System Programming Assignment-1

1. Write and test a MASM program to Display your name and program title on the output screen.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code
.STACK 100H     ; Bytes of stack spaces for the Program


.DATA

MYNAME DB "NAME : RITABROTO GANGULY", "$"
PROGTITLE DB 0DH, 0AH, "PROGRAM TITLE : PRINT NAME & TITLE$"


.CODE

MAIN PROC

    MOV AX, @DATA
    MOV DS, AX
    MOV DX, OFFSET MYNAME
    MOV AH, 09H
    INT 21H

    MOV DX, OFFSET PROGTITLE
    INT 21H


    ; TERMINATE

    MOV AH, 4CH
    INT 21H

MAIN ENDP
END MAIN


; MASM Q1.ASM && LINK Q1.OBJ && Q1.EXE
```

```
C:\ASS1>Q1.EXE
NAME : RITABROTO GANGULY
PROGRAM TITLE : PRINT NAME & TITLE
```

2. Write and test a MASM program to convert a letter from uppercase to lowercase.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code

.STACK 100H     ; Bytes of stack spaces for the Program



.DATA

UPPER DB "ENTER A LETTER IN UPPERCASE : ", "$"

LOWER DB 0DH, 0AH, "SAME LETTER IN LOWERCASE : "

CHAR DB ?, "$"



.CODE

MAIN PROC

    MOV AX, @DATA

    MOV DS, AX

    MOV DX, OFFSET UPPER

    MOV AH, 09H

    INT 21H



    ; ENTER THE LETTER
```

```
    MOV AH, 01H
    INT 21H          ; TAKE SINGLE CHARACTER INPUT IN AL

    ADD AL, 20H      ; CONVERT TO LOWERCASE BY ADDING 32
    MOV CHAR, AL     ; STORE THE LOWERCASE CHARACTER

    MOV DX, OFFSET LOWER
    MOV AH, 09H
    INT 21H



    ; TERMINATE

    MOV AH, 4CH
    INT 21H

MAIN ENDP
END MAIN



; MASM Q2.ASM && LINK Q2.OBJ && Q2.EXE
```

```
C:\ASS1>Q2.EXE
ENTER A LETTER IN UPPERCASE : A
SAME LETTER IN LOWERCASE : a
```

3. Write and test a MASM program to add two Hexadecimal Numbers.

```
.MODEL SMALL
.STACK 100H

.DATA

MSG DB "11H+31H = $"
VAL1 DB 11H    ; dec = 17
VAL2 DB 31H    ; dec = 49
VAL3 DB ?," (ASCII)$"  ; EXPECTED 65 OR 41H
DONE DB 0DH,0AH,"FIN$"

.CODE



MAIN PROC
    MOV AX,@DATA
    MOV DS,AX
    LEA DX,MSG
    MOV AH, 09H
    INT 21H

    MOV AL, VAL1
    ADD AL, VAL2    ; ADDING
    MOV VAL3, AL    ; STORING IN VAL3
    ;MOV DL, VAL3
    MOV DX, OFFSET VAL3
```

```
    MOV AH, 09H
    INT 21H        ; PRINTING VAL3

    MOV DX, OFFSET DONE
    INT 21H

    ;EXIT
    MOV AH, 4CH
    INT 21H


MAIN ENDP
END MAIN
```

```
; MASM Q3.ASM && LINK Q3.OBJ && Q3.EXE
```

```
C:\ASS1>Q3.EXE
11H+31H = B (ASCII)
FIN
```

4. Write and test a MASM program to find the second max and second min from an array.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code
.STACK 100H     ; Bytes of stack spaces for the Program

.DATA
ARR DB '5327918'
```

```
COUNT DW ($-ARR)

.CODE

BS PROC
    MOV AX, @DATA
    MOV DS, AX

    CALL PRINTARR

    MOV CX, COUNT
    DEC CX

    L1: PUSH CX             ;STORE OUTER LOOP COUNTER
      MOV SI, OFFSET ARR    ;SI POINTS TO FIRST INDEX IN ARRAY

    L2: MOV AL, [SI]        ; AL USED INSTEAD OF AX, SINCE DATA IS A BYTE LONG

      CMP [SI+1], AL        ;COMPARE ELEMENT AND THE NEXT ELEMENT
      JG L3                 ;IF ELEMENT <= NEXT ELEMENT GOTO L3
      XCHG AL, [SI+1]       ;ELSE SWAP
      MOV [SI], AL

    L3: INC SI
      LOOP L2               ;INNER LOOP

      POP CX                ;BACK TO OUTER LOOP
      LOOP L1               ;OUTER LOOP
```

```
    CALL PRINTARR

    MOV AH,02H
    MOV DL, [ARR+1]
    INT 21H             ;PRINT SECOND MIN
    MOV DL, ' '
    INT 21H

    MOV AX, offset ARR
    ADD AX, COUNT
    SUB AX, 2
    MOV SI, AX
    MOV AH, 02H
    MOV DL, [SI]
    INT 21H             ;PRINT SECOND MAX

