***Due Date: 30/10/2018 – 23.59***

***OPERATING SYSTEM FIRST PROJECT REPORT***

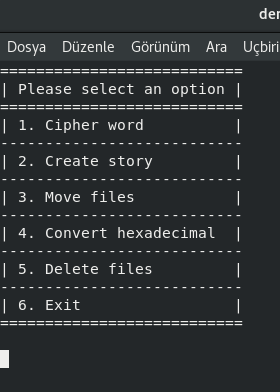
***Project members:***

*Muhammed Ali Doğan - 150115035*

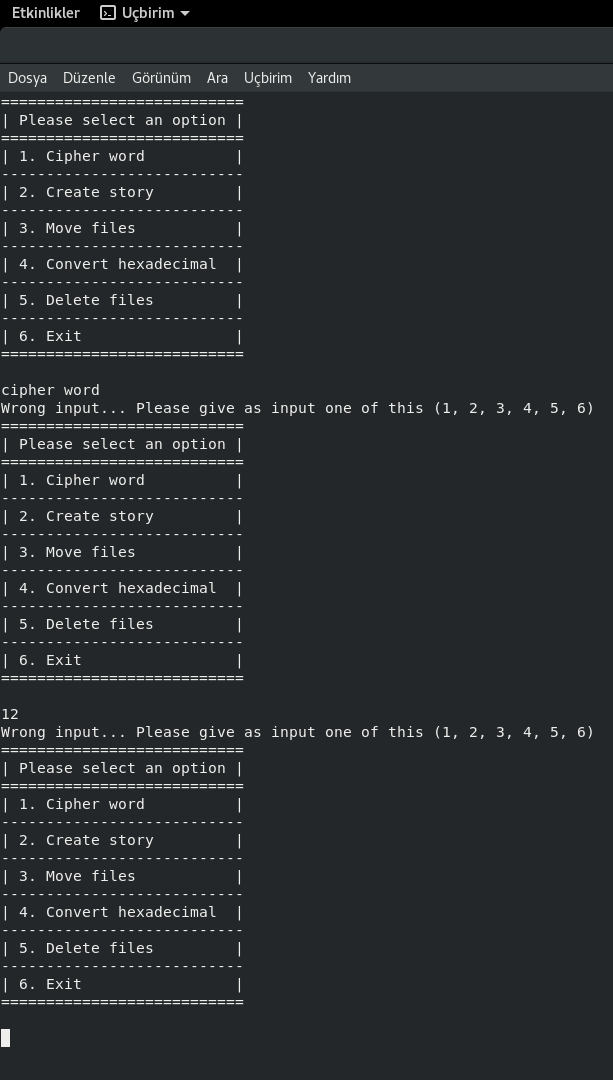
*Deniz Arda Gürzihin - 150116065*

In this project we are aiming to do several tasks. This tasks are about cipher World, creating story, moving files, converting hexadecimals and deleting files. Menu is optinal implementation but we also have done this cause it is bringing %10 extra credits.

First of all we start with implementation of Menu cause we thought that it could cause a little problem to do after we finish every program. So we start with this and we thought that it could be easier to use switch case statements to choose. Our menu and our options are in the program is as you see in the below;



Contains six options



1. Cipher word

2. Create story

3. Move files

4. Convert hexadecimal

5. Delete files

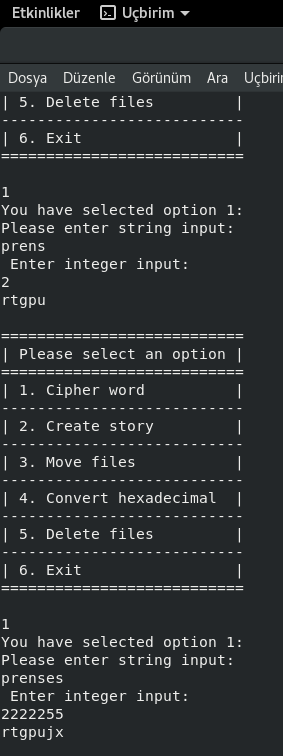
6. Exit

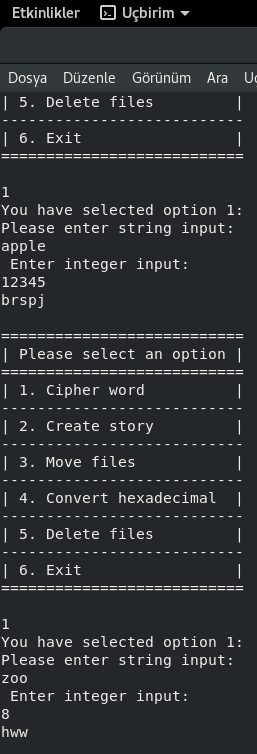
If the user supply 1, 2, 3, 4, 5, 6 as input, program is succesfully working but any other input cause program to ask user another input cause it is wrong for our code and we are clearing screen and printing menu again like this;

If we choose our input correctly then the program is starting;

