**1.**  
Napisati bash skriptu koja će u tekućem kazalu i svim podkazalima pronaći datoteke koje su izmijenjene u posljednjih 12 dana, te za svaku od tih datoteka ispisati ime i broj redaka koje datoteka sadrži. Ispis oblikovati prema primjeru u nastavku.  
  
Poziv skripte zadatak2.sh:   
$ ./zadatak2.sh  
./Podkazalo2/Skriptni\_1\_bash.log ..... 2191 redaka  
./Podkazalo2/Skriptni\_3\_Python.tex ..... 1183 redaka  
./Podkazalo2/Skriptni\_3\_Python.log ..... 1229 redaka  
./Podkazalo2/Skriptni\_2\_Perl.log ..... 2411 redaka  
./Skriptni\_1\_bash.log ..... 2191 redaka  
./Skriptni\_1\_bash.tex ..... 3302 redaka  
./Podkazalo1/Skriptni\_1\_bash.vrb ..... 33 redaka  
./Podkazalo1/Skriptni\_1\_bash.tex ..... 3302 redaka  
./Podkazalo1/Skriptni\_2\_Perl.tex ..... 4409 redaka  
./zadatak2.sh ..... 15 redaka  
./definicije.tex ..... 195 redaka  
./Skriptni\_2\_Perl.log ..... 2411 redaka

**Rješenje**:

#!/bin/bash

find -type f -mtime -12 | xargs wc -l | sed -r 's/ \*([0-9]\*) (.\*)/\2 ... \1 redaka/' | head -n -1

**2.**

Napisati bash skriptu koja će u tekućem kazalu i svim podkazalima pronaći datoteke koje su izmijenjene unutar zadanog perioda, te za svaku od tih datoteka ispisati ime i broj riječi koje datoteka sadrži. Ispisati i ukupni broj riječi u svim pronađenim datotekama. Period se izražava u danima i navodi kao argument pri pozivu skripte. Skripta treba provjeriti broj argumenata, te u slučaju neispravnog pozivanja ispisati uputu o načinu korištenja i izaći iz skripte. (Napomena: predvidjeti pozivanje skripte kao izvršne datoteke.

Poziv skripte zadatak4.sh:  
$ ./zadatak4.sh 7  
./Podkazalo2/Skriptni\_3\_Python.tex ........ 2982 rijeci  
./Podkazalo2/Skriptni\_3\_Python.log ........ 4142 rijeci  
./Podkazalo2/Skriptni\_2\_Perl.log ........ 6860 rijeci  
./Podkazalo1/Skriptni\_2\_Perl.tex ........ 15169 rijeci  
./zadatak1.sh ........ 70 rijeci  
./zadatak4.sh ........ 143 rijeci  
./Skriptni\_2\_Perl.log ........ 6860 rijeci  
./izlaz1.txt ........ 189 rijeci  
------------------------------  
Ukupno: 36415 rijeci

**Rješenje:**

#!/bin/bash

if [ $# -ne 1 ]

then

echo "krivi broj argumenata"

exit

fi

period=$1

ukupno=0

broj=0

find -mtime -$period -type f | xargs wc -w | sed -r 's/ \*([0-9]\*) (.\*)/\2...\1 riječi/' | head -n -1 #probaj s tail-om

echo "---------------"

for file in `find -mtime -$period -type f`

do

broj=`cat $file | wc -w`

ukupno=$((broj+ukupno))

done

echo "Ukupno: $ukupno riječi

**3.**

Trebalo je u kazalu (koje je zadano prvim argumentom) i svim njegovim podkazalima pronaći datoteke koje su mijenjane prije n dana (n je zadano drugim argumentom) i ispisati koliko je takvih datoteka, ime, broj rijeci u datoteci i ukupan broj rijeci u svim takvim datotekama. I trebalo je javiti i prekinuti skriptu ako je broj argumenata veći ili manji od traženog broja (dva)

