REverse Engineering Challenge (450 pts)

Your task is to reverse engineer the *decryption* algorithm in DTool.exe and then write a program that will correctly *encrypt* an arbitrary file with an arbitrary password such that the resulting encrypted file can be decrypted by DTool.exe without *ANY* modification to DTool.exe. When reversing DTool.exe, the main function begins at address 0x00434F40.

When you execute DTool.exe, it seeks a file named "encrypted.txt" in the current directory and produces the unencrypted file named "decrypted.txt." (You may also specify a file name in place of "encrypted.txt").

There are 4 files in this package:

1. **This set of directions.**
2. **The decryption tool, DTool.exe.**
3. **encrypted.txt**
4. **IfYouLaugh.jpg**

When you run the following command in a DOS command window,

```
C:\DTool.exe encrypted.txt
```

the file "decrypted.txt" is generated and is the decrypted file. It is a jpeg file, so change the extension to .jpg to view it.

Your encryption tool should take a file name and a password to generate an encrypted file that DTool.exe can decrypt. (Yes, a password was used for "encrypted.txt")

```
C:\Your_ETool.exe <filename> <password>
```

In order to submit a simple answer for this challenge, we have included the unencrypted file"IfYouLaugh.jpg." Use the following command with your encryption tool:

```
C:\ Your_ETool.exe IfYouLaugh.jpg password
```

Then submit the 8 hexadecimal digits at file offset 0x2100. You can open the encrypted file in a hex editor and copy the first 8 bytes at 0x2100. Example submission:

```
18 29 3A 4B F9 E8 D7 B6
```