```
;PROGRAM TITLE:ADDITION OF TWO 8 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV AL,[SI]
INC SI
MOV BL,[SI]
ADD AL,BL
MOV [DI],AL
INT 21H
```

```
;PROGRAM TITLE:ADDITION OF TWO 16 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV DI,0160H
CLC
MOV SI,150H
MOV AX,[SI]
ADD SI,02H
MOV BX,[SI]
ADC AX,BX
ADD SI,02H
MOV [DI],AX
INT 21H
```

```
;PROGRAM TITLE:ADDITION OF TWO 32 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV CL,02H
MOV SI,150H
MOV DI,170H
MOV BP,180H
CLC
UP:MOV AX,[SI]
ADD SI,02H
MOV BX,[DI]
ADC AX,BX
MOV [BP],AX
ADD BP,02H
ADD DI,02H
LOOP UP
INT 21H
```

```
;PROGRAM TITLE:ADDITION OF TWO 64 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
XOR AX,AX
MOV BP,180H
MOV CX,04H
CLC
UP:MOV AX,[SI]
ADD SI,02H
MOV BX,[DI]
ADC AX,BX
MOV [BP],AX
ADD BP,02H
ADD DI,02H
LOOP UP
INT 21H
```

```
;PROGRAM TITLE: SUBTRACTION OF TWO 8 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
XOR AX,AX
MOV AL,[SI]
INC SI
MOV BL,[SI]
SUB AL,BL
MOV [DI],AL
INT 21H
```

```
;PROGRAM TITLE: SUBTRACTION OF TWO 16 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
XOR AX,AX
MOV AX,[SI]
ADD SI,02H
MOV BX,[SI]
SBB AX,BX
MOV [DI],AX
INT 21H
```

```
;PROGRAM TITLE: SUBTRACTION OF TWO 32 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
MOV BP,0180H
MOV CX,02H
XOR AX,AX
CLC
UP:MOV AX,[SI]
ADD SI,02H
MOV BX,[DI]
ADD DI,02H
SBB AX,BX
MOV [BP],AX
ADD BP,02H
LOOP UP
INT 21H
```

```
;PROGRAM TITLE: SUBTRACTION OF TWO 64 BIT NUMBERS
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
MOV BP,0180H
MOV CX,04H
XOR AX,AX
CLC
UP:MOV AX,[SI]
ADD SI,02H
MOV BX,[DI]
ADD DI,02H
SBB AX,BX
MOV [BP],AX
ADD BP,02H
LOOP UP
INT 21H
```

;PROGRAM TITLE:MULTIPLICATION OF TWO 8 BIT NUMBERS

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
MOV AL,[SI]
INC SI
MOV BL,[SI]
MUL BL
MOV [DI],AL
INT 21H
```

;PROGRAM TITLE:MULTIPLICATION OF TWO 16 BIT NUMBERS

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
MOV AX,[SI]
ADD SI,02H
MOV BX,[SI]
MUL BX
MOV [DI],AX
ADD DI,02
MOV [DI],DX ;to get extra values as result
INT 21H
```

SIMPLIFIED VERSION

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
MOV AX,[SI]
MOV BX,SI+02H
MUL BX
MOV [DI],AX
MOV DI+02,DX
INT 21H
```

;PROGRAM TITLE:DIVISION OF TWO 8 BIT NUMBERS

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
XOR AX,AX
MOV AL,[SI]
INC SI
MOV BL,[SI]
DIV BL
MOV [DI],AL
INT 21H
```

;PROGRAM TITLE:DIVISION OF TWO 16 BIT NUMBERS

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
XOR AX,AX
MOV AX,[SI]
MOV BX,SI+02H
DIV BX
MOV [DI],AX
INT 21H
```

```
;PROGRAM TITLE:SUM OF N 8BIT BIT NUMBERS

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
XOR AX,AX
MOV CL,[SI]
CLC
UP:INC SI
MOV BL,[SI]
ADC AL,BL
MOV [DI],AL
LOOP UP
INT 21H
```

```
;PROGRAM TITLE:SUM OF N 8BIT NUMBER(AS PER
PROGRAM)

ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV DI,160H
MOV CL,[SI]
INC SI
MOV AL,[SI]
UP:INC SI
MOV BL,[SI]
ADC AL,BL
JNC  DOWN
INC AH
DOWN:LOOP UP
MOV [DI],AL
INC DI
MOV [DI],AH
INT 21H
```

