MPMC LAB

1. 8086 ADDITION

(Write an assembly language program to ADD two numbers of a 16 bit data)

PROGRAM:

```
MOV AX, [1000h]
MOV BX, [1002h]
MOV CL, 00h
ADD AX, BX
MOV [1004h], AX
JNC jump
INC CL
Jump:
MOV [1006h], CL
HLT
```

2. 8086 SUBTRACTION

(Write an assembly language program to SUBTRACT two numbers of a 16 bit data)

PROGRAM:

```
MOV AX, [1000h]
MOV BX, [1002h]
MOV CL, 00h
SUB AX, BX
JNC jump
INC CL
NOT AX
ADD AX, 0001h
Jump:
MOV [1004h], AX
MOV [1006h], CL
HLT
```

3. 8086 MULTIPLICATION

(Write an assembly language program to MULTIPLY two numbers of a 16 bit data)

PROGRAM:

```
MOV SI, 1100H
MOV AX, [SI]
MOV BX, [SI+2]
MUL  BX
MOV [SI+4], AX
MOV [SI+6], DX
HLT
```

4. 8086 DIVISION

(Write an assembly language program to DIVIDE two numbers of a 16 bit data)

PROGRAM:

```
Org 100h
MOV AX, 800AH
MOV BX, 1856H
DIV BX
ret
```

5. FIND SMALLEST DATA

(Write an assembly language program to find SMALLEST data in an array)

PROGRAM:

```
MOV SI, 1100h

MOV DI, 1200h

MOV CL, [SI]

INC SI

MOV AL, [SI]

DEC CL

AGAIN:

INC SI

MOV BL, [SI]

CMP AL, BL

JC AHEAD

MOV AL, BL

AHEAD:

DEC CL

JNZ AGAIN

MOV [DI], AL

HLT
```

6. FIND LARGEST DATA

(Write an assembly language program to find LARGEST data in an array)

PROGRAM:

```
MOV SI, 1100h

MOV DI, 1200h

MOV CL, [SI]

INC SI

MOV AL, [SI]

DEC CL

AGAIN:

INC SI

MOV BL, [SI]

CMP AL, BL

JNC AHEAD

MOV AL, BL

AHEAD:

DEC CL

JNZ AGAIN

MOV [DI], AL

HLT
```

7. SORT IN ASCENDING ORDER

(Write an assembly language program to sort an array in ASCENDING ORDER)

PROGRAM:

```
MOV SI, 1100h

MOV CL, [SI]

DEC CL

REPEAT:

MOV SI, 1100h

MOV CH, [SI]

DEC CH

INC SI

REPCOM:

MOV AL, [SI]

INC SI

CMP AL, [SI]

JC AHEAD

XCHG AL, [SI]

XCHG AL, [SI-1]

AHEAD:

DEC CH
```

```
JNZ REPCOM

DEC CL

JNZ REPEAT

HLT
```

8. SORT IN DESCENDING ORDER

(Write an assembly language program to sort an array in DESCENDING ORDER)

```
PROGRAM:

MOV SI, 1100h

MOV CL, [SI]

DEC CL

REPEAT:

MOV SI, 1100h

MOV CH, [SI]

DEC CH

INC SI

REPCOM:

MOV AL, [SI]

INC SI

CMP AL, [SI]
```

```
JNC AHEAD

XCHG AL, [SI]

XCHG AL, [SI-1]

AHEAD:

DEC CH

JNZ REPCOM

DEC CL

JNZ REPEAT

HLT
```