanja upita vezanih za projektni zadatak pogledati fajl sa čestim pitanjima

Moguće su male korekcije u tekstu zadatka. Prilikom pristupa poverite datum poslednje izmene teksta(navedena u nazivu dokumenta).