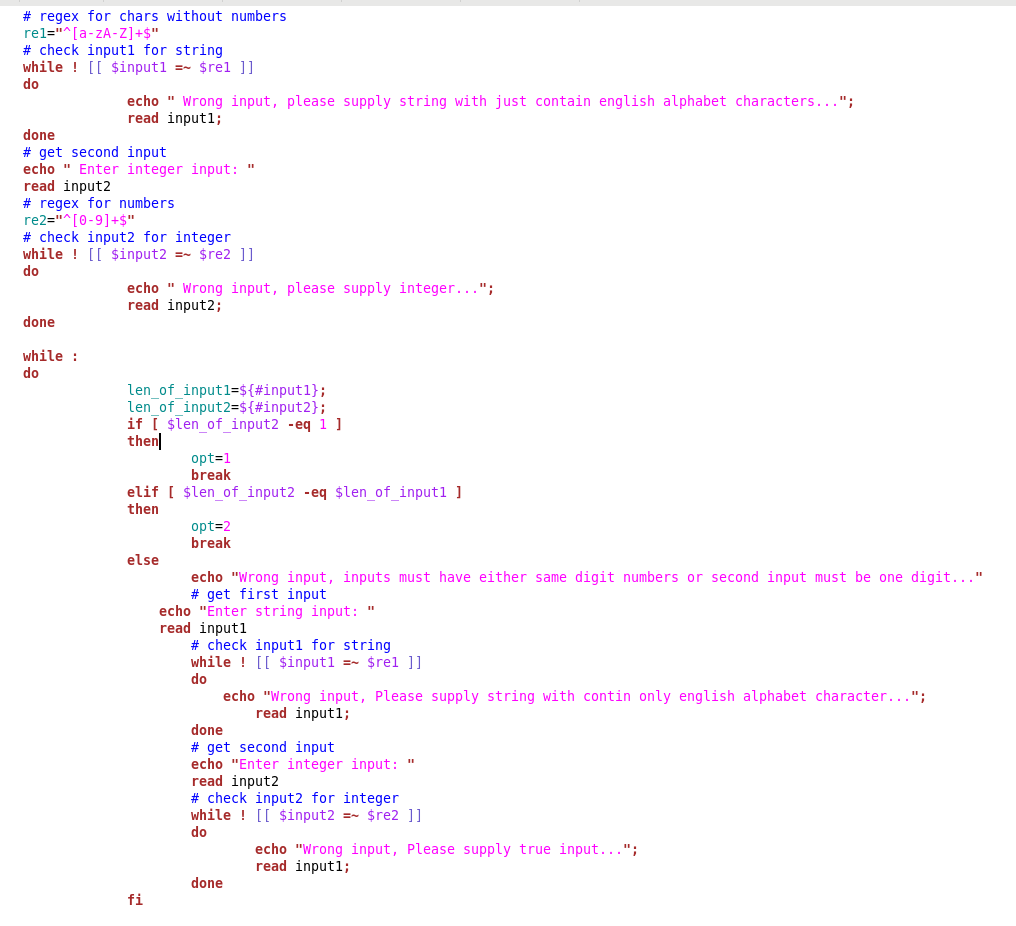
***1. Cipher word***

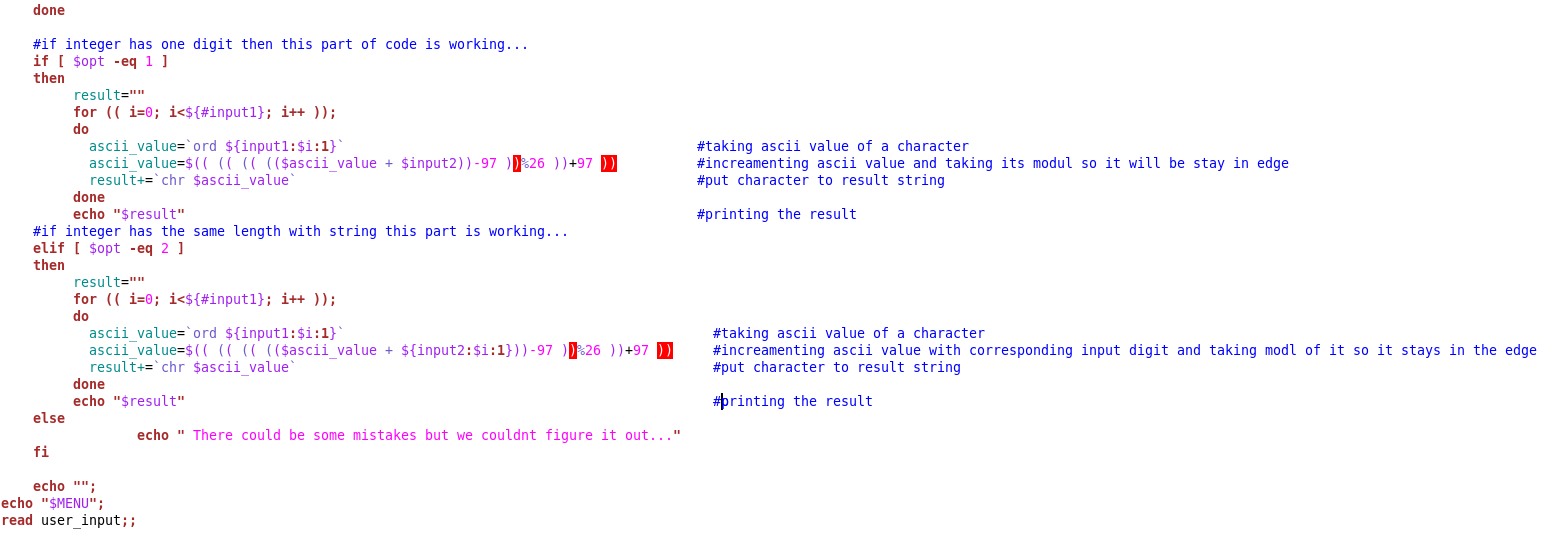
The cipher word function is about taking one string input and one integer input. And increasing the string with corresponding digit value of integer. We have two condition for this, first we can increment string all of its character if the integer input has one digit, second condition is string and integer must have the same length; if they have, we can increment string character by looking corresponding integer digit. Like in the project example, if we enter apple and 12345, we are expecting a become b cause we increment one time, p will turn r, second p become s, l will turn into p and the last e become j. Apple example and some examples, below;





