



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΜΑΤΙΚΗΣ

# Αθανάσιος Εξάρχου

1<sup>η</sup> Εργασία στο μάθημα **Λειτουργικά Συστήματα**

Ταύρος, 21 Δεκεμβρίου 2023

## Περιεχόμενα

<b>Άσκηση 1</b> .....	<b>3</b>
Κώδικας .....	3
Ενδεικτικές εκτελέσεις (screenshots): .....	4
Γενικά Σχόλια/Παρατηρήσεις .....	5
Με δυσκόλεψε / δεν υλοποίησα.....	5
<b>Άσκηση 2</b> .....	<b>6</b>
Κώδικας .....	6
Ενδεικτικές εκτελέσεις (screenshots): .....	7
Γενικά Σχόλια/Παρατηρήσεις .....	8
Με δυσκόλεψε / δεν υλοποίησα.....	8
<b>Άσκηση 3</b> .....	<b>9</b>
Κώδικας .....	9
Ενδεικτικές εκτελέσεις (screenshots): .....	9
Γενικά Σχόλια/Παρατηρήσεις .....	9
Με δυσκόλεψε / δεν υλοποίησα.....	10
<b>Άσκηση 4</b> .....	<b>11</b>
Κώδικας .....	11
Ενδεικτικές εκτελέσεις (screenshots): .....	12
Γενικά Σχόλια/Παρατηρήσεις .....	13
Με δυσκόλεψε / δεν υλοποίησα.....	13
<b>Άσκηση 5</b> .....	<b>14</b>
Κώδικας .....	14
Ενδεικτικές εκτελέσεις (screenshots): .....	15
Γενικά Σχόλια/Παρατηρήσεις .....	15
Με δυσκόλεψε / δεν υλοποίησα.....	15
<b>Συνοπτικός Πίνακας</b> .....	<b>16</b>

# Άσκηση 1

## Κώδικας

To shell script που δημιουργήθηκε μαζί με τα σχόλια είναι:

```
#!/bin/bash

#variable initialization
sChars='' #variable containing special characters
numOfChars=0 #number of characters for each type of password
if [ $# -eq 1 ] #accepts only one argument
then
    case "$1" in #accepts the first argument given by the user
        Easy*|easy*|EASY*) #different kinds of arguments have been made acceptable making it
easier for the user
            #changes the number of characters to 8(no need to change the value of
'sChars' since its needed value for this specific use has already been initialized)
            numOfChars=8
            #informs the user via appropriate message
            echo "Creating a random 'easy' password that consists of $numOfChars letters
of the English alphabet and/or numbers 0-9:"
            ;;
        Medium*|medium*|MEDIUM*) #different kinds of arguments have been made acceptable
making it easier for the user
            sChars='.!#' #adds the needed special characters intended for the 'medium'
password
            numOfChars=12 #changes the number of characters to 12
            #informs the user via appropriate message
            echo "Creating a random 'medium' password that consists of $numOfChars
letters of the English alphabet and/or numbers 0-9 and/or the characters $sChars:"
            ;;
        Hard*|hard*|HARD*) #different kinds of arguments have been made acceptable making it
easier for the user
            sChars='.!#@,?^$' #adds the needed special characters intended for the 'hard'
password
            numOfChars=20 #changes the number of characters to 20
            #informs the user via appropriate message
            echo "Creating a random 'hard' password that consists of $numOfChars letters
of the English alphabet and/or numbers 0-9 and/or the characters $sChars:"
            ;;
    *)
```

```

        #informs the user via appropriate message and exits
        echo -n "Invalid input! Exiting..."
        ;;
    esac
    #prints a random '$numOfChars'-character password and then a newline
    tr -dc [:alnum:]$sChars < /dev/urandom | head -c $numOfChars; echo
else
    echo "Please give only one argument as input. Exiting..."
fi

```

### Ενδεικτικές εκτελέσεις (screenshots):

- Επιτυχής εκτέλεση script

```

xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_1.sh easy
Creating a random 'easy' password that consists of 8 letters of the English alphabet and/or numbers 0-9:
L0mRpHQV
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_1.sh Easy
Creating a random 'easy' password that consists of 8 letters of the English alphabet and/or numbers 0-9:
kNZB5CvL
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_1.sh EASY
Creating a random 'easy' password that consists of 8 letters of the English alphabet and/or numbers 0-9:
VSClWiEs
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$

