8 BIT ADDITION:

DATA SEGMENT

  A DB 12H

  B DB 13H

  SUM DB 00H

  CARRY DB 00H

DATA ENDS

CODE SEGMENT

  ASSUME CS:CODE, DS:DATA

START:

  MOV AX,DATA

  MOV DS,AX

  MOV AL,A

  ADD AL,B

  MOV SUM,AL

  JNC SKIP

  INC CARRY

  MOV AL,01H

  HLT

SKIP:

  MOV AL, CARRY

CODE ENDS

END START

16 BIT ADDITION:

```
DATA SEGMENT

   A   DW  12121H

   B   DW  1312H

   SUM  DW  0000H

   CARRY DB  00H

DATA ENDS

CODE SEGMENT

   ASSUME CS:CODE, DS:DATA

START:

   MOV AX,DATA

   MOV DS,AX

   MOV AX,A

   ADD AX,B

   MOV SUM,AX

   JNC SKIP

   INC CARRY

   MOV AL,01H

   HLT

SKIP:

   MOV AL, CARRY

CODE ENDS

END START
```

8 BIT SUBTRACTION:

DATA SEGMENT

  A   DB 43H

  B   DB 13H

  DIF DB 00H

  BORROW DB 00H

DATA ENDS

CODE SEGMENT

  ASSUME CS:CODE, DS:DATA

START:

  MOV AX,DATA

  MOV DS,AX

  MOV AL,A

  SUB AL,B

  MOV DIF,AL

  JNB SKIP

  INC BORROW

  MOV AL,01H

  HLT

SKIP:

  MOV AL, BORROW

CODE ENDS

END START

16 BIT SUBTRACTION:

DATA SEGMENT

   A   DW 4343H

   B   DW 1313H

   DIF DW 0000H

   BORROW DB 00H

DATA ENDS

CODE SEGMENT

  ASSUME CS:CODE, DS:DATA

START:

  MOV AX,DATA

  MOV DS,AX

  MOV AX,A

  SUB AX,B

  MOV DIF,AX

  JNB SKIP

  INC BORROW

  MOV AL,01H

  HLT

SKIP:

  MOV AL, BORROW

CODE ENDS

END START

8 BIT MULTIPLICATION:

DATA SEGMENT

   A   DB 02H

   B   DB 03H

   PRODUCT DW 0000H

DATA ENDS

CODE SEGMENT

   ASSUME CS:CODE, DS:DATA

START:

   MOV AX,DATA

   MOV DS,AX

   MOV AL, A

   MOV BL, B

   MUL BL

   MOV PRODUCT, AX

CODE ENDS

END START

8 BIT DIVISION:

DATA SEGMENT

  A   DB 06H

  B   DB 04H

  QUOTIENT DB 00H

  REMAINDER DB 00H

DATA ENDS

CODE SEGMENT

  ASSUME CS:CODE, DS:DATA

START:

  MOV AX, DATA

  MOV DS, AX

  MOV AL, A

  MOV BL, B

  XOR AH, AH

  DIV BL

  MOV WORD PTR QUOTIENT, AL

  MOV REMAINDER, AH

CODE ENDS

END START

BCD to ASCII:

DATA SEGMENT

  BCD DB 02H

  ASCII DB 00H

DATA ENDS

CODE SEGMENT

  ASSUME CS:CODE, DS:DATA

START:

  MOV AX, DATA

  MOV DS, AX

  MOV AL, BCD

  AND AL, 0F0H

  MOV CL, 04H

  SHR AL, CL

  ADD AL, 30H

  MOV SI, OFFSET ASCII

  MOV [SI+1], AL

  MOV AL, BCD

  AND AL, 0FH

  ADD AL, 30H

  MOV [SI], AL

  MOV AH, 4CH

  INT 21H

END START

ASCII to BCD:

```
DATA SEGMENT

  ASCII  DB   '2'

  BCD   DB   00H

DATA ENDS

CODE SEGMENT

  ASSUME CS:CODE, DS:DATA

START:

  MOV AX, DATA

  MOV DS, AX

  MOV AL, ASCII

  SUB AL, 30H

  MOV BL, AL

  AND BL, 0FH

  MOV AH, AL

  MOV AL, BL

  SHL AL, 4

  OR AL, AH

  MOV BCD, AL

  MOV AH, 4CH

  INT 21H

END START
```