

# Računarske mreže 4I, Kolokvijum - B

24.11.2019.

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 sa imenom **rm\_kolB\_Ime\_Prezime\_miGGXXX** u kome se nalazi validan IntelliJ projekat. Izvući direktorijum iz arhive na Desktop i preimenovati ga tako da ime odgovara podacima studenta. Otvoriti IntelliJ IDEA, izabrati opciju **Open 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!** Vreme za rad: **2h**.

## 1. Tokovi podataka i niti (15p)

Napisati program koji rekurzivno obilazi direktorijum i ispisuje broj linija u svim **.c** fajlovima koristeći niti.

- Na Desktop-u se nalazi direktorijum **tests**. Obići pomenuti direktorijum i ispisati na standardni izlaz ukupan broj svih regularnih fajlova unutar tog direktorijuma. (2p)
- Za svaki pronadjeni fajl sa ekstenzijom **.c** kreirati novi URL objekat koristeći URL klasu i **FILE** protokol. Ispisati kreirane URL-ove na standardni izlaz (videti primere ispod).

Putanja	Odgovarajući URL	(2p)
tests/dir1/1.txt	FILE:///home/ispit/Desktop/tests/dir1/1.txt	

- Za svaki kreirani URL, kreirati zasebnu nit koja će otvoriti **baferisani** ulazni tok do resursa putem URL klase i pročitati sadržaj fajla (detalji obrade su u narednoj stavci). Kodnu stranu prilikom učitavanja postaviti na ASCII. (4p)
- Na standardni izlaz ispisati ukupan broj linija u svim fajlovima iz prethodne stavke tako što će svaka nit prebrojati linije za fajl koji joj je dodeljen (videti primer ispisa ispod teksta zadatka). Pritom, paziti na sinhronizaciju niti ukoliko se koristi deljeni brojač. (5p)
- Postarati se da program u slučajevima ispravno zatvori sve korišćene resurse. (1p)

```
ulaz:
izlaz: files:      20
       url:       FILE:///home/ispit/Desktop/tests/dir/dir1/dir11/palindrome.c
       url:       FILE:///home/ispit/Desktop/tests/dir/dir1/smile.c
       url:       FILE:///home/ispit/Desktop/tests/dir/dir2/dir21/pi.c
       result:    85
```

---

Okrenite stranu!

## 2. Sockets (15p)

- Napraviti Java aplikaciju koja ima ulogu servera. Pokrenuti lokalni server na portu 31415, koristeći `ServerSocket` klasu. Server za svakog povezanog klijenta pokreće zasebnu nit. (4p)
- Napraviti Java aplikaciju koja ima ulogu klijenta. Povezati se na lokalni server na portu 31415 koristeći `Socket` klasu. Nakon formiranja konekcije klijent šalje serveru relativnu putanju do fajla na fajlsistemu servera (unetu sa standardnog ulaza) u odnosu na direktorijum `tests`. Ispisati primljenu putanju na standardni izlaz na serverskoj strani, a klijentu poslati indikator da li je putanja validna ili ne (npr. putanja je nevalidna ukoliko fajl na toj putanji ne postoji). (3p)
- Zadatak servera je da izračuna zbir svih realnih brojeva koji se nalaze u datom fajlu. Server šalje klijentu zbir, a klijent rešenje ispisuje na standardni izlaz. U slučaju da u fajlu nema realnih brojeva, klijent treba da ispiše poruku *Fajl ne sadrzi realne brojeve* na standardni izlaz. (7p)
- Postarati se da su svi resursi ispravno zatvoreni u slučaju izuzetka. (1p)

```
ulaz:          dir2/1.test
izlaz server:  dir2/1.test
izlaz klijent: Validna putanja
               Fajl ne sadrzi realne brojeve
```

```
ulaz:          dir2/2.test
izlaz server:  dir2/2.test
izlaz klijent: Validna putanja
               69.41
```

```
ulaz:          dir2/3.test
izlaz server:  dir2/3.test
izlaz klijent: Validna putanja
               34.27
```