

Οδηγίες – διαβάστε προσεκτικά:

Σκοπός της εργασίας είναι η εξοικείωση στην ανάγνωση, χρήση και ανάπτυξη απλών σεναρίων φλοιού.

Τα σενάρια πρέπει να προετοιμαστούν σε απλό διορθωτή κειμένου. Οι ασκήσεις πρέπει εκτελεστούν στη γραμμή εντολών και να αποθηκευτούν, μαζί με τα σχετικά αποτελέσματα, σε ένα απλό αρχείο κειμένου. Μια απλή λύση είναι με copy-paste από τη γραμμή εντολών σε ένα αρχείο κειμένου. Στη συνέχεια μπορείτε να προσθέσετε τυχόν σχόλια. Στο τέλος μπορείτε να δημιουργήσετε ένα συμπιεσμένο αρχείο που να περιέχει ένα φάκελλο με όλα τα σενάρια φλοιού, την εκτέλεση και τα αποτελέσματα. Χρησιμοποιείτε απλό διορθωτή (editor) σε περιβάλλον Linux, όχι επεξεργαστή κειμένου, μη βάλετε ελληνικούς χαρακτήρες στο όνομα του αρχείου, μη κάνετε οποιαδήποτε διαμόρφωση σε περιβάλλον Windows. Το αρχείο πρέπει να είναι αναγνώσιμο σε απλό διορθωτή κειμένου σε περιβάλλον Linux. Το όνομα του αρχείου πρέπει να είναι itxxxx (όπου xxxx ο κωδικός σας). Φοιτητές που θα στείλουν αρχεία άλλου τύπου (πχ pdf, .doc ή docx κλπ) δεν θα βαθμολογηθούν.

Μέχρι τις 19 Μαΐου 2015 πρέπει να παρουσιαστείτε τουλάχιστο μια φορά στις ημέρες και ώρες εργαστηρίου, με αρχείο κειμένου- προσχέδιο του αρχείου της εργασίας, για να διαπιστωθεί η πρόοδος σας (θα πρέπει να έχετε ολοκληρώσει τουλάχιστο τις δύο πρώτες ασκήσεις). Φοιτητές που δε θα παρουσιαστούν για ενδιάμεση εξέταση δεν θα βαθμολογηθούν. Η παράδοση της εργασίας στο Compuθ θα γίνει μέχρι τις 29 Μαΐου 2015 και η τελική εξέταση θα την αμέσως επόμενη εβδομάδα, με πρόγραμμα που θα ανακοινωθεί.

Στη προφορική εξέταση της εργασίας θα κληθείτε να εκτελέσετε εναλλακτικές λειτουργίες με ίδια ή παρόμοια σενάρια φλοιού. Επομένως, αν βρίσκετε απαντήσεις στο Google, πρέπει να προσπαθείτε να καταλάβετε και το 'γιατί' όχι μόνο να αντιγράψετε το σενάριο φλοιού.

Τα περισσότερα παραδείγματα προέρχονται από τις σημειώσεις του μαθήματος και από το σύνδεσμο <http://www.it.uom.gr/teaching/linux/bash-guide-gr/bash-gr.html>. Επίσης μπορείτε να συμβουλευτείτε τους συνδέσμους <http://www.it.uom.gr/teaching/linux/lsst/index.html> και <http://www.it.uom.gr/teaching/linux/LinuxCommandLine.pdf>.

1. Γράψτε ένα απλό σενάριο φλοιού bash το οποίο να εμφανίζει βασικές πληροφορίες όπως όνομα χρήστη, οικείο και τρέχοντα κατάλογο, ημερομηνία και ώρα, πληροφορίες επεξεργαστή, μνήμης και λειτουργικού συστήματος, αν υπάρχουν άλλοι συνδεδεμένοι χρήστες.

Δημιουργείτε ένα κατάλογο με το όνομα scripts και προσθέστε τον στη μεταβλητή περιβάλλοντος PATH του φλοιού σας. Προσθέστε σχόλια στο σενάριο. Δείξτε την εκτέλεση του σεναρίου συνολικά και βηματικά.

