

Računarske mreže (modul Računarstvo i informatika) - praktični ispit -

Tehničke napomene

Na radnoj površini nalazi se šifrovana zip arhiva sa nazivom `mreze.r.jul.zip`. Unutar te arhive nalazi se direktorijum sa nazivom `rmr_jul_ImePrezime_mrGGXXX` (gde `mrGGXXX` predstavlja korisničko ime Vaše Alas mejl adrese). U ovom direktorijumu nalazi se validan IntelliJ projekat (koji predstavlja Vaš rad) sa paketima `filter`, `hangman` i `caesar_cipher`. Ovaj direktorijum izvući iz arhive na Desktop i preimenovati ga u skladu sa Vašim podacima. Otvoriti IntelliJ IDEA, izabrati opciju `Open project` (ne `Import project`!) i otvoriti pomenuti direktorijum. **Kodovi koji se ne prevode se neće pregledati.** Vreme za izradu ispita je **3 sata**.

Srećan rad!

Ispitni zadatak 1: Filter (15 poena)

Implementirati Java aplikaciju koja filtrira URL adrese na osnovu skupa veb adresa datih u datoteci `filter.txt` (internet lokacije na koje ukazuju veb adrese u ovoj datoteci treba da budu nedostupne). Nakon pokretanja aplikacije, korisnik unosi URL adrese sa standardnog ulaza (do kraja ulaza). Za svaku unetu URL adresu na standardni izlaz potrebno je ispisati `Dozvoljeno`, `Nedozvoljeno` ili `Protokol nije podrzan`.

Važno: Provera se *ne obavlja poređenjem stringova adresa*, već se za svaku URL adresu (i one iz `filter.txt` i one koje korisnik unese) mora odrediti IP adresa na koju ona ukazuje. Zabrana pristupa se vrši poređenjem ovih IP adresa.

```
filter.txt:
chatgpt.com
facebook.com
math.rs
poincare.matf.bg.ac.rs
```

Interakcija sa programom:

```
Unesite URL: https://google.com/
Dozvoljeno
Unesite URL: https://chatgpt.com/
Nije dozvoljeno
Unesite URL: sftp://www.matf.bg.ac.rs/
Protokol nije podrzan
Unesite URL: https://www.matf.bg.ac.rs/
Nije dozvoljeno
Unesite URL: https://poincare.matf.bg.ac.rs/~milan.mitreski
Nije dozvoljeno
```

Naredni zadatak se nalazi na sledećoj strani!

Ispitni zadatak 2: TCP igra — Vešala (eng. Hangman)

Implementirati TCP client-server aplikaciju u kojoj klijent pokušava da pogodi reč koju je server zamislio (iz datoteke `words.txt`) po uzoru na igru *vešala*.

U početku, klijent se povezuje na server, koji pokreće posebnu nit koja obradjuje tog klijenta, nakon čega server nastavlja da prihvata zahteve od drugih klijenata. Nakon što se klijent konektuje na server, prosledjuje serveru svoje ime i broj koji predstavlja broj "života" u igri vešala. Nakon toga, server zamišlja reč (za svakog klijenta posebnu) i prosledjuje klijentu poruku da je zamislio reč kao i postavku za samu igru u sledećem obliku (pretpostavimo da je broj života 6 i da je zamišljena reč **BANANA** – pretpostaviti da će sve reči biti napisane velikim slovima abecede)

Rec: -----; Broj zivota: 6; Isprobana slova:

Nakon toga, klijent odgovara jednim slovom, koje predstavlja potez u igri vešala. Ako se to slovo nalazi u reči, server otkriva sva pojavljivanja tog slova u reči, a u suprotnom smanjuje broj života klijentu. U oba slučaja, server dodaje slovo u listu isprobanih slova. Ukoliko klijent unese slovo koje je već probao ili ne unese veliko slovo abecede – server prijavljuje grešku. Na kraju, ukoliko je klijent pogodio reč, server čestita klijentu, a ukoliko je klijent ostao bez života, server će mu poželeti više sreće drugi put.

Primer:

```
> Zarko 4
Rec: -----; Broj zivota: 4; Isprobana slova:
> B
Rec: B-----; Broj zivota: 4; Isprobana slova: B
> A
Rec: BA-A-A; Broj zivota: 4; Isprobana slova: B A
> S
Rec: BA-A-A; Broj zivota: 3; Isprobana slova: B A S
> L
Rec: BA-A-A; Broj zivota: 2; Isprobana slova: B A S L
> K
Rec: BA-A-A; Broj zivota: 1; Isprobana slova: B A S L K
> N
Rec: BANANA; Cestitam Zarko!
```

Server pokrenuti na portu 5555. Server prestaje sa radom prosledjivanjem SIGINT ([CTRL+C]) signala (odnosno, nije potrebno posebno implementirati zaustavljanje serverskog dela aplikacije). **Obavezno je pravilno zatvoriti i osloboditi sve korišćene resurse.**

Naredni zadatak se nalazi na sledećoj strani!

Ispitni zadatak 3: Asocijacije (25 poena)

Implementirati UDP client-server aplikaciju u kojoj klijent pokušava da reši asocijaciju. Server učitava asocijaciju iz datoteke `asocijacija.txt` koja sadrži 5 redova, pri čemu prva četiri reda sadrže po 5 reči (svaki red odgovara jednoj od kolona A,B,C,D, a reči u redu odgovaraju poljima 1,2,3,4 te kolone, dok je peta reč rešenje kolone), a poslednji red sadrži jednu reč koja predstavlja konačno rešenje. Pretpostaviti da sva polja i rešenja se sastoje od jedne reči (bez razmaka)

```
asocijacije.txt:
festival foto-aparat tableta mjuzikl film
odelo navika berza honorar rad
hemija lanac alergija odgovor reakcija
stadion uefa fifa pesak fudbal
akcija
```

Nakon pokretanja servera, on učitava asocijaciju i nakon toga odgovara na zahteve klijenata koji mogu biti:

- **OPEN** <polje> – vraća klijentu vrednost polja – vrednost <polje> može biti A1,A2,...,D4
- **SOLVE** <kolona> <odgovor> – vraća klijentu Tacno ili Netacno u zavisnosti od toga da li je <odgovor> rešenje kolone <kolona> – vrednost <kolona> može biti A,B,C,D
- **FINAL** <odgovor> – vraća klijentu Bravo ili Netacno u zavisnosti od toga da li je <odgovor> konačno rešenje

Svaki klijent posebno vodi evidenciju o otvorenim poljima i rešenim kolonama. Klijentski deo aplikacije je dužan da spreči duplo otvaranje polja, odnosno duplo rešavanje kolona. Moguće je da klijent dva puta pokuša isto (netačno) rešenje za jednu kolonu. Kada klijent reši asocijaciju ili unese **IGIVEUP**, oslobadjaju se svi resursi i klijent završava sa radom. U slučaju pogrešnog unosa, obavezno je obraditi grešku.

Primer rada aplikacije:

```
> OPEN A1
festival
> SOLVE A muzika
Netacno
> SOLVE A film
Tacno
> OPEN D3
fifa
> SOLVE D fudbal
Tacno
> SOLVE D uefa
Kolona D je vec resena. Nista nije poslato serveru!
> FINAL montevideo
Netacno
> FINAL akcija
Bravo
```

Server pokrenuti na portu 5555. Server prestaje sa radom prosledjivanjem SIGINT ([CTRL+C]) signala (odnosno, nije potrebno posebno implementirati zaustavljanje serverskog dela aplikacije). **Obavezno je pravilno zatvoriti i osloboditi sve korišćene resurse.**