Enigma Machine

In order to encode a message, required input is a txt file that includes the sentence that is needed to be encoded.

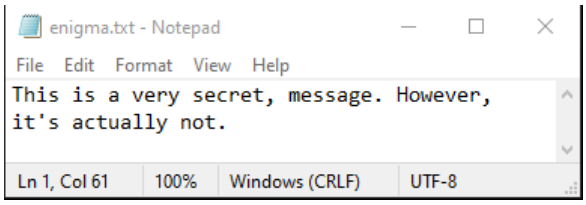

Figure 1: Example txt file

After creating the file, use the **encoder(file,word_dict)** function to create the coded message. **file** will have the name of the file that includes the message, and the **word_dict** will include the word library specified by the program but it can also be changed according to the needs of the user. This function creates a file called "encoded_file.txt", and writes the data in it. Here is an example of how the function is used.

```
encoder("enigma.txt",word_dict)
```

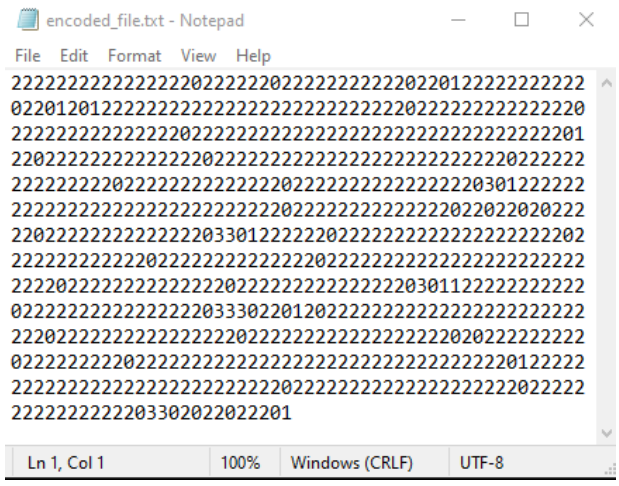

Figure 2: Example encoded file

In order to decode the encoded message, **decoder(file,word_dict)** function is used. This function reads the data from "encoded_file.txt" and writes the data into "decoded_file.txt". Here is an example of the function.

```
decoder("enigma.txt",word_dict)
```

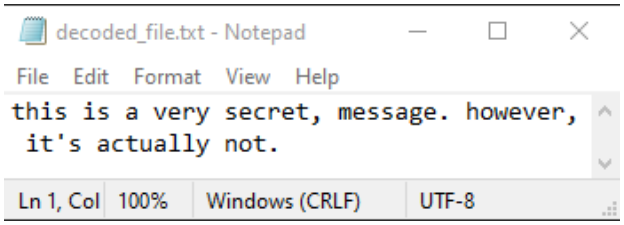

Figure 3: Example decoded file

As for the future of the project, there is only one word library at the moment. However, new ones can be added in order to have variety and make it harder to solve the enigma machine.

APPENDIX A: CODE

```
#DICTIONARY
word_dict = {"a":2,"s":22,"d":222,
             "f":2222,"g":22222,"h":222222,
             "j":2222222,"k":22222222,
             "l":222222222,"ş":2222222222,
             "i":22222222222,"q":222222222222,
             "w":2222222222222,"e":22222222222222,
             "r":222222222222222,"t":2222222222222222,
             "u":22222222222222222,"ı":222222222222222222,
             "o":2222222222222222222,"p":22222222222222222222,
             "ü":222222222222222222222,
             "z":2222222222222222222222,
             "x":22222222222222222222222,
             "c":222222222222222222222222,
             "v":2222222222222222222222222,
             "b":22222222222222222222222222,
             "n":222222222222222222222222222,
             "m":2222222222222222222222222222,
             "ö":22222222222222222222222222222,
             "ç":222222222222222222222222222222,
             "y":2222222222222222222222222222222,
             ",":3,".":33,"'":333}


#ENCODER
def encoder(file,word_dict):
    encoder_file = open(file,"r")
    file_write = open("encoded_file.txt","w")
    pre_file = encoder_file.read()
    word_list = pre_file.split(" ")
```

```
    sentence = ""
    for word in word_list:
        for char in word:
            sentence+=str(word_dict[char.lower()])
            sentence+="0"
        sentence+="1"
    file_write.write(sentence)
    return sentence

#DECODER
def decoder(file,word_dict):
    decoder_file = open(file,"r")
    decoder_file_write = open("decoded_file.txt","w")
    decoder_pre_file = decoder_file.read()
    decoder_list = decoder_pre_file.split("1")
    decoded_sentence = ""
    for word in decoder_list:
        char_list = word.split("0")
        for char in char_list:
            for i in word_dict:
                if str(char) == str(word_dict[i]):
                    decoded_sentence+=i
        decoded_sentence+=" "
    decoder_file_write.write(decoded_sentence)
    return decoded_sentence

print(encoder("enigma.txt",word_dict))
print(decoder("encoded_file.txt",word_dict))
```