**Rješenje:**  
#!/bin/bash

if [ $# -ne 2 ]

then

echo "krivi broj argumenata"

exit

fi

kazalo=$1

period=$2

ukupno=0

datoteke=0

find $kazalo -mtime -$period -type f | xargs wc -w | sed -r 's/ \*([0-9]\*) (.\*)/\2...\1 riječi/' | head -n -1

echo "---------------"

for file in `find -mtime -$period -type f`

do

datoteke=$((datoteke+1))

broj=`cat $file | wc -w`

ukupno=$((broj+ukupno))

done

echo "Ukupno: $ukupno riječi u $datoteke datoteka"

**4.**

Napisati bash skriptu koja će u zadanom kazalu i njegovim podkazalima pronaći datoteke koje su izmijenjene unutar zadanog perioda, te za svaku od tih datoteka ispisati ime i broj znakova koje datoteka sadrži. Ispisati i ukupan broj znakova u svim pronađenim datotekama, te broj datoteka. Kao argumenti se navode dva broja koji predstavljaju kraj odnosno početak promatranog perioda, izražene kao broj dana mjereno od trenutka pokretanja skripte. Traže se datoteke čije vrijeme izmjene upada u zadani interval i to tako da vrijedi:

Kraj <= Vrijeme\_izmjene < Pocetak.

Ako primjerice pozovemo skriptu sa: zadatak6.sh 0 7 očekujemo da se pronađu sve datoteke koje su mijenjane unutar zadnjih 7 dana, uključujući današnji dan.

Poziv: **zadatak6.sh 1 7** bi trebao ispisati podatke o datotekama koje su mijenjane unutar zadnjih 7 dana, ali ne u posljednjih 24 sata. Skripta treba provjeriti broj argumenata, kao i to da drugi argument ne smije biti manji od prvoga. U slučaju neispravnog pozivanja ispisati uputu o načinu korištenja i izaći iz skripte. (Napomena: predvidjeti pozivanje skripte kao izvršne datoteke.)  
Poziv skripte zadatak6.sh:

$ ./zadatak6.sh 1 10

./Podkazalo2/Skriptni\_1\_bash.log ........ 86088 znakova

./Podkazalo2/Skriptni\_2\_Perl.log ........ 92443 znakova

./Skriptni\_1\_bash.log ........ 86088 znakova

./Skriptni\_1\_bash.tex ........ 85141 znakova

./Podkazalo1/Skriptni\_1\_bash.vrb ........ 631 znakova

./Podkazalo1/Skriptni\_1\_bash.tex ........ 85141 znakova

./Podkazalo1/Skriptni\_2\_Perl.tex ........ 132373 znakova

./definicije.tex ........ 6995 znakova

./Skriptni\_2\_Perl.log ........ 92443 znakova

------------------------------

Ukupno: 667343 znakova u 9 datoteka

**Rješenje:**

#!/bin/bash

if [ $# -ne 2 ]

then

echo " krivi broj argumenata"

exit

fi

pocetak=$1

kraj=$2

broj=0

datoteke=0

if [ $kraj -lt $pocetak]

then

echo "Drugi parametar ne smije bit manji od prvog!"

exit

fi

find -mtime -$kraj -mtime +$pocetak -type f | xargs wc -c | sed -r 's/ \*([0-9]\*) (.\*)/\2...\1 znakova/' | head -n -1

echo "---------------"

for file in ` find -mtime -$kraj -mtime +$pocetak -type f `

do

datoteke=$((datoteke+1))

broj=`cat $file | wc -w`

ukupno=$((broj+ukupno))

done

echo "Ukupno: $ukupno znakova u $datoteke datoteka"

**5.**

Napisati bash skriptu koja će u log-datoteci web poslužitelja analizirati broj pristupa zadanoj stranici. Kao argumenti se navode ime stranice koja se analizira i ime log datoteke. Skripta treba ispisati broj pristupa s pojedinih IP adresa, sortirano (silazno) prema broju pristupa stranici, i to samo za one IP adrese s kojih se pristupalo barem 2 puta. *(varijacija: Podmreže ćemo razlikovati po prva 2 broja IP adrese)*

Skripta treba provjeriti broj argumenata, te čitljivost navedene log datoteke. U slučaju neispravnog pozivanja ispisati uputu o načinu korištenja i izaći iz skripte. (Napomena: predvidjeti pozivanje skripte kao izvršne datoteke.)