Also we made input control with the code below;



Some of the code pieces about how we made this;

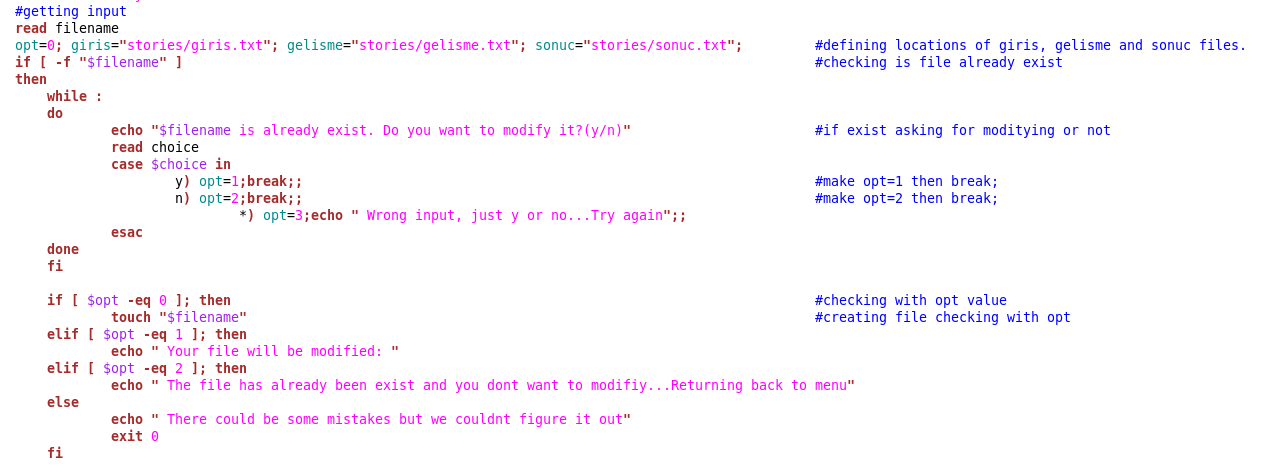
As you see, in the first condition while eq=1, that means our integer has one digit, we take string ascii code and increment it with the input integer number. And in the second condition, this will work when the string and the integer input have the same length that means eq=2. Like we did in the first condition, we will return to ascii value of the string and we will increment it with the corresponding digit of the integer number.

***2. Create Story***

###we made our code for assuming input stories: giris gelisme sonuc in a stories directory and we are taking inputs from there.

