



Faculty of Engineering & Technology
Department of Electrical & Computer Engineering
ENCS4320 - Applied Cryptography
Second Semester, 2023/2024

Homework # 2 (Programming Assignment)

Name : Hala Mohammed
ID : 1210312
Sec : 1

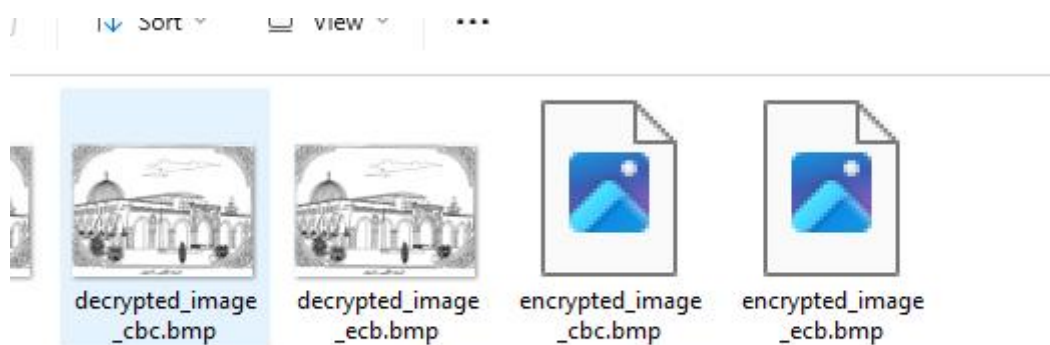
Copy of your diagram:

welcome message

```
C:\Users\halaa\Desktop\CryptoProj\CryptoProj\.venv\Scripts\python.exe C:\Users\halaa\Desktop
✧ Welcome to the TEA Encryption and Decryption Program! ✧
This program is brought to you by Eng. Hala.
Get ready to dive into the world of cryptography!
Let's start by processing your image file. 🚀
Please enter the path to the image file: "C:\Users\halaa\Desktop\CryptoProj\Aqsa.bmp"
You have entered: C:\Users\halaa\Desktop\CryptoProj\Aqsa.bmp
Please enter the 128-bit key : 0x0F1E2D3C 0x4B5A6978 0x86754321 0xCDEF0123
Please enter the 64-bit IV : 0x12345678 0x9ABCDEF0
Encrypting image using TEA-ECB mode...
Decrypting image using TEA-ECB mode...
Encrypting image using TEA-CBC mode...
Decrypting image using TEA-CBC mode...
Encryption and decryption successful. Files saved as:
    Encrypted ECB: 'C:\Users\halaa\Desktop\CryptoProj\encrypted_image_ecb.bmp'
    Decrypted ECB: 'C:\Users\halaa\Desktop\CryptoProj\decrypted_image_ecb.bmp'
    Encrypted CBC: 'C:\Users\halaa\Desktop\CryptoProj\encrypted_image_cbc.bmp'
    Decrypted CBC: 'C:\Users\halaa\Desktop\CryptoProj\decrypted_image_cbc.bmp'

Process finished with exit code 0
|
```

