

# Prva laboratorijska vježba

## Osnove korištenja operacijskog sustava Linux

October 2017

Za svaki zadatak potrebno je napisati po jednu bash skriptu.

### Zadatak 1.

- Napišite bash skriptu koja čita tekst sa standardnog ulaza i ispisuje ga na standardni izlaz. Nad pročitanim podacima ne treba raditi izmjene.
- Koristeći mogućnosti upravljanja datotekama u Linuxu ostvarite tekstualnu komunikaciju između dva korisnika na istom računalu. Pretpostavite da svaki korisnik može imati dva terminala stalno otvorena. Poruke poslane od jednog korisnika moraju odmah biti prikazane drugom korisniku. U implementaciji smijete koristiti najviše dvije datoteke. Rješenje mora biti jedna ili dvije bash skripte koje pripremaju i pokreću komunikaciju. U komentarima skriptata objasnite postupak slanja i primanja poruka.

### Zadatak 2.

Zanima nas koliko će se puta u nekoj godini dogoditi petak 13. Srećom, nalazimo se u Linux okruženju gdje nam na raspolaganju stoji naredba `ncal`, te poznajemo naredbe za pretraživanje teksta koje će nam olakšati ovaj zadatak.

- Prije svega, proučite man stranice naredbe `ncal`. Mi ćemo ju koristiti u vrlo jednostavnom obliku: `ncal <godina>`.
- Koristeći mogućnosti pretraživanja teksta prije svega iz ispisa izbacite sve linije koje ne počinju nizom znakova `Fr`.<sup>1</sup>
- Nadalje, prebrojite koliko od tih linija uopće sadrži broj 13 u sebi. Primijetite da u ovom trenutku skripta uistinu broji koliko se puta u godini dogodio petak 13.
- Sljedeće što želimo dodati je da skripta od korisnika prima godinu za koju provjerava koliko puta će se dogoditi petak 13. i zatim ispisuje poruku na `stdout`. U predlošku je primjer koji možete koristiti, a on koristi naredbu `read` za čitanje inputa korisnika. Dok korisnik ne unese godinu, skripta se neće izvršavati.

Listing 1: Čitanje inputa

```
#!/bin/bash
echo -n "Upisi godinu za koju te zanima koliko puta se dogodio petak 13.: "
read godina;

petkovi=$(ncal $godina | <ostatak super kul naredbe>)
echo "U godini $godina, petak 13. se dogodio $petkovi puta."
```

---

<sup>1</sup>Ili, jasno, `Pe` od "Petak" ili nešto treće, ovisno o lokalizacijskim postavkama.

- e) Sada kad imamo i lijep ispis zadatka, možda bi korisnika zanimalo koliko puta će se petak 13. dogoditi u narednih x godina. Koristeći istu filozofiju, nakon prvog ispisa zatražite sljedeći upis "do koje godine te zanima koliko puta će se dogoditi petak 13.: " gdje će korisnik unijeti, primjerice, godinu "2100", a program mu ispisati:

Listing 2: Ispis

```
okosl@poseidon:~$ bash labos.sh
Upisi godinu za koju te zanima koliko puta se dogodio petak 13.: 2017
U godini 2017, petak 13. se dogodio 2 puta.

Upisi godinu za do koje te zanima koliko puta ce se dogoditi petak 13.: 2100
U godini 2017, petak 13. se dogodio 2 puta.
U godini 2018, petak 13. se dogodio 2 puta.
U godini 2019, petak 13. se dogodio 2 puta.
U godini 2020, petak 13. se dogodio 2 puta.
....
U godini 2100, petak 13. se dogodio 1 puta.
```

Potreban nam je još jedan blok koji će sada iterirati kroz zadani opseg godina

Listing 3: Iteracija po opsegu

```
#!/bin/bash
...
echo "U godini $godina, petak 13. se dogodio $petkovi puta."

echo -n "Upisi godinu za do koje te zanima koliko puta ce se dogoditi petak 13.: "
read opseg

for godina in $(seq 2017 $opseg)
do
    petkovi=$(ncal $godina | <ostatak super kul naredbe>)
    echo "U godini $godina , petak 13. se dogodio $petkovi puta."
done
```

### Zadatak 3.

Mali je Ivica prije pola godine naišao na skladbu o Linuxu koja mu se jako svidjela. Iako je siguran da je skladbu preuzeo (na najlegalniji mogući način), ne sjeća se kamo ju je spremio. Stoga se odlučio javiti prijatelju, inače entuzijastičnom linuxašu, koji će mu drage volje napisati skriptu koja će riješiti sve njegove probleme, i tako proširiti *dobru riječ Linuxa*. Odlučio je da će skripti na ulaz poslati riječ za koju je siguran da se nalazi **negdje** u nazivu pjesme. Nakon izlistavanja svih .mp3 datoteka s tom riječju u imenu bit će potrebno samo upisati index pjesme, slušati ju i uživati. Taj ste, prijatelj, naravno, upravo Vi.

- Pomoću naredbe  
`wget -r --no-parent -nH -nc -R "index.html*" --cut-dirs=1 http://close.port0.org/lab1/-P /OKOSL/lab1` preuzmite datoteke, kako biste mogli jednostavnije testirati valjanost skripte.
- Proučite naredbe za puštanje mp3 zvučnih datoteka. Preporučamo instalaciju `vlc`, te korištenje `nvlc` naredbe.
- Kroz zadatak 2 ste se upoznali s načinima čitanja inputa. Isti princip iskoristite za čitanje inputa u ovom zadatku. Preporuča se komunikacija s korisnikom pomoću `echo` naredbe.
- Nakon zadovoljavajućeg izgleda inicijalne komunikacije pomoću `find` naredbe pronađite sve mp3 datoteke rekursivno iz vašeg matičnog direktorija. Izlaz preusmjerite u tekstualnu datoteku u `/tmp/lab1.dat`.

d) Čitajte tu datoteku kroz `while` petlju na način:

Listing 4: Iteracija kroz redove datoteke

```
#!/bin/bash
...
cat /tmp/lab1.dat | while read line
do
...
done
```

U svakom retku ispišite broj datoteke i njeno ime. Možete istražiti i kako promijeniti boje svakog ispisa, pa pokušati to primijeniti.

- e) Ponovno komunicirajte s korisnikom te ga tražite da ispiše valjan index mp3 datoteke. Skripta ne smije nastaviti izvršavanje dok se ne unese valjan index mp3 datoteke.
- f) Ispišite punu putanju mp3 datoteke pod tim indexom, te ju pokrenite pomoću `nvlc` (ili neke druge naredbe za pokretanje mp3 datoteke).