

VSO - Laboratorjske vaje: NALOGE

Naloge so točkovane. Za uspešen zagovor morate zbrati vsaj 8 točk in dobiti vsaj 3 točke v vsaki izmed skupin (naloge za lupino, naloge za programski jezik C).

NALOGE ZA LUPINO:

1. (1 točka)

Napišite skript, ki v trenutnem direktoriju preimenuje datoteke s podano končnico v drugo končnico. Končnici sta podani v ukazni vrstici.

Primer uporabe:

```
$ preimenuj txt doc
```

bo preimenoval:

```
aaa.txt v aaa.doc
```

```
Juhutxt v Juhudoc
```

...

* pri reševanju si lahko pomagate z ukazom sed (lahko pa uporabite tudi awk).

2. (1 točka)

Napišite skript, ki vsakih pet minut preveri, če se na sistemu pojavi nek proces in ga poskusa ubiti. Ime procesa podamo v ukazni vrstici.

Primer uporabe:

```
$ ubij xeyes
```

bo na računalniku vsakih pet minut preveril, če teče kak proces z imenom netscape, ter ga bo poskušal ubiti. Navodilo: pri tem obvezno uporabite ukaz kill.

*z ukazom ps in primernimi opcijami izpišete pognane procese; poiščete xeyes (grep) in izrežete (cut) PID, ki ga uporabite v kill. Skript ponovno poženete čez 5 minut z ukazom at.

3. (1 točka)

Napišite skript, ki pregleda datoteko /etc/passwd in izpiše število procesov za vsakega uporabnika.

Primer izpisa:

```
root 45
```

peter 2
vaje 8

* pri reševanju preučite strukturo datoteke /etc/passwd. Z ukazom cut lahko pridete do seznama vseh uporabnikov (zanimajo vas prva polja v omenjeni datoteki). Nato s pomočjo ps in grep pridete do procesov posameznega uporabnika. Te le še preštete z ukazov wc.

4. (2 točki)

Napišite skript, ki preišče vse datoteke v podanem direktoriju in njegovih poddirektorijih in izpiše najnovejšo datoteko (po datumu zadnje spremembe).

Primer izpisa:

```
$ najnovejsa /usr/etc  
/usr/etc/httpd/httpd Oct 25 12:16
```

*uporabite lahko ukaz find z ustreznimi stikali. Zelo elegantno rešitev dobite tudi z uporabo funkcije in rekurzije.

5. (2 točki)

Napišite skript, ki beleži dnevnik dosegljivosti računalnikov. Kot argument mu podamo datoteko, ki vsebuje spisek računalnikov, program pa naj vsakih pet minut preveri, če so dosegljivi. V neko datoteko (dnevnik) naj zapisuje, če kateri od računalnikov ni dosegljiv, njegovo ime in čas, ko ni bil dosegljiv.

Primer uporabe:

```
$ preverjaj spisek_racunalkov.txt
```

izpisuje v dnevnik:

```
io.fri.uni-lj.si ni dosegljiv ob 15:31  
io.fri.uni-lj.si ni dosegljiv ob 15:36  
io.fri.uni-lj.si ni dosegljiv ob 15:41  
verbena.fe.uni-lj.si ni dosegljiv ob 15:41  
...
```

* pomagajte si z ukazom ping. spisek računalnikov preberite v spremenljivko, nato pa v zanki for 'pingajte' vsak računalnik. Če se računalnik ne odzove (rezultat ping), izpišite sporočilo v dnevnik (date). Za ponoven zagon čez 5 minut uporabite ukaz at.

6. (3 točke)

Napišite skript, ki izpiše, kolikokrat ter koliko časa (skupno) so bili uporabniki v nekem času prijavljeni na sistemu. Čas podamo z dvema datumoma kot argumenta ukazne vrstice.

Primer izpisa:

```
$ prijava 15/10 20/10
vaje 5 1 dan 12:34
demsared 11 5 dni 1:12
blazlena 1 0 dni 0:01
...
```

* pri reševanju preučite ukaz last, pomagajte pa si lahko tudi s programčkom v C-ju, ki izračuna razliko med dvema datumoma.

7. (3 točke)

Preučite jezik za pregledovanje tekstnih datotek awk (oz. gawk - GNU awk) ter v njem napišite program, ki za vhod vzame neko HTML datoteko ter na standardni izhod izpiše seznam vseh povezav in slik, ki so vključene v datoteko (z <a HREF...> ali).

Primer uporabe:

```
$ gawk -f preveri.awk lalala.html
slika lala.gif
povezava http://www.fri.uni-lj.si
povezava http://www.ijs.si/slo
slika slike/kuku.jpg
...
```

Skript preizkušajte s preprosto html datoteko; dobite jo npr. z wget <http://www.amtdw.com>. Skript za awk naj ima dve vrstici, na začetku vrstic je primeren regularni izraz, ki izbere vrstice, v katerih je '<a href' oziroma '<img src'. Nato vrstico z ukazom split prerežite tako, da dobide del za npr. '<a href="'. Rezultat nato prerežite še pri narekovaju, ki zaključuje sklic...

8. (točke po dogovoru)

Zamisli si problem in ga reši s programom v lupini (po dogovoru).