# Name: Muhammad Maiz Nadeem

Reg. ID: SP21-BCS-052

# Display Mode

# Question 1:

1. Print a string vertically in the middle of screen and then scroll it vertically downwards, such that if a letter exits the screen from bottom it reappears from the top.

# 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, (13-5-1)\*160 + (40-1)\*2 |
|  |
| MOV CX, SIZE |
| MOV AH, 0x07 |
| XOR BX, BX |
|  |
| NAMEPRINT:  MOV AL, [STR1 + BX] |
| MOV ES:SI, AX |
| ADD SI, 80\*2 |
| INC BX |
| LOOP NAMEPRINT |
|  |
| PROCCALL:   CALL DOWN |
| JMP PROCCALL |
|  |
| RET |
| MAIN ENDP |
|  |
|  |
| DOWN PROC |
|  |
| MOV CX, SIZE |
| CMP SI, 1678 |
| JNE ERASE |
| PUSH SI |
| MOV SI, (25-1)\*160 + (40-1)\*2 |
| MOV ES:SI, 0X0720 |
| POP SI |
|  |
| ERASE:      CMP SI, (40-1)\*2 |
| JLE ERASE2 |
| SUB SI, 80\*2 |
| MOV ES:SI, 0X0720 |
| LOOP ERASE |
| JMP AFTER |
|  |
| ERASE2:     MOV SI, (25-1)\*160 + (40-1)\*2 |
| MOV ES:SI, 0X0720 |
| JMP ERASE |
|  |
| AFTER:      MOV CX, SIZE |
| MOV BX, 0 |
| ADD SI, 80\*2 |
| MOV DX, (40-1)\*2 |
|  |
| NAMEPRINT2: CMP SI, 25\*160 + (8)\*2 |
| JGE NAMEPRINT3 |
| MOV AL, [STR1 + BX] |
| MOV ES:SI, AX |
| ADD SI, 80\*2 |
| INC BX |
| LOOP NAMEPRINT2 |
| RET |
|  |
| NAMEPRINT3: MOV SI, DX |
| ADD DX, 160 |
| JMP NAMEPRINT2 |
|  |
| RET |
| DOWN ENDP |

# Keyboard Interrupt

# Question 2:

1. Hook keyboard interrupt such that the scrolling toggles on subsequent presses of button 'p'.

# 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, (13-5-1)\*160 + (40-1)\*2 |
|  |
| MOV CX, SIZE |
| MOV AH, 0x07 |
| XOR BX, BX |
|  |
| NAMEPRINT:  MOV AL, [STR1 + BX] |
| MOV ES:SI, AX |
| ADD SI, 80\*2 |
| INC BX |
| LOOP NAMEPRINT |
|  |
| PROCCALL:   CMP FLAG, 1 |
| JNE  SKIP |
| CALL DOWN |
| 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 |
|  |
|  |
| DOWN PROC |
|  |
| MOV CX, SIZE |
| CMP SI, 1678 |
| JNE ERASE |
| PUSH SI |
| MOV SI, (25-1)\*160 + (40-1)\*2 |
| MOV ES:SI, 0X0720 |
| POP SI |
|  |
| ERASE:      CMP SI, (40-1)\*2 |
| JLE ERASE2 |
| SUB SI, 80\*2 |
| MOV ES:SI, 0X0720 |
| LOOP ERASE |
| JMP AFTER |
|  |
| ERASE2:     MOV SI, (25-1)\*160 + (40-1)\*2 |
| MOV ES:SI, 0X0720 |
| JMP ERASE |
|  |
| AFTER:      MOV CX, SIZE |
| MOV BX, 0 |
| ADD SI, 80\*2 |
| MOV DX, (40-1)\*2 |
|  |
| NAMEPRINT2: CMP SI, 25\*160 + (8)\*2 |
| JGE NAMEPRINT3 |
| MOV AL, [STR1 + BX] |
| MOV ES:SI, AX |
| ADD SI, 80\*2 |
| INC BX |
| LOOP NAMEPRINT2 |
| RET |
|  |
| NAMEPRINT3: MOV SI, DX |
| ADD DX, 160 |
| JMP NAMEPRINT2 |
|  |
| RET |
| DOWN ENDP |

s

# Timer Interrupt

# Question 3:

1. Hook timer interrupt in addition to KB interrupt so that each scroll of string occurs after half a second.

# 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, (13-5-1)\*160 + (40-1)\*2 |
|  |
| MOV CX, SIZE |
| MOV AH, 0x07 |
| XOR BX, BX |
|  |
| NAMEPRINT:  MOV AL, [STR1 + BX] |
| MOV ES:SI, AX |
| ADD SI, 80\*2 |
| INC BX |
| LOOP NAMEPRINT |
|  |
| PROCCALL:   CMP TFLAG, 1 |
| JNE NEXTCOMP |
| MOV TFLAG, 0 |
| CALL DOWN |
|  |
| NEXTCOMP:   CMP FLAG, 1 |
| JNE SKIP |
| CALL DOWN |
|  |
| 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, 18 |
| JNE EXIT |
| MOV TFLAG, 1 |
| MOV COUNT, 0 |
|  |
|  |
| EXIT:       MOV AL, 0x020 |
| OUT 0x20, AL |
| POP AX |
| IRET |
|  |
|  |
| DOWN PROC |
|  |
| MOV CX, SIZE |
| CMP SI, 1678 |
| JNE ERASE |
| PUSH SI |
| MOV SI, (25-1)\*160 + (40-1)\*2 |
| MOV ES:SI, 0X0720 |
| POP SI |
|  |
| ERASE:      CMP SI, (40-1)\*2 |
| JLE ERASE2 |
| SUB SI, 80\*2 |
| MOV ES:SI, 0X0720 |
| LOOP ERASE |
| JMP AFTER |
|  |
| ERASE2:     MOV SI, (25-1)\*160 + (40-1)\*2 |
| MOV ES:SI, 0X0720 |
| JMP ERASE |
|  |
| AFTER:      MOV CX, SIZE |
| MOV BX, 0 |
| ADD SI, 80\*2 |
| MOV DX, (40-1)\*2 |
|  |
| NAMEPRINT2: CMP SI, 25\*160 + (8)\*2 |
| JGE NAMEPRINT3 |
| MOV AL, [STR1 + BX] |
| MOV ES:SI, AX |
| ADD SI, 80\*2 |
| INC BX |
| LOOP NAMEPRINT2 |
| RET |
|  |
| NAMEPRINT3: MOV SI, DX |
| ADD DX, 160 |
| JMP NAMEPRINT2 |
|  |
| RET |
| DOWN ENDP |