

Elektrotehnički fakultet
Katedra za računarsku tehniku i informatiku
Praktikum iz Operativnih Sistema

Septembarski ispitni rok 2009. godine

1. Objasniti rezultat rada svake od sledećih komandi. [svaki ispravan odgovor vredi po 1 poen]

```
addgroup drugari
pwd
shutdown
sudo cat /proc/5780/cmdline
du -h --max-depth=1
man tr
chmod +x *.sh
sort spisak.txt | uniq > spisakN.txt
ssh pm07001d@rtidev4.etf.rs
whereis fdisk
```

2. Ukratko objasniti namenu sistema za upravljanje različitim verzijama programskog koda. Navesti primer upotrebe najosnovnijih komandi (za SVN ili CVS, po želji). [5 poena]

3. Definisati koncept virtuelizacije. Navesti makar tri primera šta može biti predmet virtuelizacije. [5 poena]

4. Detaljno objasniti priloženi ispis dobijen pozivima komande `ls -la`. [8 poena]

```
prw-r----- 1 syslog adm          0 2009-08-23 07:00 xconsole
brw-rw----  1 root  floppy      2,  0 2009-03-22 14:19 fd0
drwx-----  2 pera  rti           4096 2009-06-02 16:58 .ssh
-rw-rw-rw-  1 pera  rti       21687096 2008 03-19 11:32 doxygen-src.tar.gz
```

5. Detaljno objasniti priloženi *makefile* i nacrtati graf zavisnosti prevođenja fajlova. [7 poena]

```
projectX: data.o main.o io.o
    g++ data.o main.o io.o -o projectX
data.o: data.h data.cpp
    g++ -c data.cpp
main.o: data.h io.h main.h main.cpp
    g++ -c main.cpp
io.o: io.h io.cpp
    g++ -c io.cpp
clean:
    rm -rf *~ *.o projectX
```

6. Napisati na programskom jeziku Java programski kod servera koji istovremeno opslužuje više klijenata, svakog na prvom sledećem slobodnom portu (npr. počinje sa slušanjem na portu 4000, odmah nakon prijema zahteva od prvog klijenta, nastavlja da sluša na portu 4001 dok u zasebnoj niti opslužuje primljeni zahtev i tako redom zaključno sa portom 5000). Primljene poruke klijenata treba neizmenjene vratiti nazad klijentima. Na serverskoj strani treba ispisati na standardnom izlazu dužinu za svaku obrađenu poruku, po formatu `port:dužina`. [13 poena]

7. Konfiguracioni tekst fajl sadrži podatke, od kojih svaki zauzima tačno jednu liniju. Linija iste sadržine može biti ponovljena na više različitih mesta u fajlu. Napisati *bash shell script* pod imenom `singleandordered`, koji će urediti ulazni fajl prema leksikografskom poretku, tako da nema ponovljene redove. Prvi parametar će biti ime ulaznog fajla. Drugi parametar će biti ime izlaznog fajla. Ako korisnik ne obezbedi odgovarajući broj parametara, ispisati korisniku poruku o načinu pravilne upotrebe i prekinuti izvršavanje. Nije potrebno vršiti bilo kakvu proveru uspešnosti korišćenih komandi. [7 poena]

8. Koristeći *POSIX threads*, napisati na jeziku C ili C++ program za traženje najmanjeg elementa u nizu celih brojeva. Glavni program prvo čita sa standardnog ulaza broj elemenata niza i elemente niza, nakon čega stvara `NUM_THREADS` niti. Svaka nit će obavljati isti posao: pronaći će najmanji element u svom delu niza i ažurirati opšti minimum. Glavni program će sačekati da stvorene niti završe izvršavanje, pa će ispisati vrednost i indeks najmanjeg elementa. Obezbediti da pristup podacima o najmanjem elementu niza bude *thread-safe*. Po želji, pretpostaviti da će korisnik programa uvek unositi broj elemenata niza koji je celobrojni umnožak broja niti. [15 poena]

Napomena: Ispit traje 180 minuta. Ispravnim odgovorom u prvom zadatku se, za svaku od komandi, smatra jedna prosto proširena rečenica koja kratko i jasno objašnjava datu komandu i njen efekat. U četvrtom i petom zadatku ne treba pisati ništa van onoga što se zadatkom traži. Ukoliko u bilo kom zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je uokviri (da bi bila lakše prepoznata prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke.