

# Λειτουργικά Συστήματα

Δημητριάδης Βασίλειος // 2116104 // LAB 02

## Μέρος 2ο

2.

- a. 

```
#!/bin/bash
ls -d */
var1=$(ls -d */ | sort | wc -l)
echo "Total directories: $var1"
```
- b. 

```
#!/bin/bash
read -p "Please type a PATH: " var_path
if [ -d "$var_path" ]; then
    ls -a $var_path
    var2=$(ls -a $var_path | wc -l)
    echo "Total elements found: $var2"
else
    echo "No such file or directory"
fi
```
- c. 

```
#!/bin/bash
var1=$(ls -a /etc | wc -l)
echo "A total of $var1 items found in /etc"
read -p "Type an alphanumeric character: " var2
ls -d /etc/*$var2*
var3=$(ls -d /etc/*$var2* | wc -l)
echo "A total of $var3 items found in /etc/*$var2*"
```

3. Ο κώδικας αυτός δημιουργεί ένα αρχείο "journal-file" στο HOME directory, στην δικιά μ περίπτωση στο **/home/vasil7112**. Στην συνέχεια γράφει ένα «Ημερολόγιο». Πρώτα κάνει append στο αρχείο την ημερομηνία. Στην συνέχεια διαβάζει ένα όνομα και αφού το κάνει append και αυτό στο αρχείο, συνεχίζει διαβάζοντας το περιεχόμενο του «ημερολογίου».

Την έξοδο του προγράμματος την κάνω με την χρήση Ctrl-C, γιατί αλλιώς μπορώ να γράφω επ άπειρον. Θεωρητικά θα έπρεπε να κάνει στο τέλος print μία διαχωριστική γραμμή «-----» που στην περίπτωση μου δεν γίνεται αφού κάνω Ctrl-C!

**Παράδειγμα εξόδου:**

## Μέρος 3ο

- a. **[aeiou]{2,}** Οποιοσδήποτε απο τους χαρακτήρες a,e,i,o,u ακριβώς 2 φορές

- b. **2.+** Οποιαδήποτε λέξη/πρόταση(μέχρι να πετύχουμε \n) που ξεκινάει με τον αριθμό 2. Πχ 2abc, 22, 23a 2z
- c. **2\$** Να τελειώνει με την λέξη «2» στο τέλος.
- d. **^[A-K]** Να ξεκινάει η λέξη με ένα γράμμα απο το A μέχρι το K. Πχ ABc Kvr
- e. **^GNU** Να ξεκινάει η λέξη με την λέξη GNU. Πχ GNUvsr, GNUDrs
- f. **..cept** Κάθε λέξη με μήκος 6 που τα τελευταία 4 γράμματα είναι cept. Πχ daccept, hiecept
- g. **t[wo]**ο Η λέξη two ή too
- h. **[^c]ode** Λέξη με μήκος 4 που να μην ξεκινάει με το γράμμα c και τα 3 τελευταία γράμματα να είναι «ode»
- i. **^[[:upper:]]** Λέξη που να ξεκινάει με έναν χαρακτήρα απο τους [, :, ;, u,p,e,r και να συνεχίζει με το «]» Π.χ: :] u] κλπ
- j. **([A-Za-z]\*)** Πρόταση που δεν έχει αριθμικά μέσα Π.χ στην πρόταση «Welcome mr Vasil7112» θα πιάσει το «Welcome mr Vasil»

## Μέρος 4ο

1. 

```
#!/bin/bash
read -p "Please type a year: " year
if [ $((year % 4)) -ne 0 ]; then
    echo "It is a common year!"
elif [ $((year % 100)) -ne 0 ]; then
    echo "It is a leap year!"
elif [ $((year % 400)) -ne 0 ]; then
    echo "It is a common year!"
else
    echo "It is a leap year!"
fi
```
2. 

```
#!/bin/bash
if [ $EUID -ne 0 ]; then
    echo "You need to run this script as superuser"
else
    clobberEnabled=$(set -o | grep noclobber | sed 's/noclobber//g' | tr -d
"[:space:]")
    echo $clobberEnabled
    if [ $clobberEnabled = "on" ]; then
        echo "Clobber is enabled!"
    else
        echo "Clobber is disabled!"
    fi
fi
```
3. Διαβάζει απο το αρχείο /etc/passwd τους χρήστες και ψάχνει να βρει πόσους χρήστες “Benjamin” υπάρχουν (Ignore-Case). Αμα δεν βρει κανέναν δεν εκτυπώνει τίποτα. Αν

βρει έναν εκτυπώνει «I found Benjamin», αν βρει δύο «I found two Benjamins» αλλιώς  
αν βρεί 3+ εκτυπώνει «There are too many Benjamins»