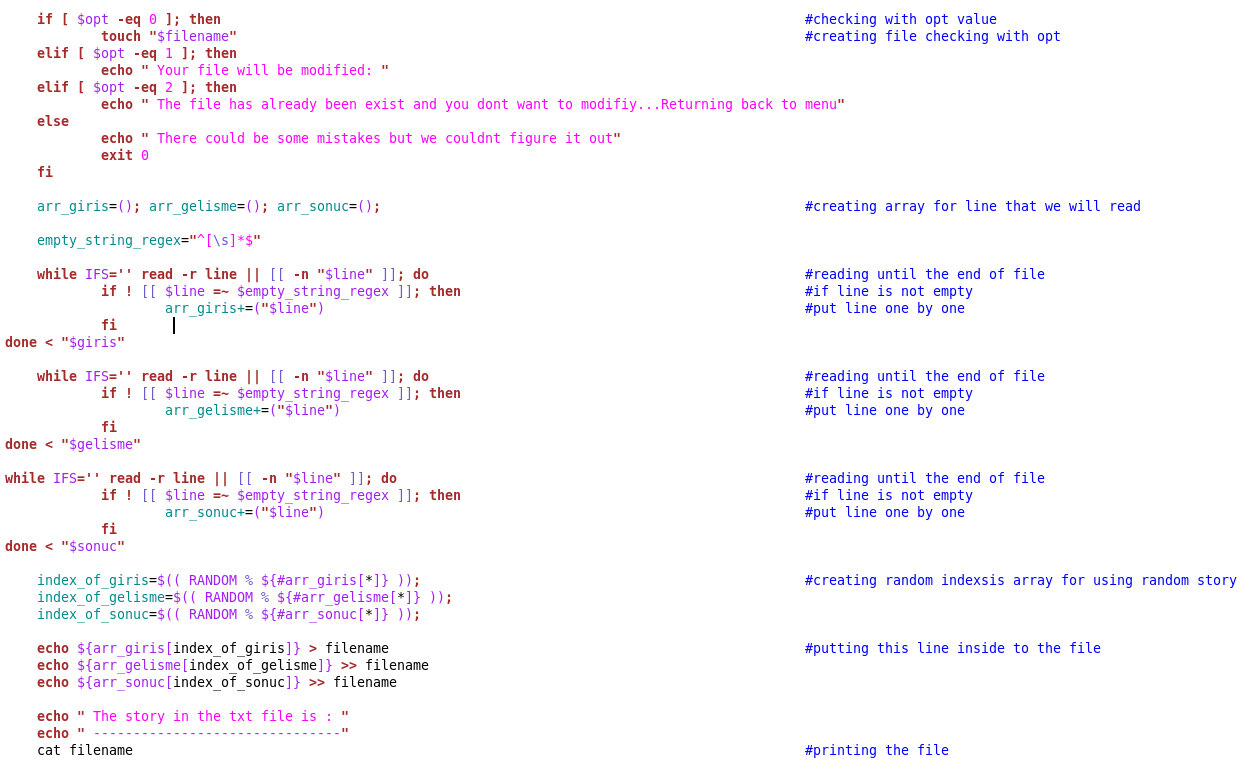
Create Story function is about creating random story. The program starts with asking user about input\_filename. After we are taking input, if file exist we are asking user for want to modify it. If yes we are modifying the text file with new story with taking just random one line from the supplied giris.txt, gelisme.txt,sonuc.txt and we are showing new created story. If no, we are showing whats inside this input\_filename file. And last option if user enters a name that not exist in the current directory, we are opening a new file named as input\_filename and doing the same step as if file exist condition. .

Some pieces from our code;