Ispis oblikovati prema primjeru u nastavku, a za testiranje se može koristiti log-datoteka iz laboratorijske vježbe.

**Rješenje:**

#!/bin/bash

if [ "$#" -ne 2 ]

then

echo "krivi broj argumenata"

exit

fi

stranica=$1

log\_dat=$2

if [ ! -r $ log\_dat ]

then

echo "datoteka nije citljiva"

exit

fi

echo "-------------------------"

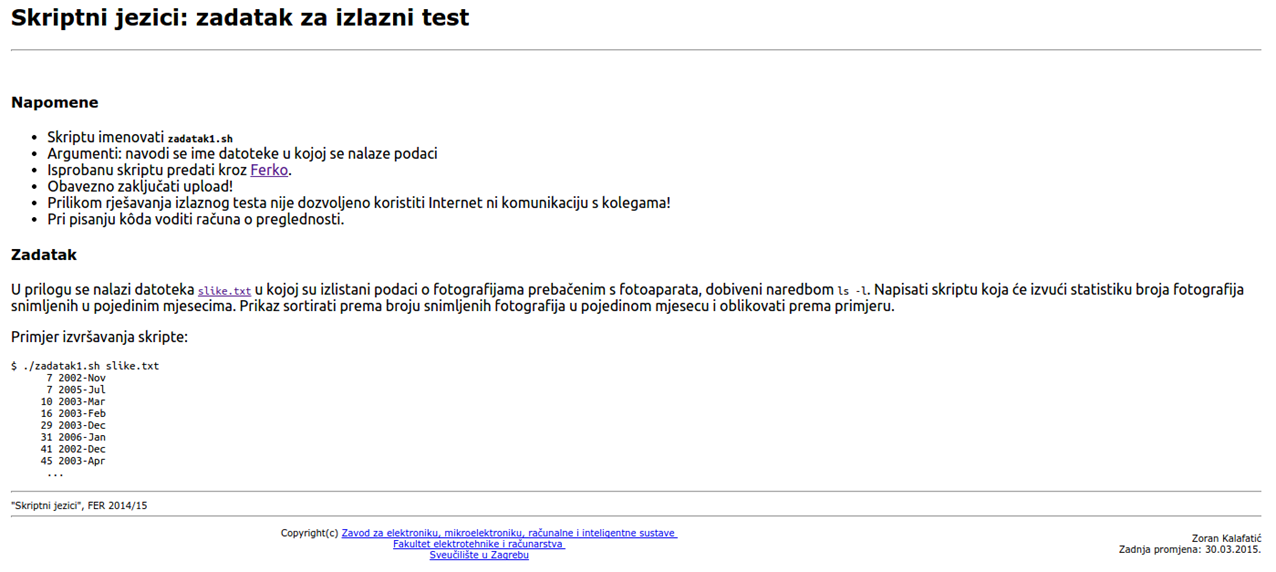
echo "Broj pristupa stranici: $stranica"

echo " IP adresa | Br.pristupa "

echo "-------------------------"

cat $ log\_dat | cut -f 1 -d " "| sort | uniq -c | sort -n -r | sed -r 's/([0-9]+)(.\*)/\2 : \1/'

