

**MARMARA UNIVERSITY**

**FACULTY OF ENGINEERING**

**COMPUTER SCIENCE & ENGINEERING DEPARTMENT**

**CSE3033**

**OPERATING SYSTEMS**

**Report of Programming Assignment #1**

*A.Tunahan Cinsoy – 150117062*

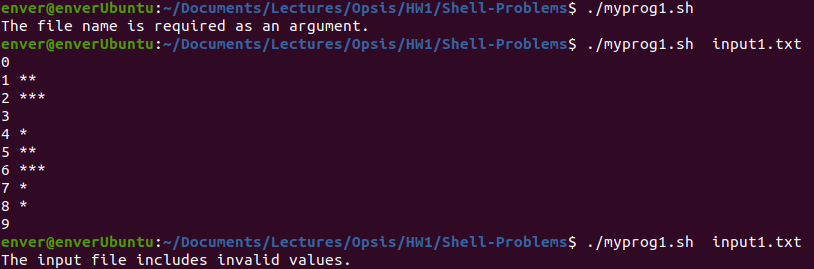
*Enver Aslan – 150115851*

*Cem Güleç – 150117828*

**Explanations About Implementations of Question 1**

In question 1, we simply created a loop that iterates given numbers in text file. While iteration process continues, our script counts the occurrences of each number, and prints occurrences of each number with stars.

Here is the screenshot of our sample executions:



∙ For first execution, we wanted to show that program can’t be run without argument.

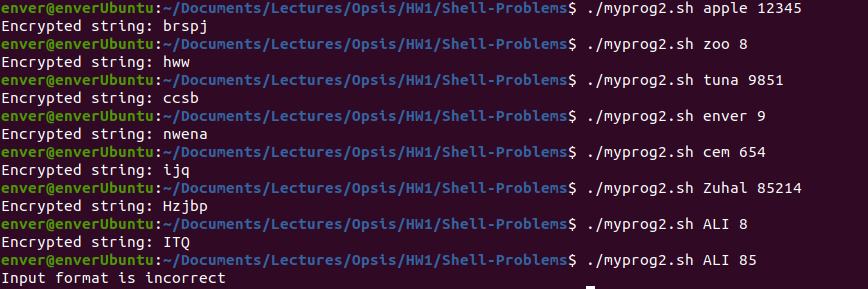
∙ Second execution is the correct execution case.

∙ Third output occurs only if input file contains a value that is not between [0-9]

**Explanations About Implementations of Question 2**

In question 2, we take two input arguments. Based on second numerical argument, we convert first string argument into a ciphered version. For each letter in the string, our script finds another letter in English alphabet advancing over the alphabet corresponding digit times. If there is only one digit as second input argument, our script applies that corresponding advance to all letters in first argument. For the conversion purpose, we’ve used ASCII values of letters.

Here is the screenshot of our sample executions:

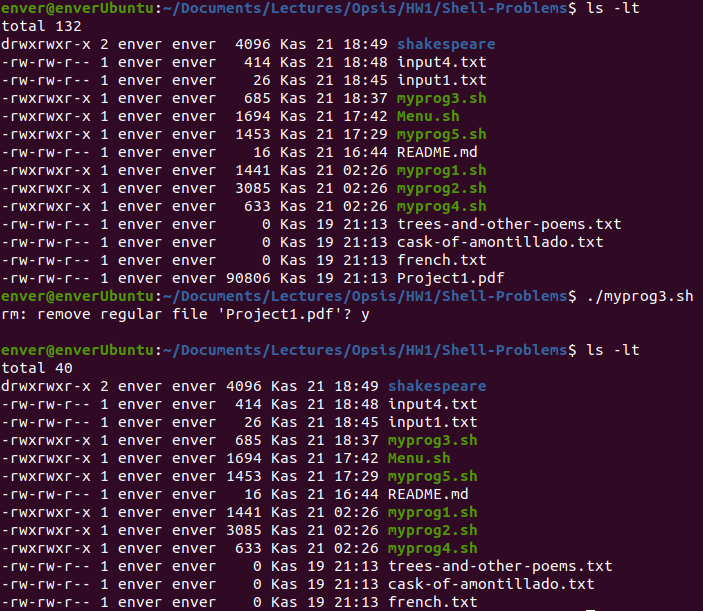


∙ Our script contains ciphering uppercase characters as well as lowercase characters.