```

### Παρατηρήσεις:

Έχω υλοποιήσει το κώδικα με τέτοιο τρόπο ώστε να δέχεται το όρισμα του χρήστη ακόμη και αν αυτό είναι 'easy' ή 'Easy' ή 'EASY' και να εκτυπώνει έναν 'εύκολο' κωδικό. Το ίδιο ισχύει για 'μέτριο' και 'δύσκολο' κωδικό (medium/Medium/MEDIUM/hard/Hard/HARD αντίστοιχα).

```

xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_1.sh easy123abc
Creating a random 'easy' password that consists of 8 letters of the English alphabet and/or numbers 0-9:
HUYetv3G
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_1.sh
Please give only one argument as input. Exiting...
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_1.sh easy medium
Please give only one argument as input. Exiting...
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_1.sh 1easy
Invalid input! Exiting...
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$

```

- Έλεγχος παραμέτρων χρήστη

### Παρατηρήσεις:

Έχω υλοποιήσει το κώδικα με τέτοιο τρόπο ώστε να δέχεται το όρισμα του χρήστη ακόμη και αν αυτό είναι "easy123abc" κλπ, θεωρώντας πως η πρόθεσή του ήταν η δημιουργία ενός τυχαίου 'easy' κωδικού. Το ίδιο ισχύει για 'μέτριο' και 'δύσκολο' κωδικό (medium123abc/Medium123abc/MEDIUM123abc/hard123abc/Hard123abc/HARD123abc αντίστοιχα).

αντίστοιχα).

Στη περίπτωση που δεν δίνεται παράμετρος ή δίνονται πάνω από μία παράμετροι, το πρόγραμμα εκτυπώνει το κατάλληλο μήνυμα και κλείνει.

### **Γενικά Σχόλια/Παρατηρήσεις**

Ωραίο, σύντομο script. Θα ήταν πιο χρήσιμο σε πραγματικές συνθήκες εάν αντί να παίρναμε όρισμα ενημερώναμε, πρώτα μέσω μηνύματος το χρήστη για τις επιλογές του και έπειτα του ζητούσαμε είσοδο για τύπο κωδικού (διότι θεωρητικά ο χρήστης δεν γνωρίζει τα περιεχόμενα του κώδικα).

### **Με δυσκόλεψε / δεν υλοποίησα**

### **Links που χρησιμοποίησα**

<https://www.geeksforgeeks.org/tr-command-in-unix-linux-with-examples/>

<https://askubuntu.com/questions/192203/how-to-use-dev-urandom>

Για τις υπόλοιπες εντολές χρησιμοποίησα κυρίως τα manual που προσφέρει το terminal.

## Άσκηση 2

### Κώδικας

Το shell script που δημιουργήθηκε μαζί με τα σχόλια είναι:

```
#!/bin/bash

#prompt statement to be printed after each iteration
PS3="Choose an option from 1 to 6 inclusively: "

#I use "select" as instructed for the convenience of the user
select option in Hostname Kernel CPU Memory Disk-Usage Exit

do #start of iteration
    case "$option" in #takes the user's choice as input
        #different inputs have been made acceptable to aid the user, in the case that an
        #implementation without the use of "select" is ever needed
        hostname*|Hostname*|HOSTNAME*)
            #informs the user via appropriate message and prints the hostname
            echo -n "System Information for: "; uname -n
            ;;
        kernel*|Kernel*|KERNEL*)
            #informs the user via appropriate message and prints the kernel version
            echo -n "Kernel Version: "; uname -r
            ;;
        cpu*|Cpu*|Cpu*|CpU*|CPU*|cPu*|cpU*|cPU*)
            #used "sed" to print "CPU" instead of "Model name:" and "awk" for a smooth
            #appearance
            lscpu | sed -n 's/Model name:/CPU:/p' | awk '{$1=$1}1'
            ;;
        memory*|Memory*|MEMORY*)
            #used "sed" to print the appropriate message for each circumstance instead of
            #"Mem:" and "awk" for 'stdout' and a smooth appearance
            free -h | sed -n 's/Mem:/Memory: Total:/p' | awk '{$1=$1}1' | awk '{printf $1
            " " $2 " " $3 " "}'
            free -h | sed -n 's/Mem:/Used:/p' | awk '{$1=$1}1' | awk '{printf $1 " " $3 "
            "}'
            free -h | sed -n 's/Mem:/Free:/p' | awk '{$1=$1}1' | awk '{print $1 " " $4}'
            ;;
    esac
done
```

```

disk-usage*|diskusage*|Disk-usage*|Diskusage*|Disk-Usage*|DiskUsage*|DISK-
USAGE*|DISKUSAGE*)
    #used "sed" to print the appropriate message for each circumstance instead of
"total" and "awk" for 'stdout' and a smooth appearance
    df -h --total | sed -n 's/total/Disk Usage: Total:/p' | awk '{$1=$1}1' | awk
'{$printf $1 " " $2 " " $3 " " $4 " "}'
    df -h --total | sed -n 's/total/Used:/p' | awk '{$1=$1}1' | awk '{$printf $1 "
" $3 " "}'
    df -h --total | sed -n 's/total/Free:/p' | awk '{$1=$1}1' | awk '{$print $1 " "
$4}'
    ;;
exit*|Exit*|EXIT*)
    echo "Exiting..." #informs the user via appropriate message
    break #exits
    ;;
*) #case where the input is invalid
    #urges the user to retry via appropriate message
    echo "Invalid input! Try again."
    ;;
esac
done #end of iteration