Images in file



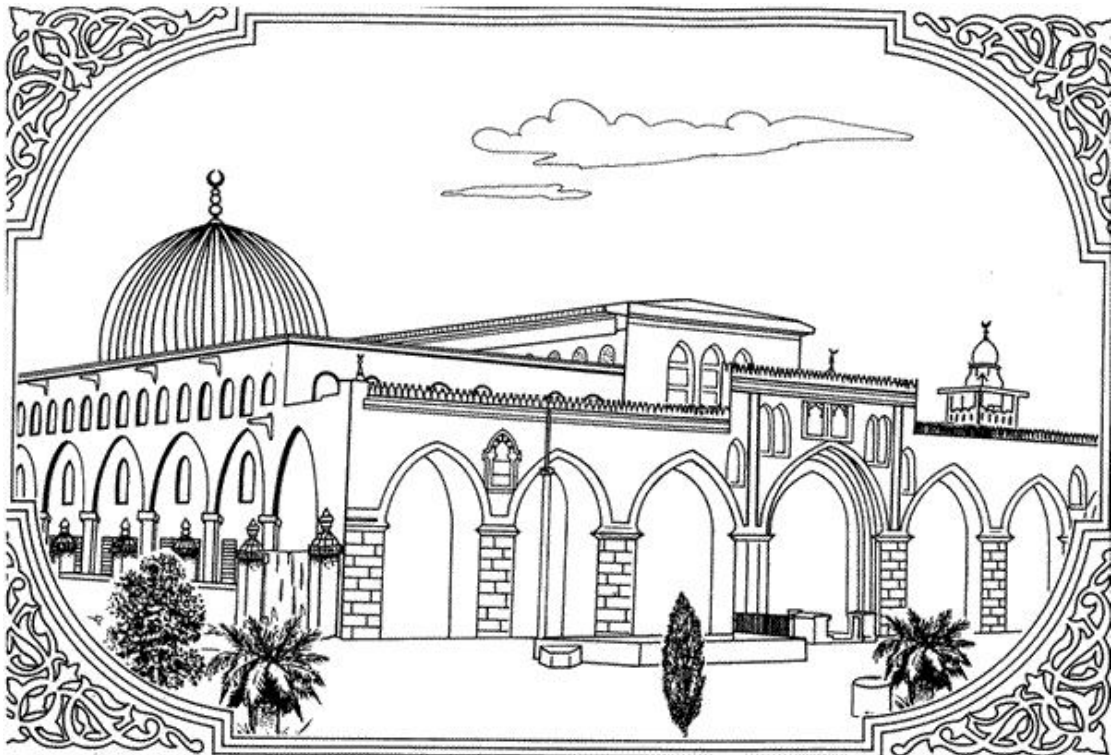
TEA-ECB Ecncrypt



encrypted_image_ecb.bmp

Sorry, Photos can't open this file because the format is currently unsupported, or the file is corrupted

TEA-ECB Decrypt



المسجد الأقصى - فلسطين.

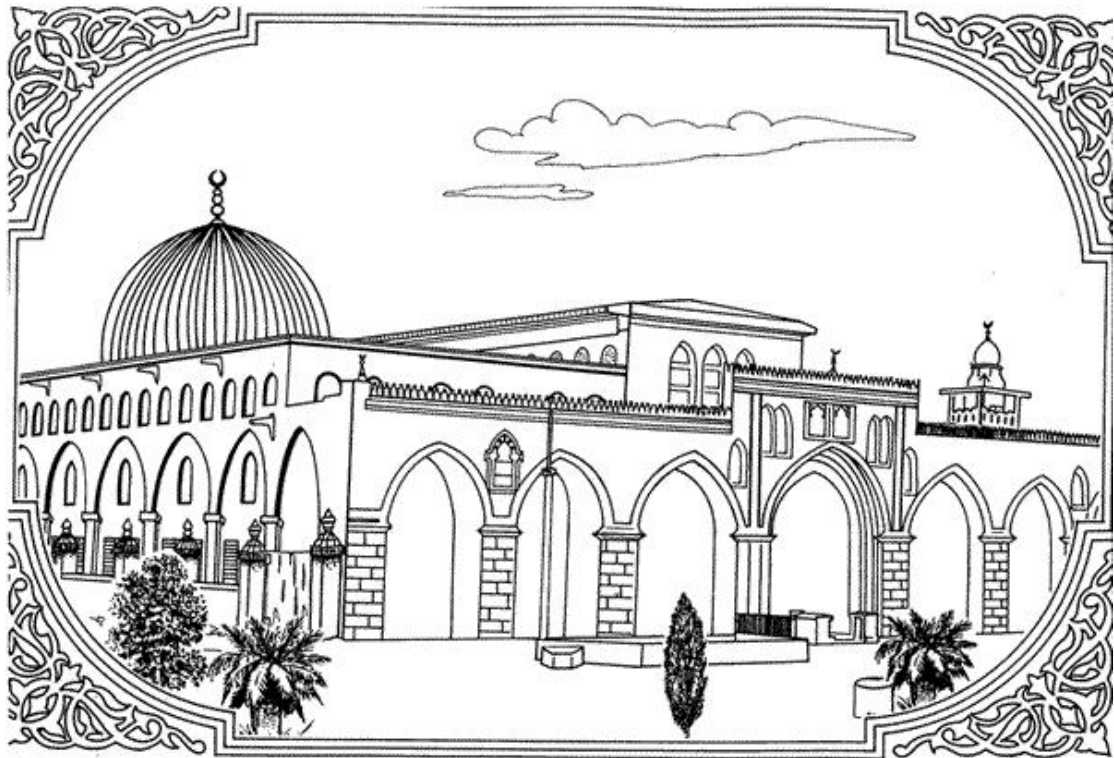
TEA-CBC Ecncrypt



encrypted_image_cbc.bmp

Sorry, Photos can't open this file because the format is currently unsupported, or the file is corrupted

TEA-CBC Decrypt



المسجد الأقصى - فلسطين .

I would want to notify you that I am having trouble accessing the encrypted images. I've tried so many times to open them, but I've never been successful. I've gone over the code in exhaustive detail, even re-do the padding function, and I've tried writing it in both C and C++. Still, I'm facing the same problem. strangely, I've seen that both methods successfully complete the decryption procedure.