**Explanations About Implementations of Question 3**

In question 3, our aim is to delete the oldest file under working directory unless there is an optional pathname as an argument. If there is a pathname, our script simply moves to that location and applies the same rule under that working directory.

Here are some screenshots of our sample executions:



∙ As seen screenshot above, “ls -lt” prompt shows us that “Project1.pdf” is the oldest file under current working directory. So, the script asks us to delete that file and when we type “y”, it deletes successfully.

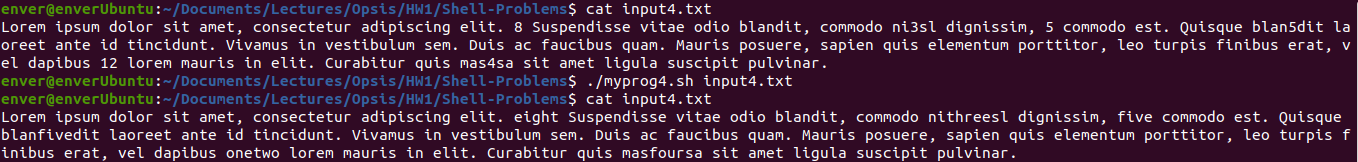


∙ When we give an optional pathname, script goes to that specific folder and applies the same procedure under that working directory.

**Explanations About Implementations of Question 4**

In question 4, our script iterates the content of the given input argument text file with the aim of finding numbers between 0 to 9 both inclusive. When it faces with numbers, it converts these numbers to text.

Here is the screenshot of our sample execution:

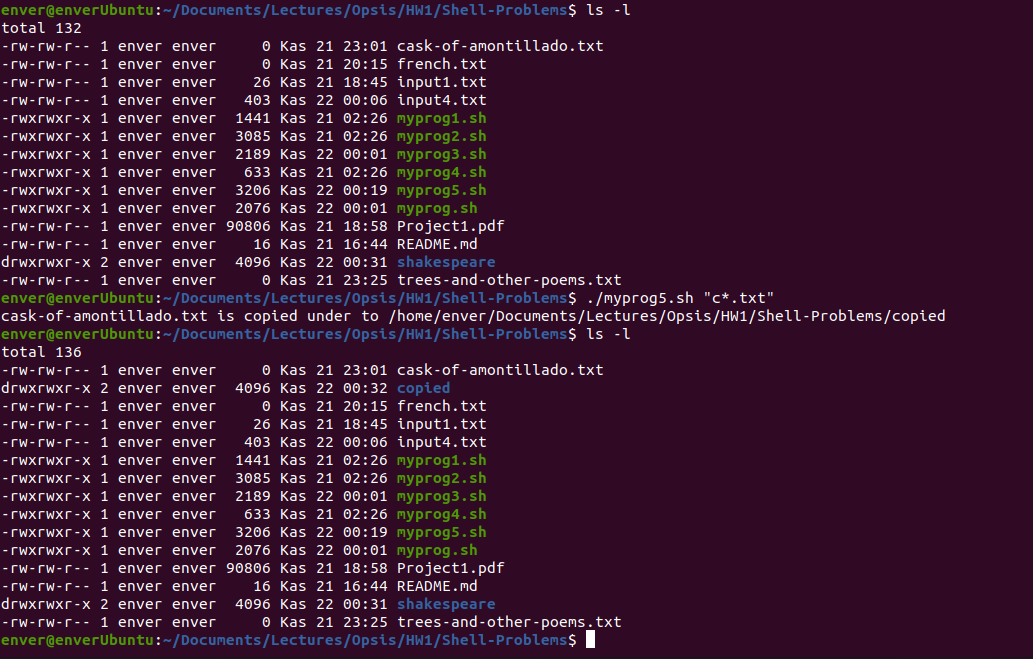


∙ When we randomly put numbers in a text file, script detects and converts these numbers to text.

