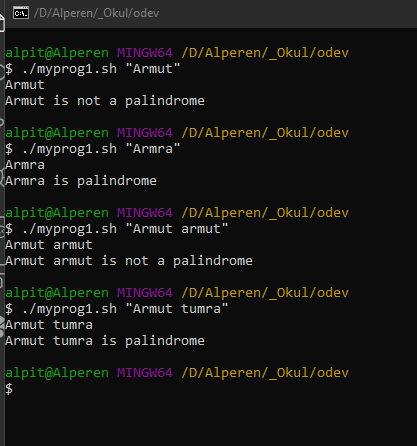
**Operating Systems Project 1 Report**

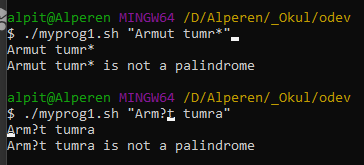
**Alperen Bayraktar 150116501**

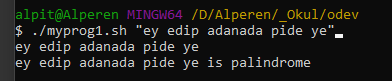
**Emre Erdem 150115501**

**Q1-**

The script first translates the input to uppercase because it must equalize “a” and “A”. Then it divides the input into half and checks the letters one from its start and end. We control if the letter is valid at 14th row. Here are few screenshots from code executions:





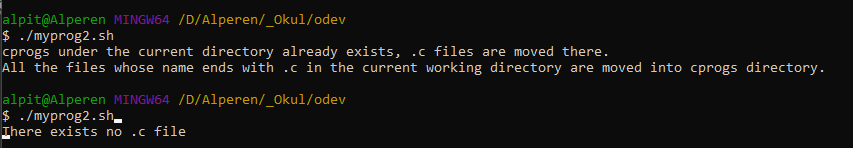


It also checks special characters.

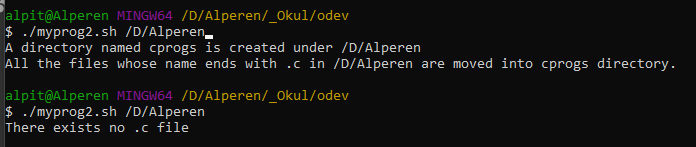
**Q2-**

Current directory is defined at the start of the code. If an input isn’t given, it performs the execution on the current directory. It runs the code only in the current directory without looking for inner directories.

X is a flag that controls if there exists any .c files. Then it checks if a directory argument was given, then does the requirements defined. Here are few screenshots from code executions:



The c files were moved back again then we try with directory argument.



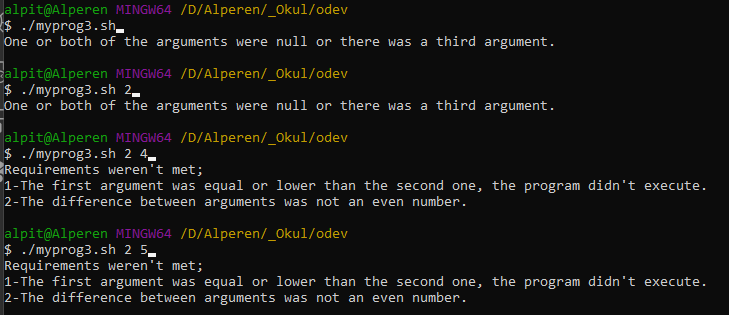
**Q3-**

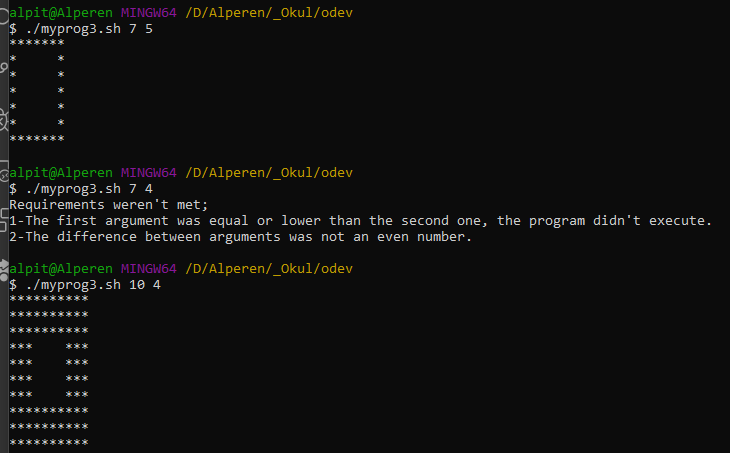
It checks the required conditions first;

* The first argument must be greater than the second.
* The difference between arguments must be even.

Then it creates the hollowed square in 3 parts.

* First part draws as many rows as the half value of the difference of arguments and then draws as many columns as the first argument.
* Second part draws as many columns as the half value of the difference of arguments and then draws as many blank spaces as the second argument then draws the half value again.
* Third part is the repetition of the first part.

