

**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 == B)
        A++;
        B++;
    else
        A = A2;
        B = A * 10;
}
while(B < C)
```

SOLUTION:

```
MOV     AX, A
MOV     DX, 0

L1:     MOV     CL, 13
DIV      CL           ;AL = A/13
MOVZX   AX, AL       ;AX = A/13

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

MOV      AX, A        ;ELSE
MUL      AX           ;AX = A2
MOV      A, AX        ;A = A2
MOV      BX, 10
MUL      BX           ; AX = A * 10
MOV      B, AX
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]**

SOLUTION:

```
mov     al, 1Ah
test   al, 3           ; AL = 1Ah

mov     al, 13h
xor    al, 74h         ; AL = 67h

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

mov     al, 2h
sub    al, 3h
adc    al, 1           ; AL = 01h
```

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

ANSWER: The **ROL** (rotate left) instruction shifts each bit to the left. The highest bit is copied into the Carry flag and the lowest bit position. Whereas **ROR Instruction** shifts each bit to the right and copies the lowest bit into the Carry flag and the highest bit position (MSB).

e.g.

```
mov     al, 11000001b
rol    al, 1           ; AL = 10000011b, CF = 1
```

```
mov     al, 11000001b
ror    al, 1           ; AL = 1110 0000b, CF = 1
```