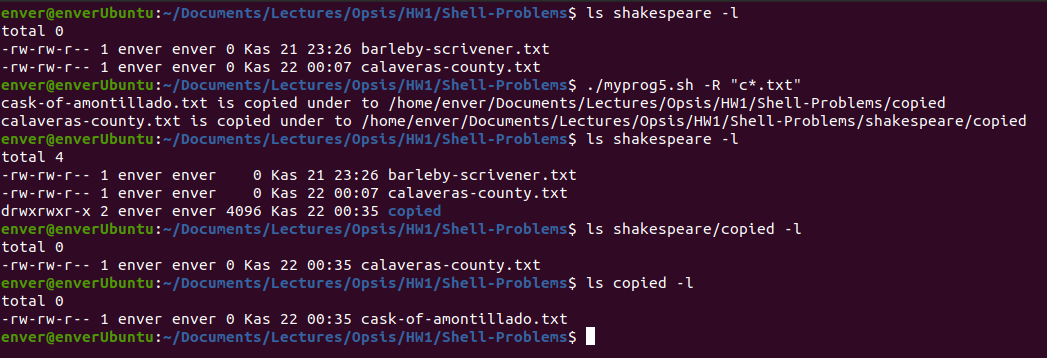
**Explanations About Implementations of Question 5**

In question 5, we wrote a script that takes a wildcard argument and a -R option which is for recursive iteration. Based on wildcard argument, script creates a folder named “copied” and puts files whose name obey the wildcard. If there is an -R option, script works recursively which means that it goes to every folder that is located under working directory and applies the same procedure to these folders as well.

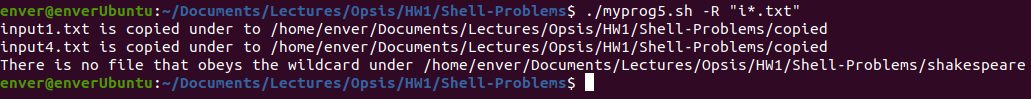
Here are some screenshots of our sample executions:



∙ When we execute myprog5.sh without -R flag, it creates a copied folder under working directory and copies files which obey given wildcard argument.



∙ When we execute myprog5.sh with -R flag, program understands that it will work recursively, then it creates a copied folder under working directories and copies files which obey given wildcard argument.

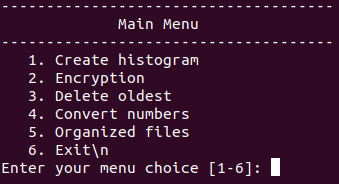


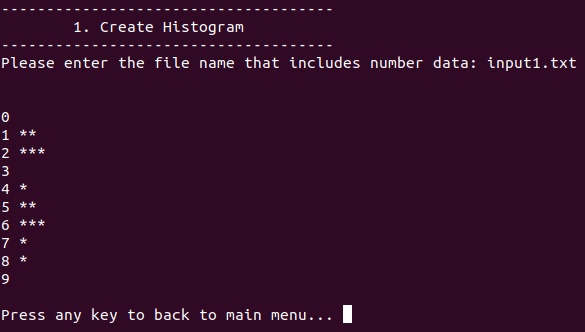
∙ If there is no file that obeys the wildcard, we simply give an error prompt to user

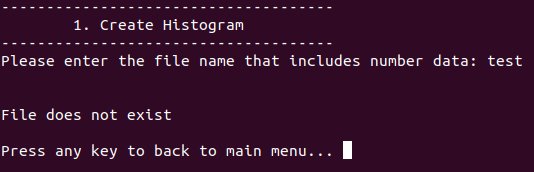
**Explanations About Implementations of Menu**

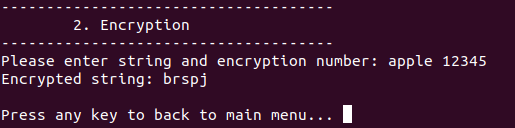
For the bonus part of the programming assignment #1, we created a menu that includes all questions above.

