Solve the following questions.

1. Implement the following pseudo-code in assembly language. Also, give the corresponding data definition directives:

[**04 points**]

```
; All values are unsigned integers
do {
    if (A % 13 == A)
        A++;
        B++;
    else
        A--;
        B = A / 10;
    }
    while(B < C)
```

SOLUTION:

```
MOV         AX, A
MOV         DX, 0

L1:         MOV   CL, 13
DIV         CL          ;DL = A%13
MOVZX       AX, DL      ;AX = A%13

CMP         AX, A       ;IF A%13 == A
JE          L2          ;THEN JUMP TO INCREMENTS

DEC         A           ;ELSE
MOV         AX, A
MOV         BL, 10
DIV         BL          ;AL = A/10
MOVZX       B, AL       ;B = A/10
JMP         L3

L2:         INC   A
            INC   B

L3:         MOV   CX, C ;while
CMP         B, CX
JB          L1
RET
```

2. Provide the contents of registers/flags where indicated (in hex-decimal), after execution of the following instructions. **[2 Points]**

```
mov      al, 1Ah
add      al, E5h
adc      al, 3              ; AL = 02h

mov      al, 13h
test     al, 74h           ; AL = 13h

mov      al, 9Bh
and      al, 80h           ; AL = 80h

mov      al, AAh
stc
sbb      al,7              ; AL = 0A2h
```

3. Elaborate the difference between ROR and RCR instructions through some working example. **[2 Points]**

ANSWER: **ROR Instruction** shifts each bit to the right and copies the lowest bit into the Carry flag and the highest bit position (MSB). Whereas **RCR** instruction shifts each bit to the right, copies the Carry flag into the MSB, and copies the LSB into the Carry flag.

```
E.g.   mov  al,01h            ; AL = 00000001b
       ror  al,1              ; AL = 10000000b, CF = 1

       mov  al,01h            ; AL = 00000001b, CF =0
       rcr  al, 1             ; AL = 00000000b, CF = 1
```