HW 7

- Due Apr 1 by 5pm
- Points 100
- Submitting a file upload
- File Types docx, pdf, asm, and lst
- Available Mar 26 at 8am - Apr 3 at 5pm

This assignment was locked Apr 3 at 5pm.

HW # 7: Theme: Conditionals, Booleans, Loops

(All main questions carry equal weight. Credit awarded to only those answers for which work has been shown.)

1. Draft a program that scans a string to determine whether a character in the register AL is present within the string. If found, the program should print “character found”, its value and its index within the string. **Submit the asm/list file and screenshots that show the output of your code for the following example string:**

    MyString BYTE "Spring is pleasant in Auburn" 0x0

    Try your program for various characters present in the string and a few characters not present in the string. Submit screen shots of these trials. Provide comments on each line.

    If AL = 'g, the index is 6.

    if AL - 'l', the index is 12

2. Write a program which encodes any string using the XOR instruction. Test it using your <last name> in the data segment to produce cipher text and then decode using the program to get plain text. Use the last two or three digits of your student id as the key. Print plane text from the data segment, print the cipher text, and then print the plain text upon execution. **Submit the asm/list file and screenshots that shows the run and output of your code.**

Write a narrative telling how the program is working. What are the strengths and weaknesses of this encryption method (30**% of points, Typewritten answer required**)?

3. Draw the stack (use any handwriting/word/pdf) before every instruction that is marked red is executed to show your understanding of the CALL and RETURN functions. Use N/A to represent unpredictable values.

```
Main Proc
4040018          mov ecx, 0AEFCh
404001C          mov ebx, 5612h
4040020          call FDIV
4040026          mov eax, ebx
          …
          …

Main EndP

FDIV  PROC
4041040          Push ebx
4041044          Push ecx
4041048          mov eax, edx
          …
          …
404A060         Pop ecx
404A062         Pop ebx
404A064         ret
FDIV EndP
```

(https://auburn.instructure.com/courses/1645612/files/256838420/download?wrap=1)