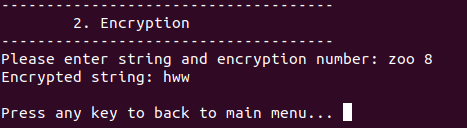
Here are some screenshots of our sample executions:

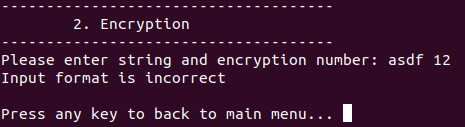


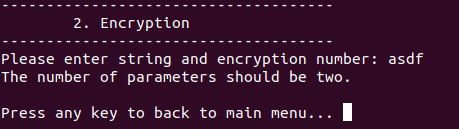


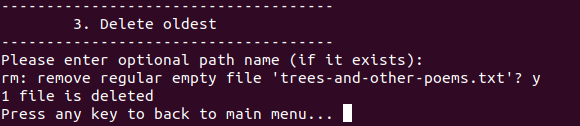


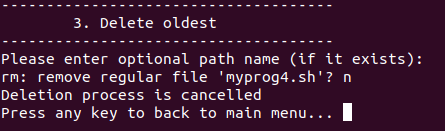


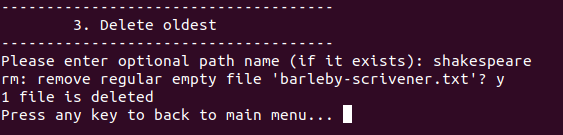


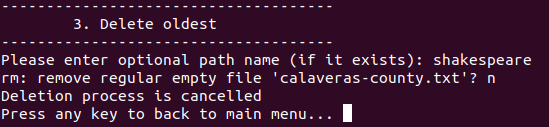


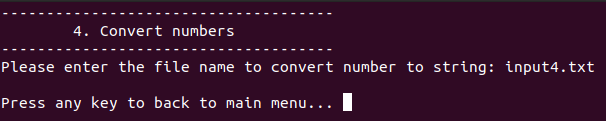


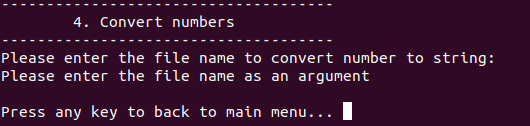


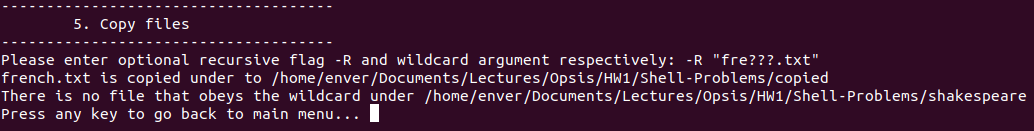


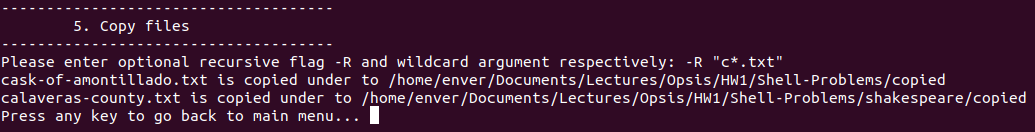


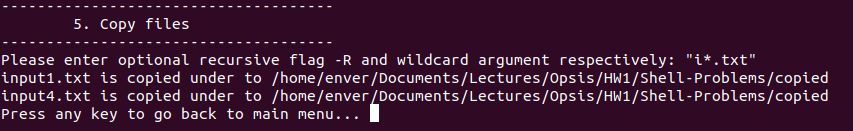












∙ As seen above, we’ve handled most of the corner cases such as file not found, deletion process and different wildcard cases.

∙ For every question, we’ve tried a lot of case to find out some minor and major bugs, the ones that we’ve caught are all fixed.

**Used Technologies**

* GitHub
* VS Code
* UNIX