2. Γράψτε ένα σενάριο φλοιού που να δημιουργεί ένα εφεδρικό αντίγραφο ενός υποκαταλόγου σας με την εντολή tar σε κάποιο διαφορετικό τμήμα του συστήματος αρχείων. Μετά συμπιέστε το εφεδρικό αντίγραφο. Στη συνέχεια αντιγράψτε με scp το συμπιεσμένο αρχείο σε ένα απομακρυσμένο υπολογιστή όπου έχετε πρόσβαση.

Γράψτε ένα σενάριο φλοιού που εκτελεί την αντίστροφη λειτουργία, δηλαδή επαναφέρει ένα κατάλογο από το εφεδρικό αντίγραφο του. Βεβαιωθείτε ότι όταν το αντίγραφο ασφαλείας αποσυμπιεστεί θα δημιουργήσει ένα φάκελο και όχι σκόρπια αρχεία.

Χρησιμοποιείτε ονόματα τοπικών μεταβλητών για όλα τα ονόματα διαδρομής των καταλόγων και του απομακρυσμένου υπολογιστή.

3. Με βάση το σενάριο της άσκησης 2 δημιουργείτε δύο εναλλακτικά σενάρια με τις εξής διαφοροποιήσεις

3.1. Οι τοπικές μεταβλητές λαμβάνουν τιμές από ορίσματα της γραμμής εντολών. Προσθέστε μήνυμα για τη σωστή χρήση του σεναρίου (όπως στην επόμενη άσκηση).

3.2 Οι τοπικές μεταβλητές λαμβάνουν τιμές διαλογικά (εντολές echo, read)

3.3 Χρησιμοποιείτε για τους διαλόγους τη γραφική διεπιφάνεια zenity. Για να δείτε τη χρήση της εκτελέστε man zenity ή zenity --help

4. Ποια είναι η λειτουργία των τριών σεναρίων φλοιού που ακολουθούν; Σχολιάστε το αναλυτικά. Εκτελέστε τις εντολές σε κάθε διάυλο (pipe) σταδιακά ώστε να αντιληφθείτε τη λειτουργία τους. Αποτυπώστε όλες τις ενδιάμεσες εκτελέσεις των σεναρίων.

```
#!/bin/bash
```

```

# This script does a very simple test for checking disk space.
space=`df -h | awk '{print $5}' | grep % | grep -v Use | sort -n | tail -1 | cut -d "%"
-f1 -`
alertvalue="80"
if [ "$space" -ge "$alertvalue" ]; then
    echo "At least one of my disks is nearly full!" | mail -s "daily diskcheck" root
else
    echo "Disk space normal" | mail -s "daily diskcheck" root
fi

#!/bin/bash
# This script can clean up files that end up in *.old.
USAGE="Usage: $0 dir1 dir2 dir3 ... dirN"
if [ "$#" == "0" ]; then
    echo "$USAGE"
    exit 1
fi
while (( "$#" )); do
    if [[ "$(ls $1)" == "" ]]; then
        echo "Empty directory $1, nothing to be done."
    else
        find $1 -type f -a -name '*.~' -exec rm -f {} \;
    fi
    shift
done

#!/bin/bash
# This script gives information about a file.
FILENAME="$1"
echo "Properties for $FILENAME:"
if [ -f $FILENAME ]; then
    echo "Size is $(ls -lh $FILENAME | awk '{ print $5 }')"
    echo "Type is $(file $FILENAME | cut -d":" -f2 -)"
    echo "Inode number is $(ls -li $FILENAME | cut -d" " -f1 -)"
    echo "$(df -h $FILENAME | grep -v Mounted | awk '{ print "On",$1", \
which is mounted as the",$6,"partition."}')"
else
    echo "File does not exist."
fi

```

5. Γράψτε ένα σενάριο φλοιού που ελέγχει τη πληρότητα του οικείου καταλόγου σας σε σχέση και εμφανίζει μήνυμα ειδοποίησης αν η πληρότητά του είναι πάνω από 80%. Το μήνυμα στέλνεται στην οθόνη όταν ο χρήστης ανοίγει ένα παράθυρο φλοιού και αποθηκεύεται και σε ένα αρχείο με όνομα \$HOME/system-messages/quota.ddmmmyyyy, όπου ddmmmyyyy αντιστοιχεί σε ημερομηνία δημιουργίας του μηνύματος. Αν υπάρχουν διαδοχικά μηνύματα τότε εμφανίζεται μόνο το τελευταίο στην οθόνη αλλά φυσικά αποθηκεύονται όλα.