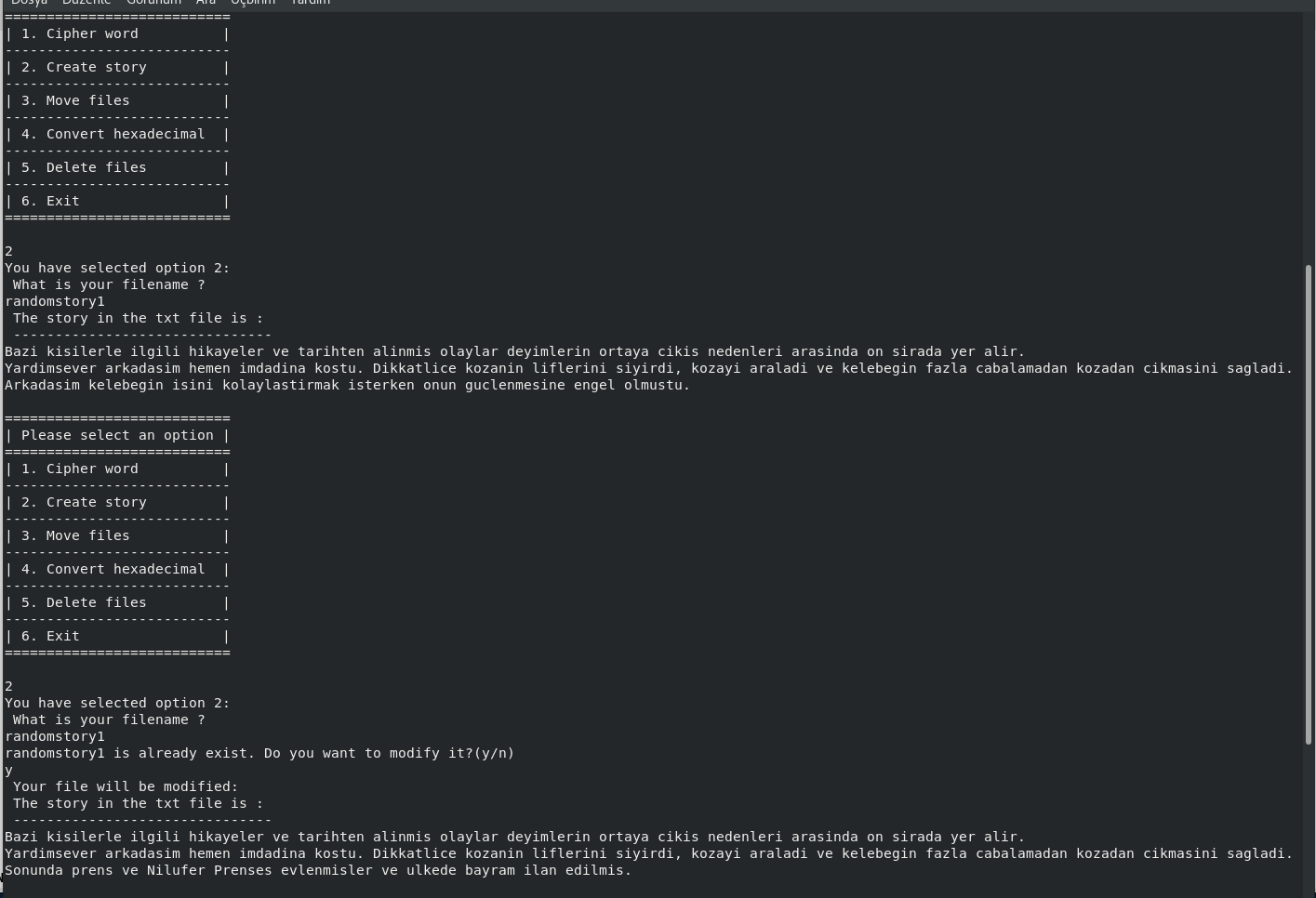


This is for error checking part

This is for reading file, creating new file and printing the file

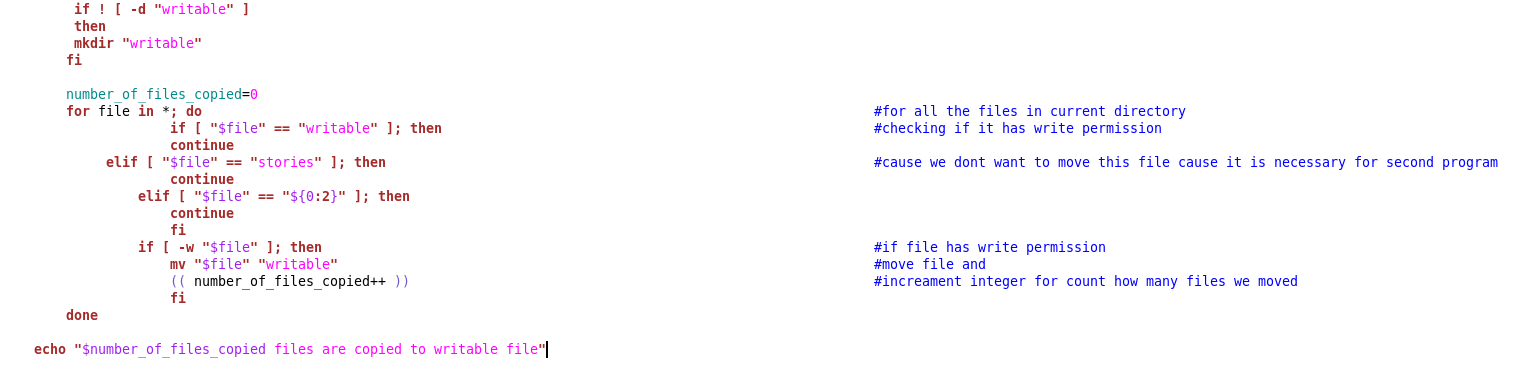


Some outputs from our program;



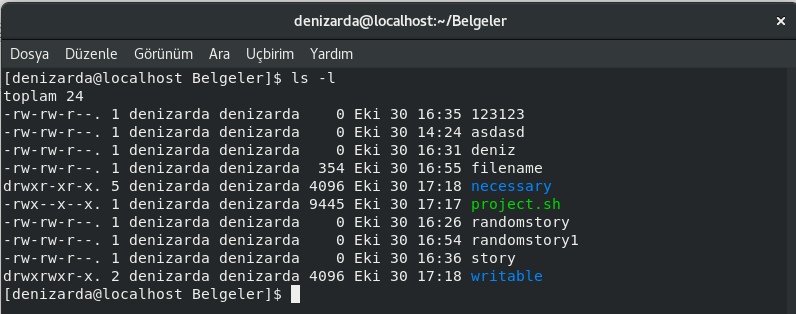
***3. Move Files***

Move files function is about moving files which has write permission for user and put them under a directory which name is writable and this directory will be under current directory. If directory is not exist, we are creting a new directory; if exist, we are moving files to writable directory. We deal with this problem with for loop, and our statement was for all files, we have checked if they have write permission it they have we moved if not we didnt.



