Name: **Muhammad Maiz Nadeem**
Reg. ID: **SP21-BCS-052**

Display Mode

Question 1:

Write a string (your name) horizontally and scroll it continuously towards the left. Each letter that disappears from the left end should reappear in the same row from right end. (You might want to use first or last row to ensure that row remains the same)

Answer:

```
ORG 100h

.DATA

    STR1    DB  'MAIZ NADEEM$'
    SIZE    DW  11

.CODE

MAIN PROC
                MOV AH, 0
                MOV AL, 3
                INT 10h

                MOV AX, 0xB800
                MOV ES, AX
                MOV SI, (13-1)*160 + (5-1)*2

                MOV CX, SIZE
                MOV AH, 0x07
                XOR BX, BX

    NAMEPRINT:  MOV AL, [STR1 + BX]
                MOV ES:SI, AX
                ADD SI, 2
                INC BX
                LOOP NAMEPRINT
```

```
    PROCCALL:   CALL LEFT
                JMP PROCCALL

                RET
MAIN ENDP


LEFT PROC

                MOV CX, SIZE

    ERASE:      CMP SI, 1920
                JLE ERASE2
                SUB SI, 2
                MOV ES:SI, 0X0720
                LOOP ERASE
                JMP AFTER

    ERASE2:     MOV SI, 2078
                MOV ES:SI, 0X0720
                JMP ERASE

    AFTER:      MOV CX, SIZE
                MOV BX, 0
                SUB SI, 2
                MOV DX, 1920

                CMP SI, 1918
                JNE  NAMEPRINT2
                MOV SI, 2078

    NAMEPRINT2: CMP SI, 2080
                JGE NAMEPRINT3
                MOV AL, [STR1 + BX]
                MOV ES:SI, AX
                ADD SI, 2
                INC BX
                LOOP NAMEPRINT2
                RET

    NAMEPRINT3: MOV SI, DX
                SUB DX, 2
                JMP NAMEPRINT2

                RET
LEFT ENDP
```

Keyboard Interrupt

Question 2:

Connect KB interrupt such that on each release of 'P' the movement of the string toggles.

Answer:

```
ORG 100h

.DATA

    STR1    DB  'MAIZ NADEEM$'
    SIZE    DW  11
    KFLAG   DB  0
    FLAG    DB  0

.CODE

MAIN PROC

                MOV BX, 0
                MOV ES, BX

                CLI
                MOV ES:[0x9*4], OFFSET [KBISR]
                MOV ES:[0x9*4+2], CS
                STI


                MOV AH, 0
                MOV AL, 3
                INT 10h

                MOV AX, 0xB800
                MOV ES, AX
                MOV SI, (13-1)*160 + (5-1)*2

                MOV CX, SIZE
                MOV AH, 0x07
                XOR BX, BX
```

```
    NAMEPRINT:  MOV AL, [STR1 + BX]
                MOV ES:SI, AX
                ADD SI, 2
                INC BX
                LOOP NAMEPRINT

    PROCCALL:   CMP FLAG, 1
                JNE  SKIP
                CALL LEFT
        SKIP:   JMP PROCCALL

                RET
MAIN ENDP

        KBISR:  PUSH AX
                IN  AL, 0x60
                CMP AL, 0x19
                JNE KNEXTCOMP
                MOV KFLAG, 1

    KNEXTCOMP:  CMP KFLAG, 1
                JNE KEXIT
                CMP AL, 0x99
                JNE KEXIT
                MOV KFLAG, 0
                CMP FLAG, 1
                JE  CHECK2
                MOV FLAG, 1
                JMP KEXIT

       CHECK2:  MOV FLAG, 0

        KEXIT:  MOV AL, 0x20
                OUT 0x20, AL
                POP AX
                IRET


LEFT PROC

                MOV CX, SIZE

    ERASE:      CMP SI, 1920
                JLE ERASE2
                SUB SI, 2
```

```
                MOV ES:SI, 0X0720
                LOOP ERASE
                JMP AFTER

    ERASE2:     MOV SI, 2078
                MOV ES:SI, 0X0720
                JMP ERASE

    AFTER:      MOV CX, SIZE
                MOV BX, 0
                SUB SI, 2
                MOV DX, 1920

                CMP SI, 1918
                JNE  NAMEPRINT2
                MOV SI, 2078

    NAMEPRINT2: CMP SI, 2080
                JGE NAMEPRINT3
                MOV AL, [STR1 + BX]
                MOV ES:SI, AX
                ADD SI, 2
                INC BX
                LOOP NAMEPRINT2
                RET

    NAMEPRINT3: MOV SI, DX
                SUB DX, 2
                JMP NAMEPRINT2

                RET
LEFT ENDP
```