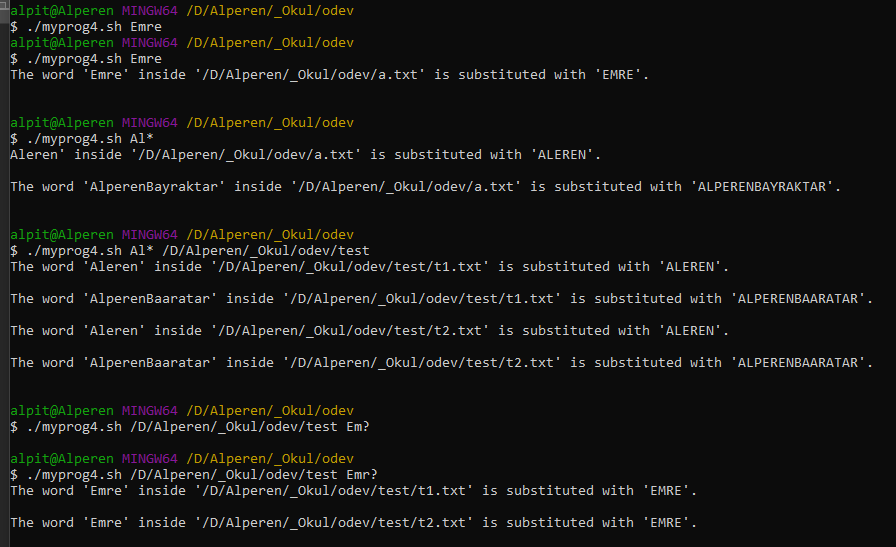




**Q4-**

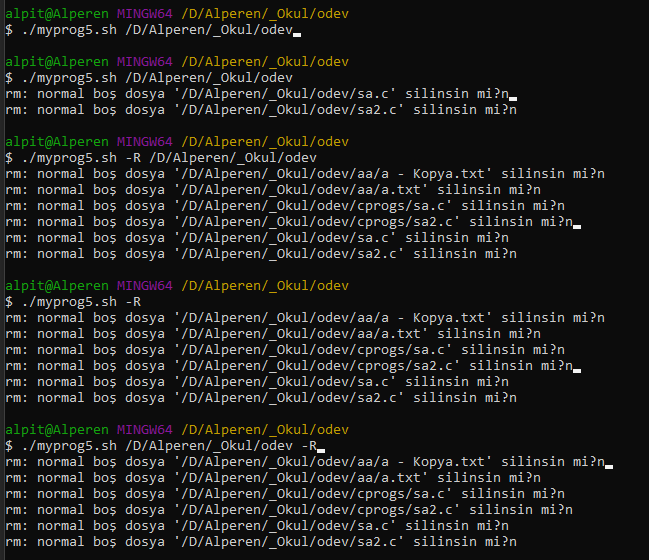
The big if-else structure is for handling all input probabilities. The explanation is given in code comments. The same structure is also implemented in fifth question. The function changeFunc() searches for files in the given directory and makes the necessary changes to each txt file.

Order of arguments are also handled, it works perfectly. Here are few screenshots from code executions:



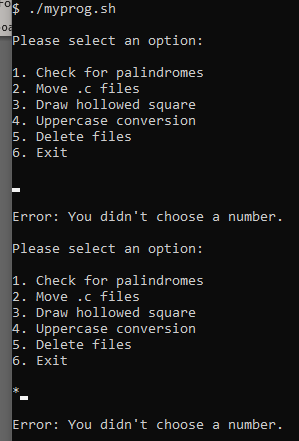
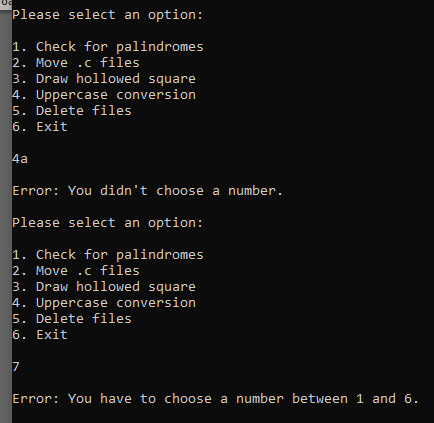
**Q5-**

We’ve defined an R flag at the start. If an argument is “-R” it works accordingly. The same huge if structure handles every possibility. The description can be found at comments. If recursive option is not selected, it only looks for the current directory. If it is selected, then checks inner directories. Here are few screenshots from code executions:

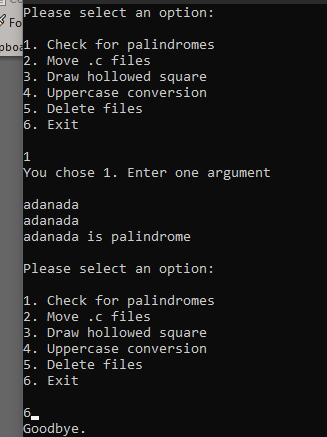


**MENU:**

Null input and special character input: Invalid input or number not in range:



It takes the user input and appends it to an array, then pass it as arguments to selected program. Program controls the validity of arguments (like “abc” as directory argument) inside the other script.

 It executes each selection accordingly.