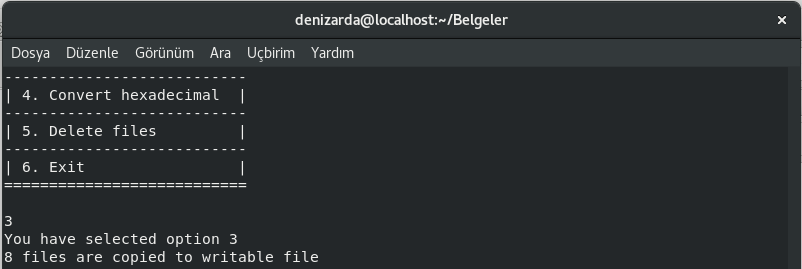
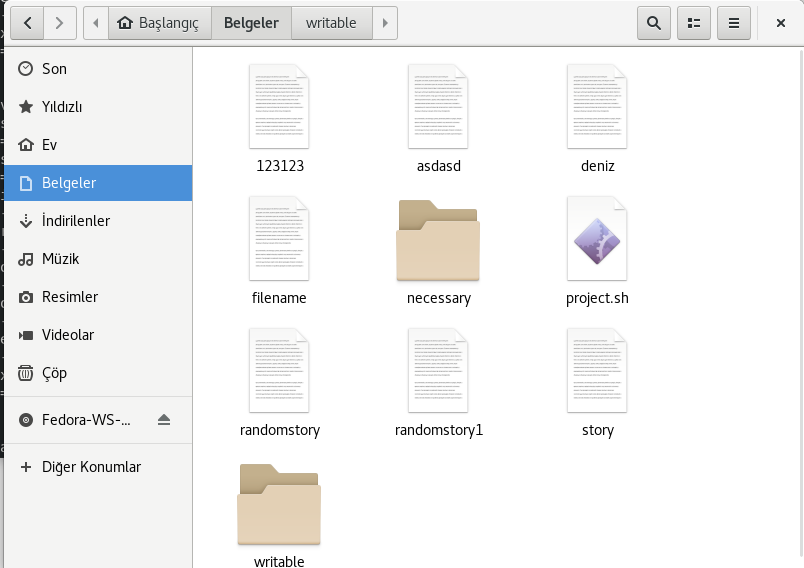
This is our code for moving files

How our code is working on terminal and checking if it is doing its job good;

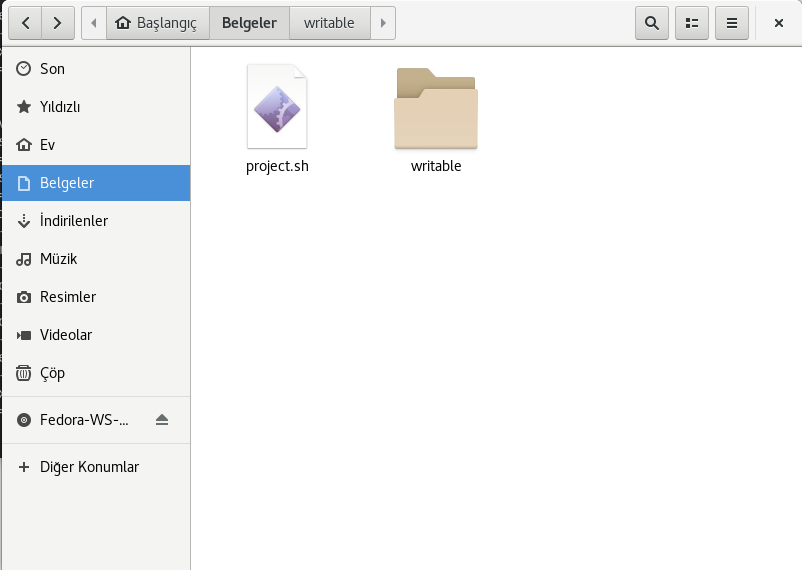


First, we checked the our current direction with ls –l command;

Secondly, we looked for the direction manually for to be %100 percent sure about mistakes



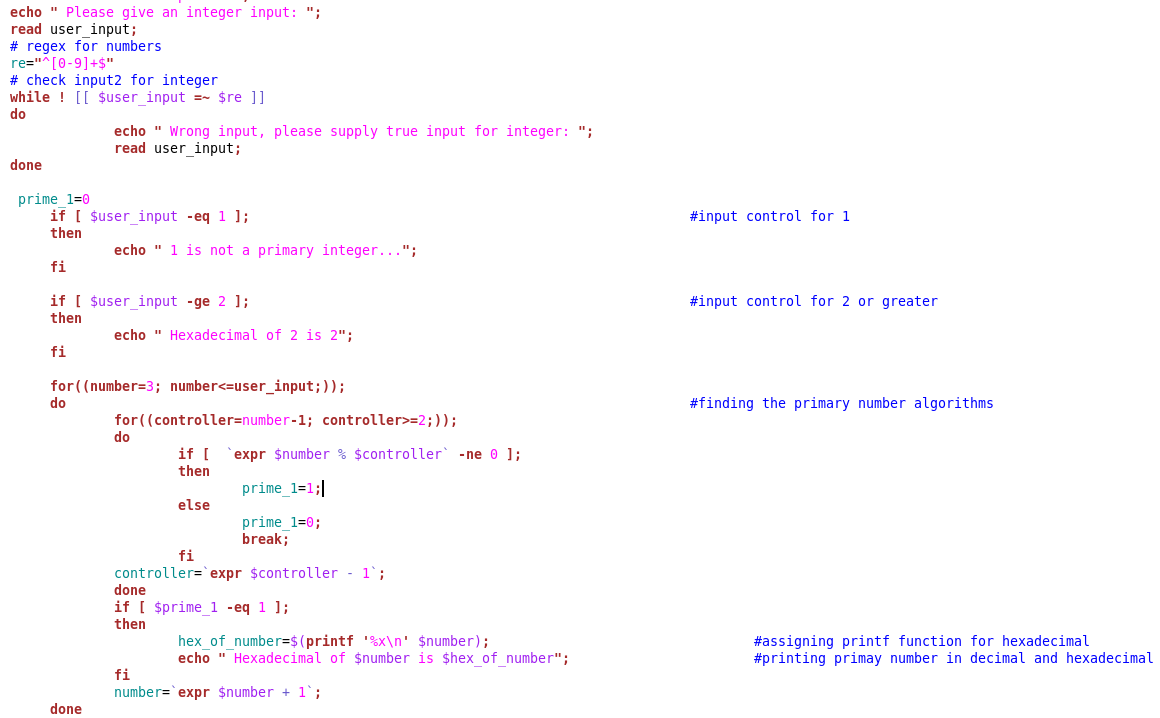
Then we created the process.



