3) add, sub, mul,div

```
DATA SEGMENT
   NO1 DW 2345H
   NO2 DW 0010H
   SUM DW ?
   DIFFERANCE DW ?
   MULTIPLICATION DW ?
   QUOTIENT DW ?
   REMAINDER DW ?
DATA ENDS
CODE SEGMENT
   ASSUME CS:CODE, DS:DATA
   START: MOV AX, DATA
   MOV DS,AX

   ;ADDITION
   MOV AX, NO1
   MOV BX, NO2
   ADD AX,BX
   MOV SUM, AX

   ;SUBTRACTION
   MOV AX, NO1
   SUB AX,BX
   MOV DIFFERANCE, AX

   ;MULTIPLICATION
   MOV DX,00H
   MOV AX,NO1
   MUL NO2;
   MOV MULTIPLICATION, AX
   MOV MULTIPLICATION+2,DX
```

```
   ;DIVISION
   MOV DX,00H
   MOV AX,NO1
   DIV BX
   MOV REMAINDER,DX
   MOV QUOTIENT, AX

   MOV AH,4CH
   INT 21H
   CODE ENDS
END START
```

4) array sum in loop

```
DATA SEGMENT
   ARRAY DB 01H,02H,03H,05H,04H
   RESULT DW ?
DATA ENDS

CODE SEGMENT
   ASSUME CS:CODE,DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX

      MOV AH,00H
      MOV CL,04H
      LEA BX, ARRAY
      MOV AL,[BX]

      LOOP1:
         INC BX
         ADD AL,[BX]
         JNC SKIP
         INC AH

      SKIP:
         DEC CL
         JNZ LOOP1

         MOV RESULT,AX
         MOV AH,4CH
         INT 21H
CODE ENDS
END START
```

5) Max number Min number

```
DATA SEGMENT
   ARRAY DB 01H,02H,03H,04H
   MAX DB ?
   DATA ENDS
CODE SEGMENT
   ASSUME CS:CODE, DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX
      MOV CX,0004H
      LEA BX,ARRAY
      MOV AL,[BX]
      LOOP1:  INC BX
            CMP AL,[BX]
            JNC SKIP
            MOV AL,[BX]
      SKIP:   DEC CX
            JNZ LOOP1
      MOV MAX,AL
      MOV AH,4CH
      INT 21H

      CODE ENDS

END START
```

```
DATA SEGMENT
   ARRAY DB 01H,02H,03H,04H,08H
   MIN DB ?
   DATA ENDS
CODE SEGMENT
   ASSUME CS:CODE, DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX
      MOV CX,04H
      MOV BL,79H
      LEA SI,ARRAY
      LOOP1:  MOV AL,[SI]
            CMP AL,BL
            JNC SKIP
            MOV BL,AL
      SKIP:  INC SI
            DEC DX
            JNZ LOOP1
      MOV MIN,BL
      MOV AH,4CH
      INT 21H
      CODE ENDS
END START
```

6) BCD multiplication

```
DATA SEGMENT

   A DB 09H
   B DB 08H

DATA ENDS
CODE SEGMENT
   ASSUME CS:CODE, DS: DATA
   START:
      MOV AX,DATA
      MOV DS, AX
      MOV AL, A
      MOV BL,B
      MUL BL
      AAM
      MOV CH,04H
      MOV CL,04H
      MOV BX,AX
      L2:ROL BX, CL
      MOV DL,BL
      AND DL,0FH
      CMP DL,89
      JBE L4
      ADD DL,07
      L4:ADD DL,30H
      MOV AH,02
      INT 21H
      DEC CH
      JNZ L2
      MOV AX,41H
      INT 21H
   CODE ENDS
END START
```

7) BCD division

```
DATA SEGMENT
   BCD DW 0302H
   NO2 DB 05H
   QUOTIENT DB ?
   REMAINDER DB ?
DATA ENDS
CODE SEGMENT
   ASSUME CS:CODE,DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX
      MOV AX,BCD
      AAD
      DIV NO2
      MOV QUOTIENT,AL
      MOV REMAINDER,AH
      MOV AH,4CH
      INT 21H
   CODE ENDS
END START
```

8) BCD addition

```
DATA SEGMENT
   NO1 DW 0294H
   NO2 DW 0325H
   RESULT DW ?
DATA ENDS
CODE SEGMENT
   ASSUME CS:CODE, DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX
      MOV AX,NO1
      MOV BX,NO2
      ADD AL,BL
      DAA
      MOV B.RESULT,AL
      MOV AH,4CH
      INT 0021H
   CODE ENDS
END START
```

9) BCD to ASCII

```
DATA SEGMENT
   NUM DW 2200H
DATA ENDS
CODE SEGMENT
   ASSUME CS:CODE,DS:DATA
   START:
   MOV AX,DATA
   MOV DS,AX
   MOV AX,NUM
   ADD AX,000FH
   ADD AX,30H

   MOV BX,NUM
   MOV BX,00F0H
   ROR BX,04
   ADD BX,30H

   MOV CX,NUM
   AND CX,0F000H
   ROR CX,12
   ADD CX,30H

   MOV DX,NUM
   AND DX,0F000H
   ROR DX,12
   ADD DX,30H
   MOV AH, 4CH
   INT 21H
   CODE ENDS
END START
```