**6.**



**Rješenje:**

#!/bin/bash

if [ $# -ne 1 ]

then

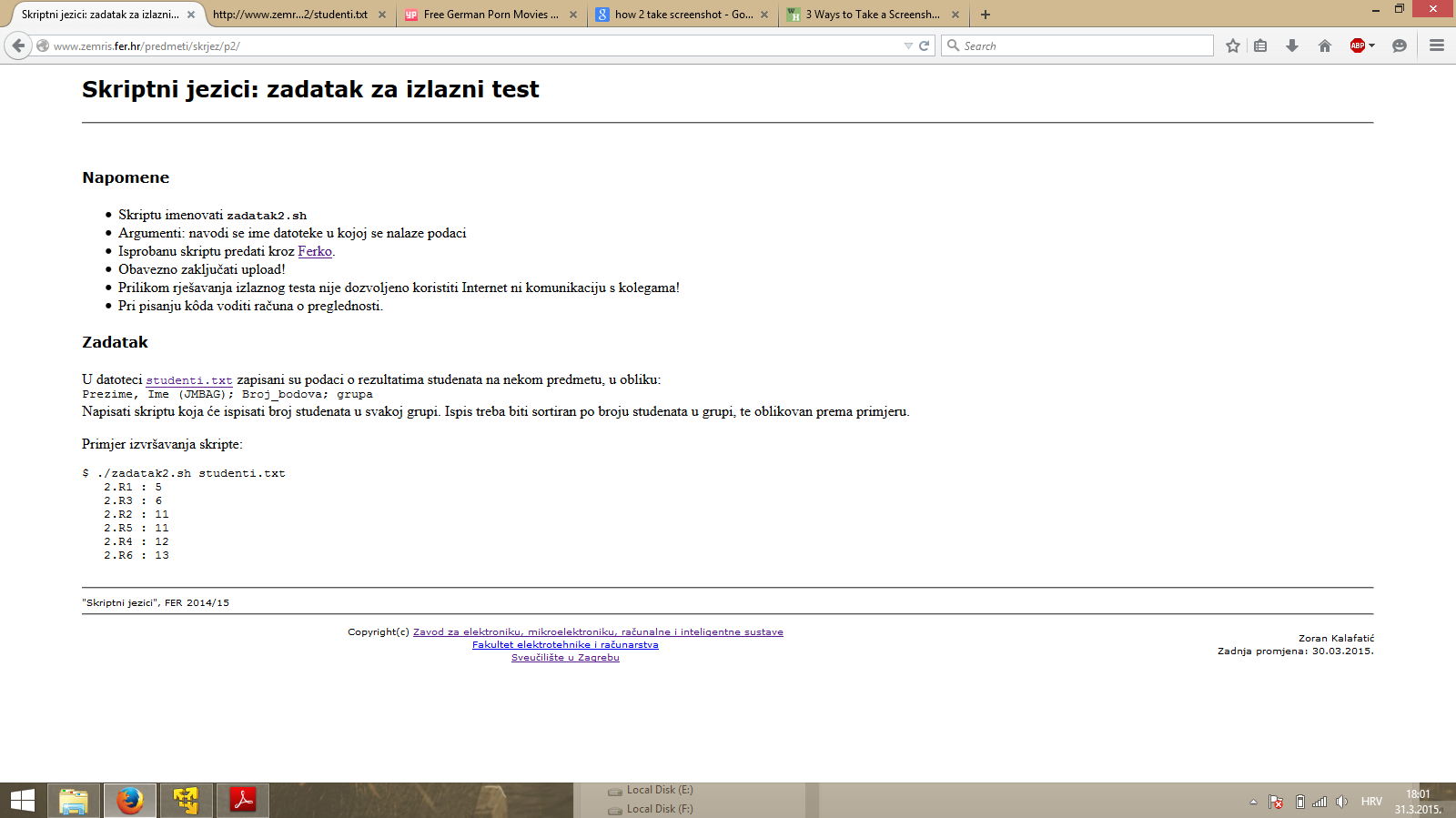
echo "krivi broj argumenata"

exit

fi

slike=$1

cat $slike | tr -s " " | cut -d " " -f 6,8 | sed -r 's/([a-zA-Z]{3}) ([0-9]{4})/\2 \1/' | sort -n | uniq -c | sort -n