Timer Interrupt

Question 3:

Connect Timer interrupt such that the each shift in string occurs after 2 seconds.

Answer:

```
ORG 100h
```

```
.DATA

    STR1    DB  'MAIZ NADEEM$'
    SIZE    DW  11
    KFLAG   DB  0
    TFLAG   DB  0
    FLAG    DB  0
    COUNT   DB  0

.CODE

MAIN PROC

                MOV BX, 0
                MOV ES, BX

                CLI
                MOV ES:[0x8*4], OFFSET [TISR]
                MOV ES:[0x8*4+2], CS
                MOV ES:[0x9*4], OFFSET [KBISR]
                MOV ES:[0x9*4+2], CS
                STI


                MOV AH, 0
                MOV AL, 3
                INT 10h

                MOV AX, 0xB800
                MOV ES, AX
                MOV SI, (13-1)*160 + (5-1)*2

                MOV CX, SIZE
                MOV AH, 0x07
                XOR BX, BX

    NAMEPRINT:  MOV AL, [STR1 + BX]
                MOV ES:SI, AX
                ADD SI, 2
                INC BX
                LOOP NAMEPRINT

    PROCCALL:   CMP TFLAG, 1
                JNE NEXTCOMP
                MOV TFLAG, 0
                CALL LEFT
```

```
    NEXTCOMP:   CMP FLAG, 1
                JNE SKIP
                CALL LEFT

    SKIP:       JMP PROCCALL

                RET
MAIN ENDP

    KBISR:      PUSH AX
                IN  AL, 0x60
                CMP AL, 0x19
                JNE KNEXTCOMP
                MOV KFLAG, 1

    KNEXTCOMP:  CMP KFLAG, 1
                JNE KEXIT
                CMP AL, 0x99
                JNE KEXIT
                MOV KFLAG, 0
                CMP FLAG, 1
                JE  CHECK2
                MOV FLAG, 1
                JMP KEXIT

    CHECK2:     MOV FLAG, 0

    KEXIT:      MOV AL, 0x20
                OUT 0x20, AL
                POP AX
                IRET


    TISR:       PUSH AX
                INC COUNT
                CMP COUNT, 36
                JNE EXIT
                MOV TFLAG, 1
                MOV COUNT, 0


    EXIT:       MOV AL, 0x020
                OUT 0x20, AL
                POP AX
                IRET
```

```
LEFT PROC

                MOV CX, SIZE

    ERASE:      CMP SI, 1920
                JLE ERASE2
                SUB SI, 2
                MOV ES:SI, 0X0720
                LOOP ERASE
                JMP AFTER

    ERASE2:     MOV SI, 2078
                MOV ES:SI, 0X0720
                JMP ERASE

    AFTER:      MOV CX, SIZE
                MOV BX, 0
                SUB SI, 2
                MOV DX, 1920

                CMP SI, 1918
                JNE  NAMEPRINT2
                MOV SI, 2078

    NAMEPRINT2: CMP SI, 2080
                JGE NAMEPRINT3
                MOV AL, [STR1 + BX]
                MOV ES:SI, AX
                ADD SI, 2
                INC BX
                LOOP NAMEPRINT2
                RET

    NAMEPRINT3: MOV SI, DX
                SUB DX, 2
                JMP NAMEPRINT2

                RET
LEFT ENDP
```