COMP B10 – INTRO TO PROGRAMMING METHODOLOGIES USING PYTHON
ASSIGNMENT #6 (FILE ENCRYPTION)
(DICTIONARIES, FILES, EXCEPTIONS)

Create the Python code for a program that will allow users to encrypt or decrypt a file.

The operational logic for the program is as follows:

- The program will display a menu and get the user's selection. Notice the error message on the sample run. The program will terminate without error if the user enters 'Q" for their option.

- If the user selects to encrypt a file, the program will repeatedly prompt the user for the text file name to ENCRYPT until the user supplied file name can be opened or the user presses the <Enter> key alone to indicate they want to abort the program. The program will display an error message if the file does not exist. **The program will terminate normally if the user presses <Enter> without a file name.**

- If the user selects to decrypt a file, the program will repeatedly prompt the user for the text file name to DECRYPT until the user supplied file name can be opened or the user presses the <Enter> key alone to indicate they want to abort the program. The program will display an error message if the file does not exist. **The program will terminate normally if the user presses <Enter> without a file name.**

- If the input file can be opened successfully, the user will be prompted for an output file name. Assume that there will be no errors with the output file name.

- The program will then convert the input file using the dictionary provided in the TEMPLATE program. After this, the program will terminate normally.

Additional notes:
- You MUST use the provided program template and DICTIONARY for this program.
- Code your program WITHOUT using global variables
- You must effectively modularize your program into the functions provided in the template. You may modify the function definitions to pass arguments or return values, but the supplied named functions must be used. Here is what each function must accomplish:
  - main() – Control overall flow of the program.
  - displayMenuAndGetOption () – Displays the menu and returns the option selected by the user.
  - getFiles() – Prompts the user for the files to use and returns either the file handles (objects) or file names. Loops until the user either types in a valid input file name OR presses <Enter> for the input file name indicating they want to abort. Your routine may return a status flag indicating the success status of getting valid files or whether the user just pressed enter for the input file name.
  - convert() – Performs the encryption or decryption of the input file and saves the result to the output file. Accepts the input file and output file objects as parameters.

- **When doing your dictionary translation, if the character that you are translating DOES NOT exist in the dictionary, use the actual character for the substitution.**
- Your messages and prompts should look EXACTLY like those in the sample. Duplicating all blank lines and spacing.
- Pay attention to spelling and all other things visual -- you will be graded on this.
- Staple multi-page outputs.
- Use appropriate white-space and line-continuations in your source code.
- Upload ONLY your Python program to Canvas in the designated area.

SAMPLE RUNs:

```
File Encryption Program

E = Encrypt a file
D = Decrypt a file
Q = Quit the program

Enter menu selection (E, D, or Q): x

Error - Invalid option.

Run complete. Press the Enter key to exit.

File Encryption Program

E = Encrypt a file
D = Decrypt a file
Q = Quit the program

Enter menu selection (E, D, or Q): e

Enter the file to ENCRYPT. Press Enter alone to abort: filedoesnotexist.txt
Error - that file does not exist. Try again.

Enter the file to ENCRYPT. Press Enter alone to abort: filetoencrypt.txt
Enter the output file name: encryptedfile.txt

Run complete. Press the Enter key to exit.

File Encryption Program

E = Encrypt a file
D = Decrypt a file
Q = Quit the program

Enter menu selection (E, D, or Q): d

Enter the file to DECRYPT. Press Enter alone to abort: encryptedfile.txt
Enter the output file name: decryptedfile.txt

Run complete. Press the Enter key to exit.
```

Contents of filetoencrypt.txt

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ01234567890,./<>?;':"[]\{}|`~!@#$%^&*()_+-=
The quick brown fox jumped over the lazy dogs.
Did you know that the above sentence contains every letter in the alphabet?
```

Contents of encryptedFile.txt

```
0987654321zyxwvutsrqponmlk)(*&^%$#@!ZYXWVUTSRQPONMLKajihgfedcba:?/><.=",'{}\[]|`~JIHGFEDCBA_-+;
Q36 tp28z 9svnw 5vm 1pxu67 vo6s q36 y0kl 7v4r?
&27 lvp zwvn q30q q36 09vo6 r6wq6w86 8vwq02wr 6o6sl y6qq6s 2w q36 0yu3096q.
```

Contents of decryptedFile.txt

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ01234567890,./<>?;':"[]\{}|`~!@#$%^&*()_+-=
The quick brown fox jumped over the lazy dogs.
Did you know that the above sentence contains every letter in the alphabet?
```