```
; PROGRAM TITLE:SUM OF N 16BIT BIT NUMBERS

ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV DI,0160H
CLC
MOV CL,[SI]
INC SI
MOV AX,[SI]
UP:INC SI
INC SI
MOV BX,[SI]
ADD AX,BX
JNC DOWN
INC DX
DOWN:LOOP UP
MOV [DI],AX
ADD DI,02H
MOV [DI],DX
INT 21H
```

```
;PROGRAM TITLE:SUM OF N 16BIT NUMBER(AS PER
PROGRAM)

ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV DI,160H
MOV CL,[SI]
INC SI
MOV AX,[SI]
UP:ADD SI,02H
MOV BX,[SI]
ADC AX,BX
JNC  DOWN
INC DX
DOWN:LOOP UP
MOV [DI],AX
ADD DI,02H
MOV [DI],DX
INT 21H
```

```
;PROGRAM TITLE:REVERSE A STRING

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
MOV CL,06H
ADD DI,05H
XOR AX,AX
UP:MOV AL,[SI]
INC SI
MOV [DI],AL
DEC DI
LOOP UP
INT 21H
```

```
;PROGRAM TITLE:MOVING A STRING
;MOVE FROM DS:SI TO ES:DI

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV ES,AX
XOR AX,AX
MOV SI,0150H
MOV DI,0170H
MOV CL,[SI]
CLD
INC SI
REPZ MOVSB
INT 21H
```

```
;PROGRAM TITLE:COMPARISION OF TWO STRNGS
INCLUDE 'EMU8086.INC'
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV ES,AX
XOR AX,AX
MOV SI,200H
MOV DI,210H
MOV CL,05H
REP CMPSB
JZ DOWN
MOV BL,00H
PRINT 'STRINGS ARE NOT EQUAL'
JMP LAST
DOWN:MOV BL,01H
PRINT 'STRINGS ARE EQUAL'
LAST:INT 21H
```

```
;PROGRAM TITLE:NEGATE A STRING

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV ES,AX
XOR AX,AX
MOV SI,0150H
MOV DI,0170H
CLD
UP:MOV AL,[SI]
NEG AL
STOSB
INC SI     ;Store byte in AL into ES:[DI]. Update DI.
LOOP UP
INT 21H
```

```
;PROGRAM TITLE:FINDING LENGTH OF A STRING
ORG 0100H
MOV AX,0700H
MOV ES,AX
XOR AX,AX
MOV DI,0150H
MOV SI,0160H
XOR CX,CX
CLD
UP:SCASB
JZ DOWN
INC CL
JMP UP
DOWN:MOV [SI],CL
INT 21H
```

```
;PROGRAM TITLE:LARGEST NUMBER

ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV DI,160H
MOV CL,[SI]
INC SI
MOV AL,[SI]
UP:INC SI
MOV BL,[SI]
CMP AL,BL
JNB DOWN
MOV AL,BL
DOWN:LOOP UP
MOV [DI],AL
INT 21H
```

```
;PROGRAM TITLE:SMALLEST NUMBER

ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV DI,160H
MOV CL,[SI]
INC SI
MOV AL,[SI]
UP:INC SI
MOV BL,[SI]
CMP AL,BL
JB DOWN
MOV AL,BL
DOWN:LOOP UP
MOV [DI],AL
INT 21H
```

```
;PROGRAM TITLE:LARGEST NUMBER

ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV DI,160H
MOV CL,[SI]
INC SI
MOV AL,[SI]
UP:INC SI
MOV BL,[SI]
CMP AL,BL
JNB DOWN
MOV AL,BL
DOWN:LOOP UP
MOV [DI],AL
INT 21H
```

```
;PROGRAM TITLE:SMALLEST NUMBER

ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV DI,160H
MOV CL,[SI]
INC SI
MOV AL,[SI]
UP:INC SI
MOV BL,[SI]
CMP AL,BL
JB DOWN
MOV AL,BL
DOWN:LOOP UP
MOV [DI],AL
INT 21H
```

;PROGRAM TITLE:ASCENDING ORDER

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV CL,[SI]
DEC CL
UP2:MOV SI,0150H
MOV CH,[SI]
DEC CH
INC SI
UP1:MOV AL,[SI]
INC SI
CMP AL,[SI]
JC DOWN
XCHG AL,[SI]
DEC SI
XCHG AL,[SI]
INC SI
DOWN:DEC CH
JNZ UP1
DEC CL
JNZ UP2
INT 21H
```