```

### Ενδεικτικές εκτελέσεις (screenshots):

- Επιτυχής εκτέλεση script

```

xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_2.sh
1) Hostname
2) Kernel
3) CPU
4) Memory
5) Disk-Usage
6) Exit
Choose an option from 1 to 6 inclusively: 1
System Information for: xrathws-ubuntu22
Choose an option from 1 to 6 inclusively: 2
Kernel Version: 6.2.0-37-generic
Choose an option from 1 to 6 inclusively: 3
CPU: AMD Ryzen 5 5600X 6-Core Processor
Choose an option from 1 to 6 inclusively: 4
Memory: Total: 19Gi Used: 2.4Gi Free: 11Gi
Choose an option from 1 to 6 inclusively: 5
Disk Usage: Total: 43G Used: 20G Free: 22G
Choose an option from 1 to 6 inclusively: 6
Exiting...
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$

```

### Παρατηρήσεις:

Στο 4ο option(Memory:...) δεν μου εμφανίζει τα Ghz επειδή έχω AMD, αν δοκιμάσω την ίδια

εντολή σε άλλη συσκευή, με intel για παράδειγμα λειτουργεί κανονικά.

- Έλεγχος παραμέτρων χρήστη

```
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_2.sh
1) Hostname
2) Kernel
3) CPU
4) Memory
5) Disk-Usage
6) Exit
Choose an option from 1 to 6 inclusively: 0
Invalid input! Try again.
Choose an option from 1 to 6 inclusively: 7
Invalid input! Try again.
Choose an option from 1 to 6 inclusively: CPU
Invalid input! Try again.
Choose an option from 1 to 6 inclusively: 6
Exiting...
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$
```

Παρατηρήσεις:

Ο έλεγχος λειτουργεί σωστά αφού δεν δέχεται άλλη απάντηση παρά αριθμό από το 1 μέχρι το 6.

### Γενικά Σχόλια/Παρατηρήσεις

Χρήσιμο script, ρεαλιστικό για πραγματικές συνθήκες λόγω της αλληλεπίδρασης με το χρήστη.

### Με δυσκόλεψε / δεν υλοποίησα

Με δυσκόλεψε αρκετά η χρήση της εντολής awk. Ίσως για αυτό ευθύνεται ο ασυνήθιστος τρόπος με τον οποίο υλοποίησα το κώδικα.

### Links που χρησιμοποίησα

[https://www.thegeekstuff.com/2008/09/bash-shell-take-control-of-ps1-ps2-ps3-ps4-and-prompt\\_command/](https://www.thegeekstuff.com/2008/09/bash-shell-take-control-of-ps1-ps2-ps3-ps4-and-prompt_command/)

<https://stackoverflow.com/questions/2310605/how-to-remove-leading-whitespace-from-each-line-in-a-file>

<https://www.geeksforgeeks.org/sed-command-in-linux-unix-with-examples/>

Για τις υπόλοιπες εντολές χρησιμοποίησα κυρίως τα manual που προσφέρει το terminal.



