MIT Tutorial 1

Name: Krunal Rank
Adm. No: U18CO081

1.WALP to sort the characters of your roll no string (e.g. U18CO001) in ascending order (00118COU) in-place. Use ASCII values for comparison.

```
;<Question 1>

jmp start

;data
rollno: db 85, 49, 56, 67, 79, 48, 56, 49
bucket: ds 128

;code
start:    lxi h,rollno
count:    mov a,m
     cmp b
     jz sort
     xchg
     lxi h,bucket
     mov c,a
     dad b
     inr m
     xchg
     inx h
     jmp count
sort:     adi 1h
     mvi b,0080h
     mvi c,0000h
     lxi d,rollno
     lxi h,bucket
     push h
loop:     pop h
     mov a,m
     inx h
     push h
     xchg
write:    dcr a
     jm writeE
     mov m,c
     inx h
```

```
     jmp write
writeE: adi 1h
     xchg
     dcr b
     inr c
     mvi a,0000h
     cmp b
     jz end
     jmp loop
end: hlt
```

2. What modifications will you do in (1.) to sort it in descending order?

To sort it in descending order, following changes must be done:

- Change immediate value loaded in C in line 25 to 0x007f
- Instead of loading bucket address in HL pair in line 27, load address(bucket) + C
- Change inx H in line 31 to dcr H
- Change inr C in line 42 to dcr C

3. How will you perform (1.) without changing the original data (not in-place) (Hint: use ds directive to reserve storage for the sorted output)?

A similar process has been done in (1.) where input is taken using db and bucket is created using ds. Similarly, an output storage location can be created using ds and in line 26, instead of loading rollno, we can load the output base address in DE pair.

4. What would be the result in (1.) if the roll no string input was first converted to lowercase?

If the roll number input string is first converted to lowercase, then no effect would have been seen on the output. The only effect that will be seen will be the last alphabetical characters will be lowercase.
For example, the output for U18CO081 is 01188COU.
Then the output for u18co081 is 01188cou.
This is because digits have ascii values from 48-57, uppercase alphabets have ascii values from 65-90 and lowercase alphabets have ascii values from 97-122.

5. The management decided to change the prefix from U to UG and department code from CO to COED. WALP to convert your old roll no. (U18CO001) to the new format (UG18COED001) (Hint: use ds directive to reserve extra bytes, to avoid overwriting code segment).

```
;<Question 4>

jmp start

;data
rollno: db 85, 49, 56, 67, 79, 48, 56, 49
modifiedrollno: ds 11

;code
start:    lxi d,modifiedrollno
     lxi h,rollno
     mov a,m
     xchg
     mov m,a ;U
     inx h
     mvi m,47h ; G
     inx h
     inx d
     xchg
     mov a,m
     xchg
     mov m,a ;1
     inx h
     inx d
     xchg
     mov a,m
     xchg
     mov m,a ;8
     inx h
     inx d
     xchg
     mov a,m
     xchg
     mov m,a ;C
     inx h
     inx d
     xchg
```

```
     mov a,m
     xchg
     mov m,a ;O
     inx h
     mvi m,45h ;E
     inx h
     mvi m,44h ;D
     inx h
     inx d
     xchg
     mov a,m
     xchg
     mov m,a ;0
     inx h
     inx d
     xchg
     mov a,m
     xchg
     mov m,a ;0
     inx h
     inx d
     xchg
     mov a,m
     xchg
     mov m,a ;1
hlt
```

6. In order to simplify a few tasks, sometimes, it is sufficient to work with just your serial no. instead of the entire roll no. string. WALP that does this conversion and stores the output in the accumulator. (U18CO010 -> A, U18CO016 -> 10, etc.).

```
;<Question 6>

jmp start

;data
rollno: db 85, 49, 56, 67, 79, 48, 56, 49

;code
start:    lxi h,rollno
     lxi d,05h
     dad d
```

```
     mov a,m
     sbi 30h
     mov b,a
     mvi a,00h
     mvi d,0ah
loop:     dcr d
     jm loopE
     add b
     jmp loop
loopE:    mov c,a
     inx h
     mov a,m
     sbi 30h
     add c
     mov b,a
     mvi a,00h
     mvi d,0ah
loop1:    dcr d
     jm loopE1
     add b
     jmp loop1
loopE1: mov c,a
     inx h
     mov a,m
     sbi 30h
     add c
     hlt
```

7. Your program (6.) will fail for which roll nos? Modify it, so that it doesn't fail for the cases you've mentioned. Use a register pair for the output instead of the accumulator.

The above code fails for roll numbers greater than 255. It can be fixed by implementing storage of roll numbers in register pairs instead of the accumulator.