

Računarske mreže, Ispit - JUN2 2022

Pročitati sve zadatke **pažljivo** pre rada - sve što nije navedeno ne mora da se implementira!

Na Desktop-u se nalazi zip arhiva. Unutar arhive se nalazi direktorijum u formatu `rm_rok_Ime.Prezime_mXGGXXX` u kome se nalazi validan IntelliJ projekat. Izvući direktorijum iz arhive na Desktop i ubaciti svoje podatke u ime. Otvoriti IntelliJ IDEA, izabrati opciju `Open project` (ne `Import project`!) i otvoriti pomenuti direktorijum. Sve kodove ostaviti unutar već kreiranih Java fajlova. **Kodovi koji se ne prevode se neće pregledati.**
Nepoštovanje formata ulaza/izlaza nosi kaznu od -10% poena na zadatku!

1. Tokovi podataka i niti (15p)

Napisati program koji obrađuje URL-ove unesene u pregledač i na standardni izlaz ispisuje odgovarajuće poruke.

- U direktorijumu `tests` na Desktop-u se nalazi datoteka `log.txt`, koja predstavlja log unesenih URL-ova u nekom pregledaču, od kojih su neki validni, a neki ne. Svaku liniju iz log datoteke treba obraditi u zasebnoj niti na način naveden u nastavku. (2p)
- Proveriti da li je URL validan. Ako nije, ispisati na standardni izlaz poruku u formatu: (2p)

`BADURL <LOS_URL_IZ_LOG_FAJLA>`

- U slučaju da je protokol `FILE`, proveriti da li ta datoteka postoji na našem sistemu. Ako postoji, otvoriti je putem URL klase i prebrojati linije u njoj, a poruku ispisati u formatu: (3p)

`OK <PUTANJA_DO_FAJLA> <BROJ_LINIJA>`

- U slučaju da datoteka ne postoji, ispisati poruku u formatu: (2p)

`NOTFOUND <PUTANJA_DO_FAJLA>`

- U slučaju bilo kog drugog protokola, "proslediti" zahtev dalje ispisom poruke u sledećem formatu: (3p)

`FORWARD <POCETNI_URL> [<TIMESTAMP>]`

- Voditi računa o tome da su resursi pravilno zatvoreni. (1p)
- Sinhronizovati ispis na standardni izlaz. (2p)

primer:

ulaz:

```
http://www.matf.bg.ac.rs/dir1/dir2/file1.txt
sftp://2001:0db8:85a3::8a2e:0370:7334/dir1/text.txt
https://test.net/path/image.jpg
FILE:///home/ispit/Desktop/tests/dir1/test1.txt
file:///home/ispit/books/rm.pdf
```

izlaz:

```
FORWARD http://www.matf.bg.ac.rs/dir1/dir2/file1.txt [Sat Jun 04 18:33:17 CEST 2022]
BADURL sftp://2001:0db8:85a3::8a2e:0370:7334/dir1/text.txt
FORWARD https://test.net/path/image.jpg [Sat Jun 04 18:33:17 CEST 2022]
OK /home/ispit/Desktop/tests/dir1/test1.txt 7
NOTFOUND /home/ispit/books/rm.pdf
```

*Napomena: Ohrabrujemo studente da koriste **netcat** kako bi testirali delimične implementacije i otkrili greške pre vremena. Takođe, ukoliko se npr. preskoči implementacija servera, može se mock-ovati server putem **netcat-a**.*

Okrenite stranu!

2. Neblokirajući izbori (25p)

U toku su izbori za studentski parlament. U fajlu `lista.txt` nalaze se kandidati za koje je moguće glasati. Napraviti klient-server Java aplikaciju koja vodi statistiku o izbornim glasovima.

- Napisati Java klasu koja ima ulogu **blokirajućeg** TCP klijenta koristeći **Java Channels API**. Formirati konekciju sa lokalnim serverom na portu 12345. Nakon formiranja konekcije uneti glas sa standardnog ulaza i poslati ga serveru. Odgovor od servera ispisati na standardni izlaz. (8p)
- Napisati Java klasu koja ima ulogu lokalnog **neblokirajućeg** TCP servera koji osluškuje na portu 12345 koristeći **Java Channels API**. Server vodi evidenciju o trenutnom broju glasova. Na svakih 5 sekundi, ako je do tada bar neko glasao, ispisati trenutni procenat glasova za svakog kandidata i trenutni procenat nevalidnih glasova. Validni su glasovi samo za kandidate iz fajla `lista.txt`. Za svaki primljeni glas od klijenta poslati odgovor. Ako je klijent poslao nevažeći glas, kao odgovor poslati obaveštenje da glas nije validan. U suprotnom poslati zahvalnicu na glasanju i trenutni procenat glasova za izabranog kandidata zaokružen na dve decimale. (15p)
- Postarati se da su svi resursi ispravno zatvoreni u slučaju izuzetka (2p)

Primer rada:

```
ulaz : Marko
izlaz: Hvala sto ste glasali , trenutni procenat glasova za vases kandidata je 1.00

ulaz : Pera
izlaz: Vas glas nije validan

ulaz : maja
izlaz: Hvala sto ste glasali , trenutni procenat glasova za vases kandidata je 0.33
```

3. Protocol handlers (15p)

Implementirati podršku za URL-ove koji koriste `evote` protokol. Opis protokola je dat u prethodnom zadatku.

- Prilikom otvaranja konekcije, formirati vezu koristeći **Socket API**. Povezati se na server i port na osnovu URL-a i otvoriti ulazni tok do odgovora od strane servera. (5p)
- Omogućiti slanje upita pomoću parametra `kandidat` URL-a, primer (paziti na enkodiranje karaktera unutar URL-a):

```
evote://localhost:1337?kandidat=Marko+Markovic
```

Server šalje nazad rezultat koji klijent ispisuje kao u primeru ispod. (5p)

- Ukoliko port nije naveden unutar URL-a, iskoristiti predefinisani podrazumevani port isti kao u prethodnom zadatku. (1p)
- Predefinisati `getInputStream()` metod da vraća ulazni tok do odgovora od strane servera ukoliko je konekcija ostvorena, a `null` ako nije. (1p)
- Postarati se da je moguće bezbedno koristiti implementirani handler u višenitnom okruženju. (1p)
- Napisati jednostavan test - kreirati URL, otvoriti konekciju do resursa i ispisati sve podatke koje server pošalje. (2p)

Primer rada:

```
URL: evote://localhost
izlaz: Upit nije kompletan.

URL: evote://localhost:12345?test=5
izlaz: Upit nije kompletan.

URL: evote://localhost:7337?kandidat=Marko
izlaz: Hvala sto ste glasali , trenutni procenat glasova za vases kandidata je 1.00

URL: evote//localhost?kandidat=Pera
izlaz: Vas glas nije validan
```