

INF3331 - Oblig1

Gøran Frost

September 19, 2014

1 Bash

1.1 find_new_files.sh

“List opp alle filer som er modifisert de siste n dagene. La utskriften inkludere og være sortet på filstørrelse.”

```
./find_new_files.sh dir n
```

Scriptet `find_new_files.sh` leter igjennom *dir* etter filer (*find* som har blitt endret de siste n dager (*ctime*, og så for hver slik fil skriver ut diskforbruket (*du* til filen og filnavnet.

1.2 find_word.sh

“Find alle filer som inneholder et gitt ord. Merk at vi snakker om på innsiden av filen, ikke filnavnet.”

```
./find_word.sh dir word
```

Scriptet leter igjennom *dir* etter alle filer og mapper (*find*). Så går vi igjennom alle resultatene, og om de er filer sjekker vi om de inneholder *word* ved hjelp av *grep*. Hvis filen inneholder *word*, skriver vi ut filnavnet.

1.3 sized_delete.sh

“Slett alle filer i filtreet med størrelse større enn en gitt verdi. Størrelsen er gitt i kilobyte. Print ut navnene på filene som slettes.”

```
./sized_delete.sh dir size
```

Her går vi også igjennom alle filene i *dir* (*find -type f*, og sjekker enkelt ved hjelp av *stat* størrelsen på filen. Dette sammenlignes med $size * 1000$ (for å gjøre om fra KB), og så sletter vi filen om den er større enn *size* KB.

1.4 sort_file.sh

“Sorter linjene i en fil og lagre dem i en ny fil.”

```
./sort_file file1 file2
```

Siden vi allerede har et program som sorterer linjer i filer (*sort*), sender vi bare filnavnene til denne.

2 Python

bruk

```
./generate_file_tree.py [-sdDev] [--seed] target folders files
```

argumenter

```
-s, --size:      maks filstørrelse i KB
-d, --depth:     maks katalogdybde
-S, --start:     mtime/atime start
-e, --end:       mtime/atime end
-v, --verbose:   print kataloger/filer som opprettes
--seed:         seed for RNG
target:         rotkatalog for filtreet
folders:        maks antall underkataloger i hver katalog
files:          maks antall filer i hver katalog
```

Brukseksempel:

```
./generate_file_tree.py -s 199 -v -d 5 tree/ 4 4
```

2.1 random_string()

```
random_string(length, prefix, legal_chars)
```

parametere

```
length : int
    lengde på returnert streng, minus prefix. default: 6
prefix : string
    prefix for strengen. default = ""
legal_chars : string
    streng av tilatte tegn. default = "[\w]"
```

returnerer

```
res : string
```

2.2 generate_tree()

generate_tree(target, dirs, rec_depth, verbose)

parametere

- target : string
 - path til rotkatalogen til treet
- dirs : int
 - maksimum antall underkataloger per katalog
- rec_depth : int
 - maks dybde fra rotkatalogen
- verbose : bool
 - hvis True, print alle kataloger som opprettes

2.3 populate_tree()

populate_tree(target, files, size, start_time, end_time, verbose)

parametere

- target : string
 - path til katalogen som filer skal lages i
- files : int
 - maks antall filer i denne katalogen
- size : int
 - maks størrelse på hver fil, i KB
- start_time : int
 - minste mulige verdi for atime / mtime på filen (unix-format)
- end_time : int
 - største mulige verdi for atime / mtime på filen (unix-format)
- verbose : bool
 - hvis True, print alle filer som opprettes

2.4 Argument Parsing

argparse-modulen brukes for å håndtere alle argumenter.