As you see, the process was succesfull and we moved every possible files with writing permission .

***4. Convert hexadecimal***

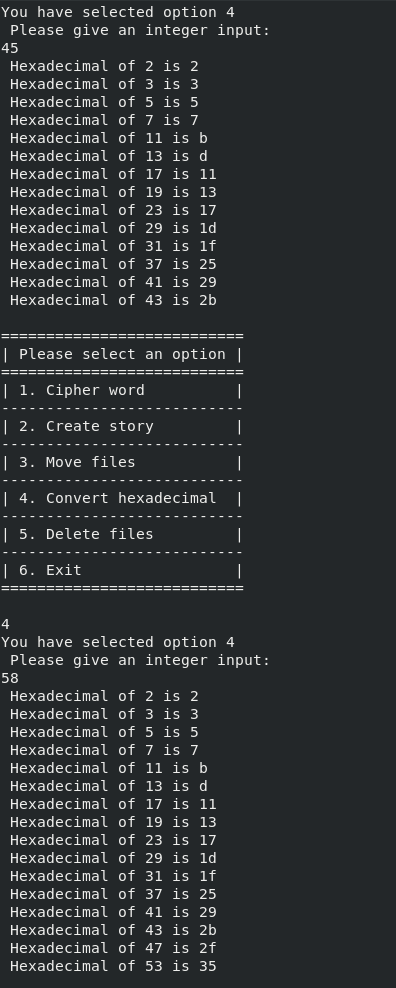
Convert hexadecimal function is about finding primary numbers less than supplied integer and converting them to hexadecimal numbers. First we take an integer input from the user and we check if the input is true or not. Then we use an algorithm to find primary numbers, when we find them we convert them into hexadecimal with printf function which we find in shell.



As seen above, this is our code for this process. And the best piece of code is above converting decimal to hexadecimal with this:

***Printf ‘%x\n’ $input***

In here input is the primary number while we are finding in the algorithms inside of two for.