# Άσκηση 3

## Κώδικας

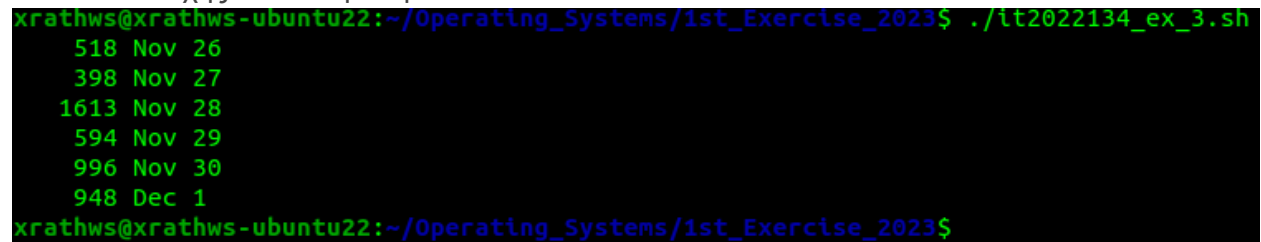
Το shell script που δημιουργήθηκε μαζί με τα σχόλια είναι:

```
#!/bin/bash

#"awk" takes as 'stdin' the contents of "syslog", but before it prints anything,
#that which would have been printed is then given as input to "uniq -c" with the use
of a 'pipe'
#and is instead printed as the number of times it would have appeared together with
the fitting month and date
awk '{print $1 " " $2}' < syslog | uniq -c
```

Ενδεικτικές εκτελέσεις (screenshots):

- Επιτυχής εκτέλεση script



```
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./lt2022134_ex_3.sh
518 Nov 26
398 Nov 27
1613 Nov 28
594 Nov 29
996 Nov 30
948 Dec 1
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$
```

Παρατηρήσεις:

Έχω αφήσει επίτηδες τα κενά μπροστά από τις τιμές για να φαίνονται καλύτερα οι στήλες.

- Έλεγχος παραμέτρων χρήστη

Παρατηρήσεις:

Δεν υπάρχει είσοδος από το χρήστη, οπότε δεν υφίσταται έλεγχος.

## Γενικά Σχόλια/Παρατηρήσεις

Χρησιμοποίησα το "syslog" της εκφώνησης(διότι δεν έχω permission στο /var/log/syslog) το οποίο βρίσκεται στο ίδιο directory που κάνω compile. Αν δεν ίσχυε αυτό θα μπορούσα απλά να πρόσθετα στο κώδικα το αντίστοιχο path.

## Με δυσκόλεψε / δεν υλοποίησα

Το συγκεκριμένο script μου πήρε πολύ χρόνο μέχρι να το ολοκληρώσω σε σχέση με τα δύο

πρώτα scripts, γιατί προσπαθούσα να χρησιμοποιήσω όλες τις “Hint” εντολές μαζί...  
Συνειδητοποίησα όμως μετά από αρκετό debugging, πως οι ημερομηνίες του “syslog” είναι ήδη ταξινομημένες, οπότε υλοποίησα το κώδικα μόνο με τη χρήση των “awk” και “uniq”.

### **Links που χρησιμοποίησα**

<https://www.geeksforgeeks.org/uniq-command-in-linux-with-examples/>

<https://www.geeksforgeeks.org/awk-command-unixlinux-examples/>

Επιπλέον χρησιμοποίησα τα manual που προσφέρει το terminal.

# Άσκηση 4

## Κώδικας

To shell script που δημιουργήθηκε μαζί με τα σχόλια είναι:

```
#!/bin/bash

if [ $# -eq 1 ] #case where only one argument was given
then
    if [ -d "$1" ]
    then #case where the argument provided is a directory
        echo "\"$1\" is a directory." #prints appropriate message
        #smoothen the appearance
        echo "-----"
        echo "The 5 largest files(starting from the largest) in \"$1\" are:" #prints
appropriate message
        #finds the files in the directory, checks their individual disk-usage and prints
the 5 largest ones' name and size
        find "$1" -type f -exec du -h {} \; | sort -rh | head -n 5 | awk '{print $2 "
with a size of " $1}'
        #smoothen the appearance
        echo "-----"
        touch "counter.txt" #creates a file to contain the number of files with more than
one hard links
        #finds the number of files with more than one hard links and places it into
"counter.txt"
        find "$1" -type f -links +1 | grep -c '/' > counter.txt
        if [ $(cat counter.txt) -gt 0 ]
        then #case where the number of files with more than one hard links exceeds zero
            echo "There exist files with more than one hard links in \"$1\" " #prints
appropriate message
            else #case where the number of files with more than one hard links does not
exceed zero
                echo "There exist no files with more than one hard links in \"$1\" " #prints
appropriate message
            fi
        #smoothen the appearance
```