**7.** 

**Rješenje:**  
#!/bin/bash

if [ $# -ne 1 ]

then

echo "krivi broj argumenata"

exit

fi

studenti=$1

cat $studenti | cut -d ';' -f 3 | sort -n | uniq -c | sort -n

**8.**

U prilogu je datoteka passwd.txt, u kojoj su zapisani podaci o korisnicima jednog računala. Po jedan redak pripada svakom korisniku, a unutar svakog retka su polja razdvojena dvotočkama. Prvo polje je korisničko ime, drugo polje je lozinka koja se nalazi u jednoj drugoj datoteci što je naznačeno slovom x. Potom dolazi korisnički identifikator (32-bitni broj) i identifikator grupe (također 32-bitni broj). Zatim je upisano ime i prezime studenta razdvojeno razmakom, matični direktorij korisnika koji sadrži i godinu upisa te na kraju ljuska koju korisnik upotrebljava. Zadatak je napisati skriptu koja će generirati popis prezimena po padajućoj popularnosti, oblikovan prema primjeru.

Primjer izvršavanja skripte:

$ ./zadatak3.sh passwd.txt

22 Horvat

20 Tomic

20 Kovacevic

14 Stimac

14 Babic

12 Maric

11 Peric

11 Martinovic

11 Markovic

10 Tolic

10 Simic

10 Pavlovic

10 Pavic

10 Kovacic

10 Kovac

10 Brkic

...

**Rješenje:**

#!/bin/bash

if [ $# -ne 1 ]

then

echo "krivi broj argumenata"

exit

fi

passwd=$1

cat $passwd | cut -d ':' -f 5 | cut -d " " -f 2 | cut -d "," -f 1 | sort | uniq -c | sort -nr

**9.**

ko argumenti se predaju direktorij sa skriptama i broj redaka.

drugi argument je opcionalan i ako nije zadan, pretpostavljena vrijednost je 6.

moraš napraviti provjeru za prvi argument dal je stvarno direktorij.

tvoja skripta mora za svaku skriptu iz tog direktorija ispisati prvih nekolko redaka (koji je zadan s drugim argumentom, odn 6 ako nije).  
  
**Rješenje:**

#!/bin/bash

if [ $# -lt 1 -o $# -gt 3 ]

then

echo "krivi broj argumenata"

exit

fi

kazalo=$1

if [ $# -eq 2 ]

then

broj=$2

else

broj=6

fi

if [ ! -d $kazalo ]

then

echo "$kazalo nije direktorij"

exit

fi

for file in `find $kazalo -maxdepth 1 -type f`

do

if [ ! -d $file ]

then

echo $file

head $file -n $broj

echo "---------"

fi

done

**10.**

U datoteci se nalazi popis riječi. U prilogu se nalazi jedna takva datoteka (words.txt). Napisati bash skriptu koja će ispisati frekvencije riječi koje počinju s pojedinim slovom, pri čemu ne treba praviti razliku između velikih i malih slova. Ispis treba sortirati silazno po frekvencijama, kao u primjeru (ispisuje se broj riječi koje počinju s navedenim slovom i samo slovo). Kao argument se navodi ime datoteke rječnika. Skripta treba provjeriti broj argumenata, te je li argument valjano ime datoteke. U slučaju neispravnog pozivanja treba ispisati poruku o pogrešci.

Primjer izvršavanja skripte:

$ ./zadatak5.sh words.txt

2738 s

2514 c

1955 p

1767 a

1671 b

1461 m

1365 d

1326 t

1077 h

1050 i

1043 e

1038 r

1015 f

910 g

896 l

714 w

538 n

505 o

418 v

309 k

305 j

201 u

133 q

111 y

60 z

13 x

2 1

1 9

1 8

1 7

1 6

1 5

1 4

1 3

1 2

**Rješenje:**

#!/bin/bash

if [ "$#" -ne 1 ]

then

echo "krivi broj argumenata"

exit

fi

words=$1

if [ ! -f $words ]

then

echo "ime datoteke nije valjano"

exit

fi

sort $words | cut -c 1 | uniq -c -i | sort -n -r