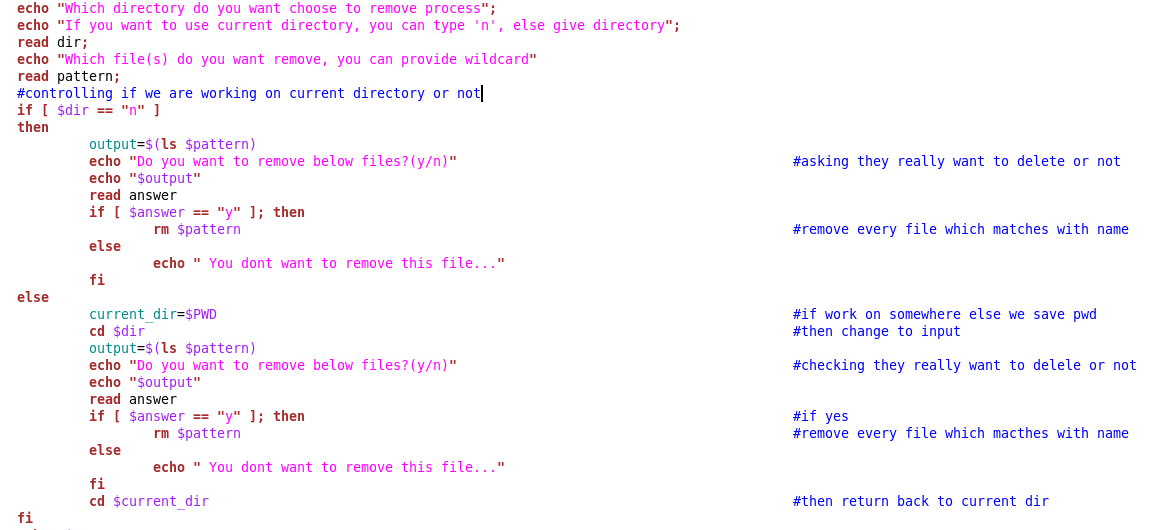


Sample output of our process

***5. Delete Files***

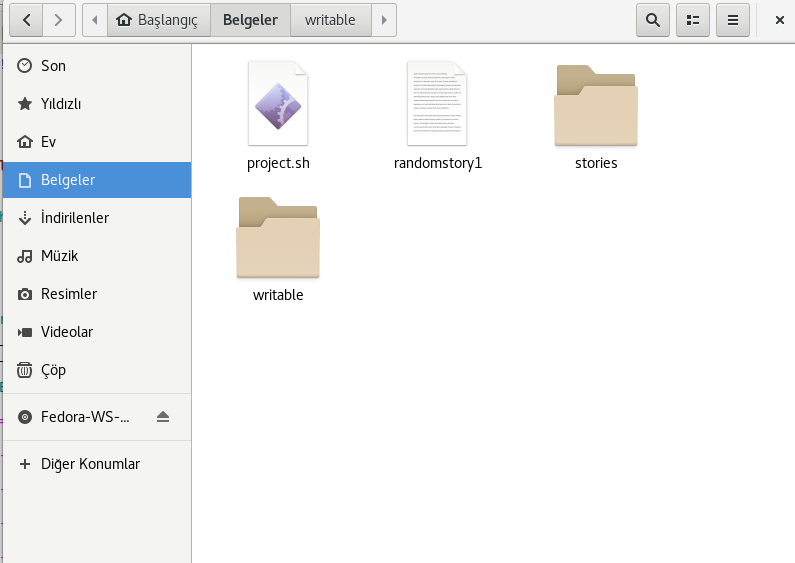
Delete files function is about deleting files with the given condition. This conditions are if the user supply just wildcard argument, then process has to find all files obey this argument and delete them. If user supply both wildcard argument and the pathname; then the process has to look for the supplied directory and make the delete operation in that directory.

We manage this with the code given below;

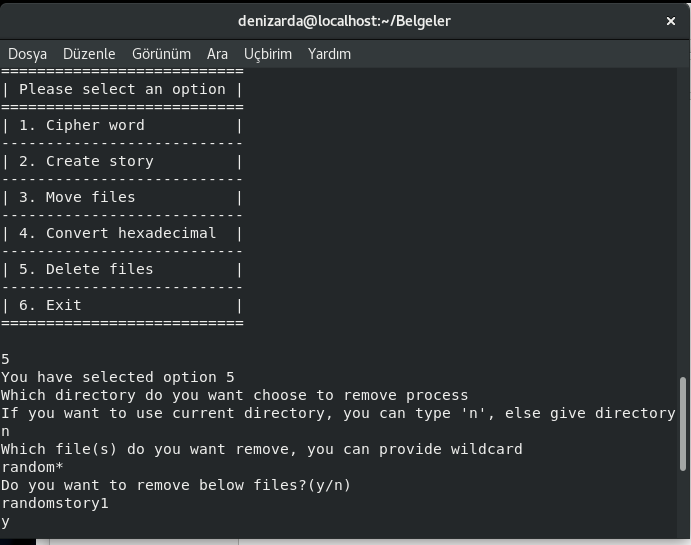


Firstly we are asking user about which directory does user want to work in if n press, then we are working on the current directory. If user supply a directory then we are changing our current to directory to supplied directory with cd command and we are creating command variable and saving our current directory for when we finish the delete process to return. Then we are asking user about wildcard argument, when user give this argument we are creating a pattern. Then we are finding how many files does user has and showing them. If user press ‘y’ and really want to delete it, we are removing the files with rm functions.

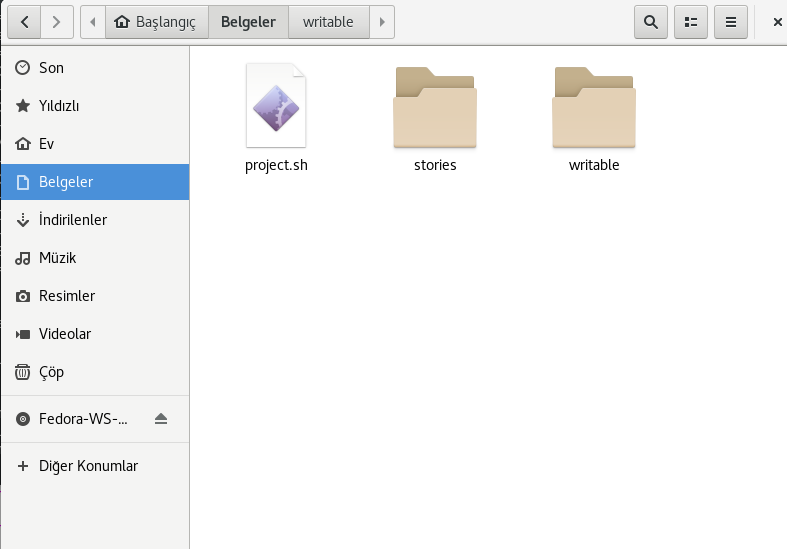
Sample example is given below;



Showing current directory



Executing process with directory as current and wilcard argument is random\*



Checking the file which supplied the properties, has been deleted or not, we see the process succesfull.

***6. Exit***

***echo “The program is succesfully exit.”***