    MOV AH, 4CH         ;TERMINATE PROCESS
    INT 21H

BS ENDP

PRINTARR PROC

    MOV AH, 02H

    MOV CX, COUNT
    MOV SI, OFFSET ARR
```

```
    LP: MOV DL, [SI]
        INT 21H
        MOV DL, ' '
        INT 21H
        INC SI
        LOOP LP

    MOV DL, 0DH
    INT 21H
    MOV DL, 0AH
    INT 21H

    RET

PRINTARR ENDP

END BS
```

5. Write and test a MASM program to display a terminating message.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code
.STACK 100H     ; Bytes of stack spaces for the Program
```

```
.DATA

MESSAGE DB "TERMINATING....", 0DH, 0AH, "TERMINATED !", "$"



.CODE

MAIN PROC

    MOV AX, @DATA
    MOV DS, AX
    LEA DX, MESSAGE
    MOV AH, 09H
    INT 21H



    ; TERMINATE

    MOV AH, 4CH
    INT 21H

MAIN ENDP
END MAIN

; MASM Q5.ASM && LINK Q5.OBJ && Q5.EXE
```

```
C:\ASS1>Q5.EXE
TERMINATING....
TERMINATED !
```

6. Write and test a MASM program to Take a character from the keyboard and print it.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code
.STACK 100H     ; Bytes of stack spaces for the Program


.DATA

MSG1 DB "ENTER A CHARACTER : ", "$"
MSG2 DB 0DH, 0AH, "ENTERED CHARACTER : "
CHAR DB ?, "$"


.CODE

MAIN PROC

    MOV AX, @DATA
    MOV DS, AX
    LEA DX, MSG1
    MOV AH, 09H
    INT 21H


    ; ENTER THE LETTER

    MOV AH, 01H
```

```
    INT 21H
    ADD Al, 00H
    MOV CHAR, Al      ; STORE FOR PRINTING

    LEA DX, MSG2
    MOV AH, 09H
    INT 21H


    ; TERMINATE

    MOV AH, 4CH
    INT 21H

MAIN ENDP
END MAIN


; MASM Q6.ASM && LINK Q6.OBJ && Q6.EXE
```

```
C:\ASS1>Q6.EXE
ENTER A CHARACTER : B
ENTERED CHARACTER : B
```

7. Write and test a MASM program to validate second numbers is less than the first.

```
.MODEL SMALL
```

```
.STACK 100H

.DATA

MSG1 DB "ENTER TWO NUMBERS",13,10,"$"
NUM1 DB ?
NUM2 DB ?
MSG2 DB 13,10,"SECOND NUMBER IS SMALLER OK..$"
MSG3 DB 13,10,"SECOND NUMBER IS GREATER FAILED VALIDATION ..$"
IN1 DB 13,10,"ENTER FIRST NUMBER: $"
IN2 DB 13,10,"ENTER SECOND NUMBER: $"

.CODE

MAIN PROC

    MOV AX,@DATA
    MOV DS,AX
    LEA DX,MSG1
    MOV AH,9
    INT 21H

    ; FIRST NUMBER
    MOV AH,9
    LEA DX, IN1
    INT 21H ;PROMPT
    MOV AH,1
    INT 21H  ;INPUT
```

```
    MOV NUM1,AL

    ; SECOND NUMBER
    MOV AH,9
    LEA DX, IN2
    INT 21H ;PROMPT
    MOV AH,1
    INT 21H ;INPUT
    MOV NUM2, AL

    MOV AL, NUM1
    CMP AL, NUM2
    JGE OK ; VALIDATION OK
    MOV AH,9
    LEA DX,MSG3
    INT 21H
    JMP EXIT

OK:
    MOV AH,9
    LEA DX, MSG2
    INT 21H
    ;EXIT

EXIT:
    MOV AH,4CH
    INT 21H
```

```
MAIN ENDP

END MAIN


; MASM Q7.ASM && LINK Q7.OBJ && Q7.EXE
```

```
C:\ASS1>Q7.EXE
ENTER TWO NUMBERS

ENTER FIRST NUMBER: 1
ENTER SECOND NUMBER: 2
SECOND NUMBER IS GREATER FAILED VALIDATION ..
C:\ASS1>Q7.EXE
ENTER TWO NUMBERS

