

# MPI Assignment - 2

Name - Rohit Singh

Roll no. - 2820011

Disc. - B.Tech CSE

Section - A1

Date - 17 Nov 2022

- ① W.A.P to count no. of 1's in a byte and write it into BL register.
- ② let a byte stores a number 50H

MOV AL, 0050H

MOV BL, 0000H

MOV CL, 8

L1: SHL AL, 1

JNC L2

INC BL

L2: DEC CL

JNZ L1

HLT

- ② Write an ALP to find the max. byte from a block of 256 bytes starting at 53000H.

MOV AX, 5000H

MOV DS, AX

MOV BX, 3000H

MOV CX, 0100H

MOV AH, 00H

NEXT: CMP AH, [BX]

JAE PASS

MOV AH, [BX]

PASS: INC BX  
DEC CX  
JNZ NEXT  
HLT

Maximum byte stored in AH register

- ③ Write an ALP to find the average of an array of size 256 bytes of unsigned integers starting at 6A000H.

④

```
MOV AX, 6000H
MOV DS, AX
XOR AX, AX
MOV BX, 00H
MOV CX, 100H
AGAIN: ADD AL, [BX + 200H]
        ADC AH, 00H
NEXT:  INC BX
        LOOP AGAIN
        XOR DX, DX
        DIV BX
        HLT
```

- ④ Write an ALP that counts the no. of letters 'M' in a string of size 256 bytes starts at 5F600H. (4D is the ASCII code of letter M).

⑤

```
MOV AX, 5000H
MOV DS, AX
LGA SI, [F600H]
```

```
MOV CX, 100H  
MOV DL, 0DH  
MOV AL, 4DH  
AGAIN: CMP AL, [SI]  
JNE NEXT  
INC DL  
NEXT: INC SI  
LOOP AGAIN  
HLT
```

- ⑤ Write a piece of code that transfers a block of 256 bytes stored at locations starting at ~~34~~<sup>starting</sup> 000H. Store the result at 3600H.

```
MOV AX, 3000H  
MOV DS, AX  
MOV BX, 0000H  
MOV CX, 0100H  
NEXT: MOV AL, [BX + 4000H]  
MOV [BX + 6000H], AL  
INC BX  
DEC CX  
JNZ NEXT  
HLT
```

- ⑥ Write an ALP to find the factorial of a number stored at location in memory.

- ⑥ Let offset address of the memory (where no. is stored) be [SI] at 5000 base address

```
MOV AX, 5000H  
MOV DS, AX  
MOV CX, [BX]  
MOV AX, F0001H  
LOOP: MUL CX  
DEC CX  
JNZ LOOP  
HLT
```

- ⑦ Write an ALP to find the minimum value of a byte from a block of 256 bytes starting at 53000H. Store the result at 56000H.

```
MOV AX, 5000H  
MOV DS, AX  
MOV BX, 3000H  
MOV CX, 0100H  
MOV AH, 00H  
NEXT: CMP AH, [BX]  
JBE PASS  
MOV AH, [BX]  
PASS: INC BX  
DEC CX  
JNZ NEXT  
MOV [3000H], AH  
HLT
```

- ⑧ Write an ALP to find the 2's complement of a block of 100 bytes starting at 53000H and store the result starting from 54000H.

⑨

```

MOV AX, 5000H
MOV DS, AX
MOV SI, 3000H
MOV DI, 4000H
MOV BX, 64H
NEXT: NEG [SI]
MOV AX, [SI]
MOV [DI], AX
INC SI
INC DI
DEC BX
JNZ NEXT
HLT

```

⑩. Move a byte string, 16 bytes long, from the offset 0200H to 0300H in the segment 7000H.

⑩

```

MOV AX, 7000H
MOV DS, AX
MOV SI, 0200H
MOV DI, 0300H
MOV CX, 16
AGAIN: MOV AL, [SI]
INC SI
MOV [DI], AL
INC DI
LOOP AGAIN
HLT

```