

Računarske mreže, Ispit - PREDROK 2024

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 IntelliJ projekat u formatu **rm.rok.Ime.Prezime.mXGGXXX**.
Dekompresovati arhivu na Desktop i ubaciti svoje podatke u ime pomenutog direktorijuma.
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 (20p)

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:
BADURL "LOS_URL_IZ_LOG_FAJLA" (2p)
- 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:
OK PUTANJA_DO_FAJLA BROJ_LINIJA (3p)
- U slučaju da datoteka ne postoji, ispisati poruku u formatu:
NOTFOUND PUTANJA_DO_FAJLA (2p)
- U slučaju bilo kog drugog protokola, "proslediti" zahtev dalje ispisom poruke u sledećem formatu:
FORWARD POCETNI_URL [TIMESTAMP] (3p)
(2p)
- Voditi računa o tome da su resursi pravilno zatvoreni. (2p)
- Sinhronizovati ispis na standardni izlaz. (2p)

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. TCP Sockets (20p)

Napraviti osnovu za TCP klijent-server Java aplikaciju koja pruža informacije o odlaznim letovima sa aerodroma.

- U direktorijumu aerodroma, unutar direktorijuma tests na Desktop-u, nalaze se tekstualni fajlovi koji sadrže informacije o odlaznim letovima sa nekog aerodroma za taj dan. Ime fajla predstavlja ime grada u kome se aerodrom nalazi, a svaka linija fajla sadrži informacije o jednom letu i oblika je:
[JEDINSTVENA ŠIFRA LETA] [GRAD SLETANJA] [VREME POLETANJA] [VREME SLETANJA].
Na serverskoj strani keširati ove podatke kako se ne bi čitali iznova za svakog klijenta. (4p)
- Napraviti Java klasu koja ima ulogu lokalnog TCP servera koji osluškuje na portu 12345. Svakom novom klijentu server šalje spisak svih gradova za čije aerodrome ima informacije o letovima. (4p)
- Napraviti Java klasu koja ima ulogu lokalnog TCP klijenta. Nakon uspostavljanja konekcije sa serverom na portu 12345, klijent čeka spisak gradova od servera i ispisuje ih na standardni izlaz. Nakon toga, klijent dodatno šalje serveru ime grada, uneto sa standardnog ulaza, sa čijeg aerodroma želi da dobije informacije o odlaznim letovima, a zatim ponovo čeka odgovor od servera koji ispisuje na standardni izlaz i završava sa radom. (7p)
- U zavisnosti od imena grada, server šalje keširane informacije iz odgovarajućeg fajla. (4p)
- Postarati se da su svi resursi ispravno zatvoreni u slučaju izuzetka. (1p)

3. UDP Sockets (20p)

Napisati Java aplikaciju koja dohvata informacije o studentu na osnovu njegovog indeksa.

- Napisati Java klasu koja ima ulogu UDP klijenta. Poslati inicijalni datagram koji sadrži indeks u formatu XXX/GGGG i jednu od komandi prosek ili ime lokalnom serveru koji osluškuje na portu 12321. Primiti datagram od servera koji sadrži informaciju koja je zatražena. Podatke u datagramu kodirati proizvoljno. (4p)
- Napisati Java klasu koja ima ulogu servera koji osluškuje na portu 12321. Iz tekstualnog fajla studenti.txt učitati informacije o studentima i keširati ih na serveru. Moguće je koristiti proizvoljnu kolekciju podataka. (2p)
- Primiti datagram od klijenta, isparsirati komandu, i vratiti datagram sa zatraženom informacijom o studentu. (4p)
- U slučaju nepostojećeg indeksa, ili nevalidne komande, vratiti klijentu poruku "Zahtev nije validan". (2p)
- Postarati se da su svi resursi pravilno zatvoreni u slučaju izuzetka. (3p)

KLIJENT : 002/2018 ime	SERVER : Pera Peric
KLIJENT : 100/2018 prosek	SERVER : 6.3
KLIJENT : 101/2018 prosek	SERVER : Zahtev nije validan!
KLIJENT : 100/2018 email	SERVER : Zahtev nije validan!