ENTER FIRST NUMBER: 3
ENTER SECOND NUMBER: 2
SECOND NUMBER IS SMALLER OK..
```

8. Write and test a MASM program to find maximum and minimum from an array.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code

.STACK 100H     ; Bytes of stack spaces for the Program


.DATA

ARR DB '5327918'

COUNT DW ($-ARR)


.CODE


BS PROC
```

```
MOV AX, @DATA
MOV DS, AX

CALL PRINTARR

MOV CX, COUNT
DEC CX

L1: PUSH CX              ;STORE OUTER LOOP COUNTER
   MOV SI, OFFSET ARR    ;SI POINTS TO FIRST INDEX IN ARRAY

L2: MOV AL, [SI]         ; AL USED INSTEAD OF AX, SINCE DATA IS A BYTE LONG

   CMP [SI+1], AL        ;COMPARE ELEMENT AND THE NEXT ELEMENT
   JG L3                 ;IF ELEMENT <= NEXT ELEMENT GOTO L3
   XCHG AL, [SI+1]       ;ELSE SWAP
   MOV [SI], AL

L3: INC SI
   LOOP L2               ;INNER LOOP

   POP CX                ;BACK TO OUTER LOOP
   LOOP L1               ;OUTER LOOP

;CALL PRINTARR

MOV AH,02H
MOV DL, [ARR]
```

```
    INT 21H                 ;PRINT SECOND MIN
    MOV DL, ' '
    INT 21H

    MOV AX, offset ARR
    ADD AX, COUNT
    SUB AX, 1
    MOV SI, AX
    MOV AH, 02H
    MOV DL, [SI]
    INT 21H                 ;PRINT SECOND MAX

    MOV AH, 4CH             ;TERMINATE PROCESS
    INT 21H

BS ENDP

PRINTARR PROC

    MOV AH, 02H

    MOV CX, COUNT
    MOV SI, OFFSET ARR

    LP: MOV DL, [SI]
      INT 21H
      MOV DL, ' '
      INT 21H
```

```
    INC SI
    LOOP LP

  MOV DL, 0DH
  INT 21H
  MOV DL, 0AH
  INT 21H

  RET

PRINTARR ENDP

END BS
```

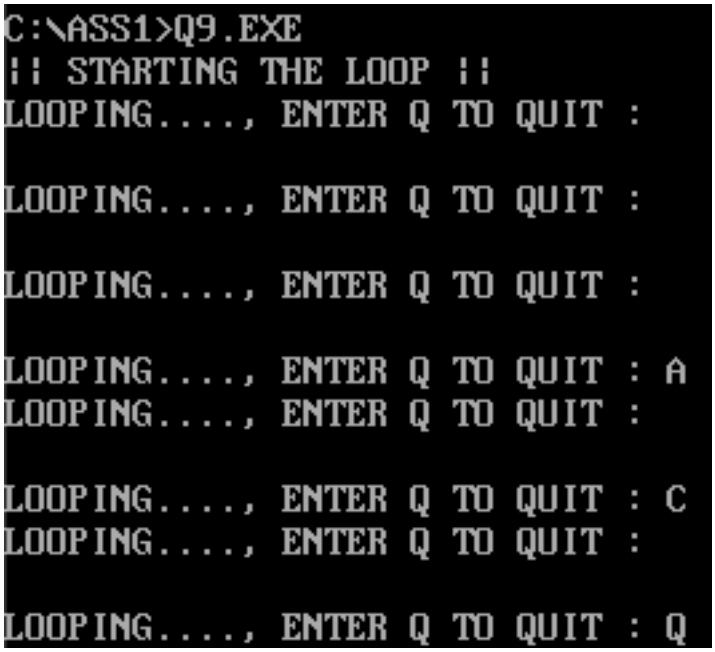
```
C:\ASS1>Q8.EXE
5 3 2 7 9 1 8
1 9
```

9. Write and test a MASM program to loop until the user decides to quit.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code
.STACK 100H     ; Bytes of stack spaces for the Program


.DATA

MSG1 DB "|| STARTING THE LOOP ||","$"
LMSG DB 13,10, "LOOPING...., ENTER Q TO QUIT : ","$"
```

```
.CODE

MAIN PROC

    MOV AX, @DATA
    MOV DS, AX
    LEA DX, MSG1
    MOV AH, 09H
    INT 21H

LOOP1:
    MOV AH, 09H
    LEA DX, LMSG
    INT 21H
    MOV AH, 01H
    INT 21H
    CMP AL, 'Q'
    JNZ LOOP1


    ;TERMINATE

    MOV AH, 4CH
    INT 21H

MAIN ENDP
END MAIN
```

; MASM Q9.ASM && LINK Q9.OBJ && Q9.EXE



10. Write and test a MASM program to print all the characters from A-Z.

```
.MODEL SMALL    ; Program uses a minimum of 64K of memory for code
.STACK 100H     ; Bytes of stack spaces for the Program

.CODE

MAIN PROC

    MOV AH, 02H
    MOV DL, 'A'
    MOV CX, 1AH     ; Stores 26 in CX
```

```
LOOP1:
    INT 21H
    INC DL
    LOOP LOOP1

    ;TERMINATE

    MOV AH, 4CH
    INT 21H

MAIN ENDP
END MAIN

; MASM Q10.ASM && LINK Q10.OBJ && Q10.EXE
```