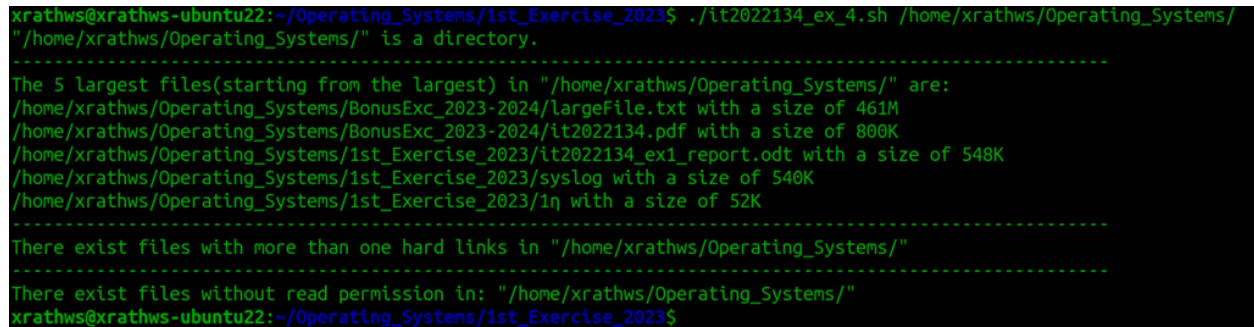
```

echo "-----"
-----"
if find "$1" -type f -exec ls -o {} \; | grep -s '^--' | grep -q '/';
then #case where the number of files without read permission exceeds zero
    echo "There exist files without read permission in: \"$1\" #prints
appropriate message
    else #case where the number of files without read permission does not exceed zero
    echo "There exist no files without read permission in: \"$1\" #prints
appropriate message
    fi
else #case where the argument provided is not a directory
    echo "\"$1\" is not a directory. Exiting..." #informs the user their input is
invalid and exits
    fi
else #case where the number of arguments given does not equal 1
    #informs the user the number of directories they entered is invalid and exits
    echo "Please give just one directory as argument. Exiting..."
fi

```

Ενδεικτικές εκτελέσεις (screenshots):

- Επιτυχής εκτέλεση script



```

xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_4.sh /home/xrathws/Operating_Systems/
"/home/xrathws/Operating_Systems/" is a directory.
-----
The 5 largest files(starting from the largest) in "/home/xrathws/Operating_Systems/" are:
/home/xrathws/Operating_Systems/BonusExc_2023-2024/largeFile.txt with a size of 461M
/home/xrathws/Operating_Systems/BonusExc_2023-2024/it2022134.pdf with a size of 800K
/home/xrathws/Operating_Systems/1st_Exercise_2023/it2022134_ex1_report.odt with a size of 548K
/home/xrathws/Operating_Systems/1st_Exercise_2023/syslog with a size of 540K
/home/xrathws/Operating_Systems/1st_Exercise_2023/1η with a size of 52K
-----
There exist files with more than one hard links in "/home/xrathws/Operating_Systems/"
-----
There exist files without read permission in: "/home/xrathws/Operating_Systems/"
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$

```

Παρατηρήσεις:

Έχω υλοποιήσει το κώδικα με τέτοιο τρόπο, ώστε να προσπελαύνει τα αρχεία των καταλόγων μέσα στο κατάλογο που έδωσε ο χρήστης.

- Έλεγχος παραμέτρων χρήση

```
krathws@krathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./lt2022134_ex_4.sh /home/krathws/Operating_Systems/1st_Exercise_2023/ 2ndARGUMENT
Please give just one directory as argument. Exiting...
krathws@krathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./lt2022134_ex_4.sh NOT_A_DIRECTORY
"NOT_A_DIRECTORY" is not a directory. Exiting...
krathws@krathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./lt2022134_ex_4.sh
Please give just one directory as argument. Exiting...
krathws@krathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$
```

Παρατηρήσεις:

Έχω εξετάσει όλες τις περιπτώσεις όπου ο χρήστης δεν δίνει ένα μόνο υπάρχων κατάλογο.

### Γενικά Σχόλια/Παρατηρήσεις

Όχι τόσο χρήσιμο script σε πραγματικές συνθήκες, εφόσον ο χρήστης πρέπει να γνωρίζει από πριν ότι για την ομαλή εκτέλεση του προγράμματος χρειάζεται να δώσει σαν είσοδο ένα κατάλογο.

### Με δυσκόλεψε / δεν υλοποίησα

Ήταν σύνθετη η χρήση της εντολής find, αφού σε συνδιασμό με το -exec εκτελεί δεύτερη εντολή “μέσα” της.