;PROGRAM TITLE:DESCENDING ORDER

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
XOR AX,AX
MOV SI,0150H
MOV CL,[SI]
DEC CL
UP2:MOV SI,0150H
MOV CH,[SI]
DEC CH
INC SI
UP1:MOV AL,[SI]
INC SI
CMP AL,[SI]
JNC DOWN
XCHG AL,[SI]
DEC SI
XCHG AL,[SI]
INC SI
DOWN:DEC CH
JNZ UP1
DEC CL
JNZ UP2
INT 21H
```

```
; ;PROGRAM TITLE:PACKED BCD TO UNPACKED BCD

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV AL,[SI]
MOV CL,04H
AND AL,0FH
MOV [DI],AL
MOV AL,[SI]
AND AL,0F0H
ROR AL,CL
INC DI
MOV [DI],AL
INT 21H
```

```
;PROGRAM TITLE:UNPACKED BCD TO PACKED BCD

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV CL,04H
MOV AX,[SI]
ROR AL,CL
SHR AX,CL
MOV [DI],AX
INT 21H
```

```
;PROGRAM TITLE:SUM OF SQUARES OF GIVEN NUMBERS

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV CL,[SI]
MOV BL,00H
UP:INC SI
MOV AL,[SI]
MUL AL
ADD AL,BL
MOV BL,AL
LOOP UP
MOV [DI],BL
INT 21H
```

```
; PROGRAM TITLE:FIND AVERAGE OF N NUMBERS

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0170H
XOR AX,AX
MOV CL,[SI]
MOV DL,CL
DEC CL
INC SI
CLC
MOV AL,[SI]
UP:INC SI
MOV BL,[SI]
ADC AL,BL
LOOP UP
DIV DL
MOV [DI],AL
INT 21H
```

```
;PROGRAM TITLE:NUMBER OF 0'S AND 1'S IN ARRAY

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV CL,08H
MOV AL,[SI]
RPT:ROL AL,01H
JC ONE
INC BL
JMP NXT
ONE:INC BH
NXT:LOOP RPT
MOV [DI],BL
INC DI
MOV [DI],BH
INT 21H
```

```
;PROGRAM TITLE:CONVERT BCD TO GRAY CODE

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV AL,[SI]
MOV BL,AL
SHR AL,01H
XOR BL,AL
INC SI
MOV [DI],BL
INT 21H
```

; PROGRAM TITLE:FIND +VE AND -VE NUMBERS IN AN ARRAY

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,160H
MOV CL,[SI]
INC SI
UP2:MOV AL,[SI]
ROL AL,01H
JC UP1
INC BL
JMP DOWN
UP1:INC DL
DOWN:INC SI
LOOP UP2
MOV [DI],BL
INC DI
MOV [DI],DL
INT 21H
```

;PROGRAM TITLE:SQUARE ROOT OF A PERFECT SQUARE

```
ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV BL,[SI]
MOV CL,00H
UP:MOV AL,CL
MUL CL
CMP AL,BL
JZ DOWN
INC CL
JNZ UP
DOWN:MOV [DI],CL
INT 21H
```

```
;PROGRAM TITLE: FIBONACCI SERIES

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV CL,[SI]
MOV AL,01H
MOV BL,00H
UP:MOV DL,AL
ADD AL,BL
MOV [DI],AL
INC DI
MOV BL,DL
LOOP UP
INT 21H
```

```
;PROGRAM TITLE:EVEN AND ODD NUMBERS

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
XOR AX,AX
XOR DX,DX
MOV CL,[SI]
UP:INC SI
MOV AL,[SI]
MOV BL,02
DIV BL
CMP AH,00H
JZ DOWN
INC DH
JMP DOWN1
DOWN:INC DL
DOWN1:LOOP UP
MOV [DI],DL
INC DI
MOV [DI],DH
INT 21H
```

```
; PROGRAM TITLE:FACTORIAL OF A GIVEN NUMBER

ORG 0100H
MOV AX,0700H
MOV DS,AX
MOV SI,0150H
MOV DI,0160H
MOV CX,[SI]
MOV AX,0001H
UP:MUL CX
LOOP UP
MOV [DI],AX
INT 21H
```