

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

### Tehničke napomene

Na radnoj površini nalazi se šifrovana zip arhiva sa nazivom `mreze.r.sep.zip`. Unutar te arhive nalazi se direktorijum sa nazivom `rmr_sep_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 `ports`, `quiz` i `wordle`. 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: Analiza mrežnih portova po protokolima (15 poena)

Potrebno je implementirati Java aplikaciju koja proverava dostupnost mrežnih servisa na osnovu zadatih IP adresa i portova. Aplikacija najpre učitava listu IP adresa iz fajla `ips.txt`, gde se svaka adresa nalazi u posebnom redu.

Zatim aplikacija pokreće pet niti, pri čemu je svaka nit zadužena za proveru dostupnosti jednog mrežnog protokola. Konkretno, jedna nit proverava HTTP servis na portu 80, druga HTTPS na portu 443, treća FTP na portu 21, četvrta SSH na portu 22, a peta SMTP na portu 25. Svaka nit prolazi kroz sve IP adrese iz ulaznog fajla i pomoću klase `Socket` (TCP) pokušava da uspostavi konekciju na odgovarajući port. Ako je konekcija uspešno ostvarena, smatra se da je taj servis aktivan.

Po završetku provere svih niti, aplikacija treba da upiše broj aktivnih servisa za svaki protokol u izlazni fajl `protocol_stats.txt`. Na taj način, korisnik dobija pregled dostupnosti standardnih mrežnih servisa za sve zadate IP adrese, pri čemu se provera obavlja paralelno putem više niti.

`ips.txt`:

```
192.168.0.10
192.168.0.15
192.168.0.123
```

`protocol_stats.txt`:

```
HTTP (80): 3 active
HTTPS (443): 1 active
FTP (21): 0 active
SSH (22): 2 active
SMTP (25): 1 active
```

## Ispitni zadatak 2: UDP kviz — Pitanja i odgovori (25 poena)

Implementirati UDP client-server aplikaciju koja simulira jednostavan kviz.

- Server učitava listu pitanja i tačnih odgovora iz fajla `questions.txt`, gde svaka linija ima format: `Pitanje|Odgovor`.
- Klijent se povezuje na server, koji mu šalje jedno po jedno pitanje.
- Klijent odgovara, a server proverava da li je odgovor tačan. Ako jeste, šalje **Tačno!**, u suprotnom **Netačno!**.

Primer fajla `questions.txt`:

```
Glavni grad Francuske?|Pariz
Koliko je 5+3?|8
Koja boja je nebo?|Plava
Najveći okean?|Pacifik
Ko je autor "Na Drini ćuprija"?|Ivo Andrić
```

Server pokrenuti na portu 9000. Svaki zahtev i odgovor predstavljaju jedan UDP paket. Server može istovremeno obradivati više kviz sesija (nije obavezna implementacija paralelizma). Server prestaje sa radom prosledjivanjem SIGINT ([CTRL+C]). **Obavezno je pravilno zatvoriti i osloboditi sve korišćene resurse.**

**Naredni zadatak se nalazi na sledećoj strani!**

### Ispitni zadatak 3: Wordle (25 poena)

Implementirati igricu Wordle kao TCP Java client-server aplikaciju. Wordle je igrice gde klijent pokušava da pogodi petoslovnu reč koju je server zamislio (spisak reči koje server može da zamisli i koje klijenti mogu da pogodjaju se nalazi u datoteci `wordle.txt`) u ne više od šest pokušaja. Nakon svakog pokušaja, za svako slovo reči koju je klijent poslao određuje da li se nalazi u zamišljenoj reči (ako ne, to obeležava sa -) i da li se nalazi na pravom mestu (+) ili ne (\*). Jednostavnosti radi, može se pretpostaviti da zamišljene reči, kao i reči koje klijent šalje kao svoje pokušaje, ne sadrže dva ista slova. Ako klijent pogodi reč, server odgovara sa **Pogodak. Cestitam!**, a u slučaju da klijent ne pogodi reč u šest pokušaja, server odgovara sa **Nisi pogodio. Više sreće drugi put!**. U slučaju pogrešnog ili nevalidnog unosa, obraditi grešku.

Serverski deo implementirati kao višenitnu aplikaciju, koji prihvata konekcije klijenata i za svakog klijenta pokreće nit koja ga obradjuje, nakon čega server dalje prihvata konekcije klijenata. Server zamišlja istu reč za sve klijente.

Primer rada:

```
> LEAKY
-*---
> CREAM
-+*-*
> PINKY
+*---
> PRIME
Pogodak. Cestitam!
```

Server pokrenuti na portu 5555. Server prestaje sa radom prosledjivanjem SIGINT ([CTRL+C]). **Obavezno je pravilno zatvoriti i osloboditi sve korišćene resurse.**