### Links που χρησιμοποίησα

<https://www.geeksforgeeks.org/find-command-in-linux-with-examples/>

<https://unix.stackexchange.com/questions/480012/how-do-you-count-the-total-number-of-hard-links-within-a-directory>

<https://www.redhat.com/sysadmin/linux-file-permissions-explained>

Επιπλέον χρησιμοποίησα τα manual που προσφέρει το terminal.

# Άσκηση 5

## Κώδικας

To shell script που δημιουργήθηκε μαζί με τα σχόλια είναι:

```
#!/bin/bash

## could have put the site's link inside a variable
## I see however no need for that in the current instance of the exercise

#expiration date variable displays diagnostic information about the SSL connection to the
server,
#extracts the expiry date and prints out its value (errors are placed in /dev/null, so they
essentially vanish)
exp=$(echo | openssl s_client -connect "www.hua.gr":443 2> /dev/null | openssl x509 -noout
-enddate | awk -F= '{print $2}')

#takes the expiry date and converts it to type-integer
expiry=$(date -d "$exp" +%s)
#takes the current date and converts it to type-integer
cur_date=$(date +%s)

echo "Certificate expiration date: $exp" #prints the expiration date
echo -n "Current date: "; date #prints the current date

#compares the ssl certificate expiration date with the current date and prints the
appropriate message
if [ "$expiry" -lt "$cur_date" ];
then #case where the ssl certificate has not expired
    echo "The certificate has expired."
else #case where the ssl certificate has expired
    echo "The certificate has yet to expire."
fi
```

## Ενδεικτικές εκτελέσεις (screenshots):

- Επιτυχής εκτέλεση script

```
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_5.sh
Certificate expiration date: Oct 31 23:59:59 2024 GMT
Current date: Wed 20 Dec 21:17:08 EET 2023
The certificate has yet to expire.
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$
```

Παρατηρήσεις:

Χρησιμοποίησα το [www.hua.gr](http://www.hua.gr) σαν link.

Εκτυπώνω και την ημερομηνία συστήματος του χρήστη για λόγους αποτελεσματικότητας.

- Έλεγχος παραμέτρων χρήστη

Παρατηρήσεις:

Δεν υπάρχει είσοδος από το χρήστη, οπότε δεν υφίσταται έλεγχος. Στη περίπτωση βέβαια που ο προγραμματιστής βάλει ένα link(πχ.: [expired.badssl.com](http://expired.badssl.com)) του οποίου το certificate έχει λήξει, το πρόγραμμα θα εμφανίζει:

```
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$ ./it2022134_ex_5.sh
Certificate expiration date: Apr 12 23:59:59 2015 GMT
Current date: Wed 20 Dec 21:28:29 EET 2023
The certificate has expired.
xrathws@xrathws-ubuntu22:~/Operating_Systems/1st_Exercise_2023$
```

## Γενικά Σχόλια/Παρατηρήσεις

Θα ήταν πιο συνετό ο χρήστης να ήταν αυτός που δίνει το server του οποίου το certificate θέλει να ελέγξει αν έχει λήξει.

## Με δυσκόλεψε / δεν υλοποίησα

```
%R      24-hour hour and minute; same as %H:%M
%s      seconds since 1970-01-01 00:00:00 UTC
%S      second (00..60)
```

Πήρε αρκετό χρόνο μέχρι να βρώ το %s.

## Links που χρησιμοποίησα

<https://www.misterpki.com/openssl-s-client/>

<https://www.openssl.org/docs/man1.1.1/man1/x509.html>

[https://linuxhint.com/what\\_is\\_dev\\_null/](https://linuxhint.com/what_is_dev_null/)

Επιπλέον χρησιμοποίησα τα manual που προσφέρει το terminal.

## Συνοπτικός Πίνακας

1η Εργασία		
	Υλοποιήθηκε (ΝΑΙ/ΟΧΙ/ΜΕΡΙΚΩΣ)	Παρατηρήσεις
Άσκηση 1	ΝΑΙ	Σύντομη υλοποίηση.
Άσκηση 2	ΝΑΙ	Χρήσιμο script για πραγματικές συνθήκες.
Άσκηση 3	ΝΑΙ	Δύσκολη υλοποίηση.
Άσκηση 4	ΝΑΙ	Μη χρήσιμο script για πραγματικές συνθήκες.
Άσκηση 5	ΝΑΙ	Ενδιαφέρον υλοποίηση.