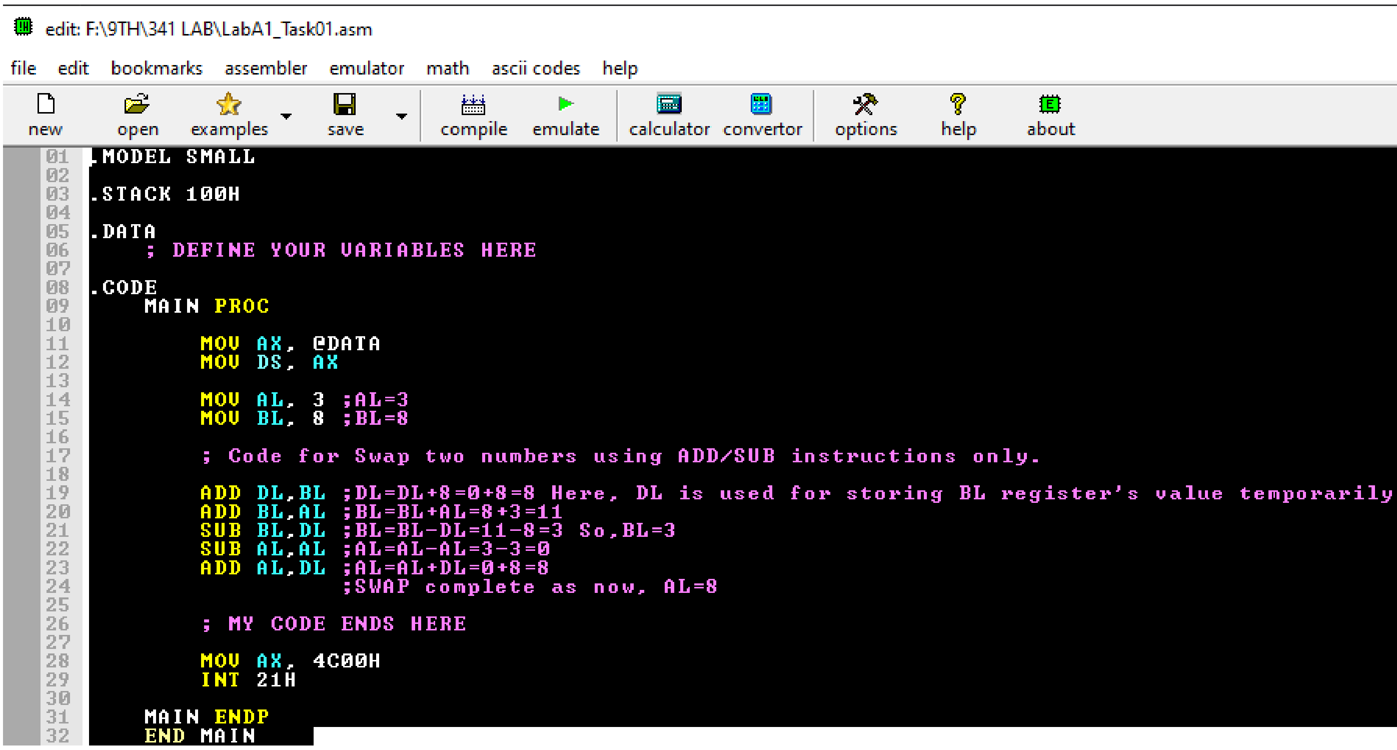
CSE341: Microprocessors
Department of Computer Science and Engineering

Assignment 01

Task 01

Swap two numbers using **ADD/SUB** instructions only.



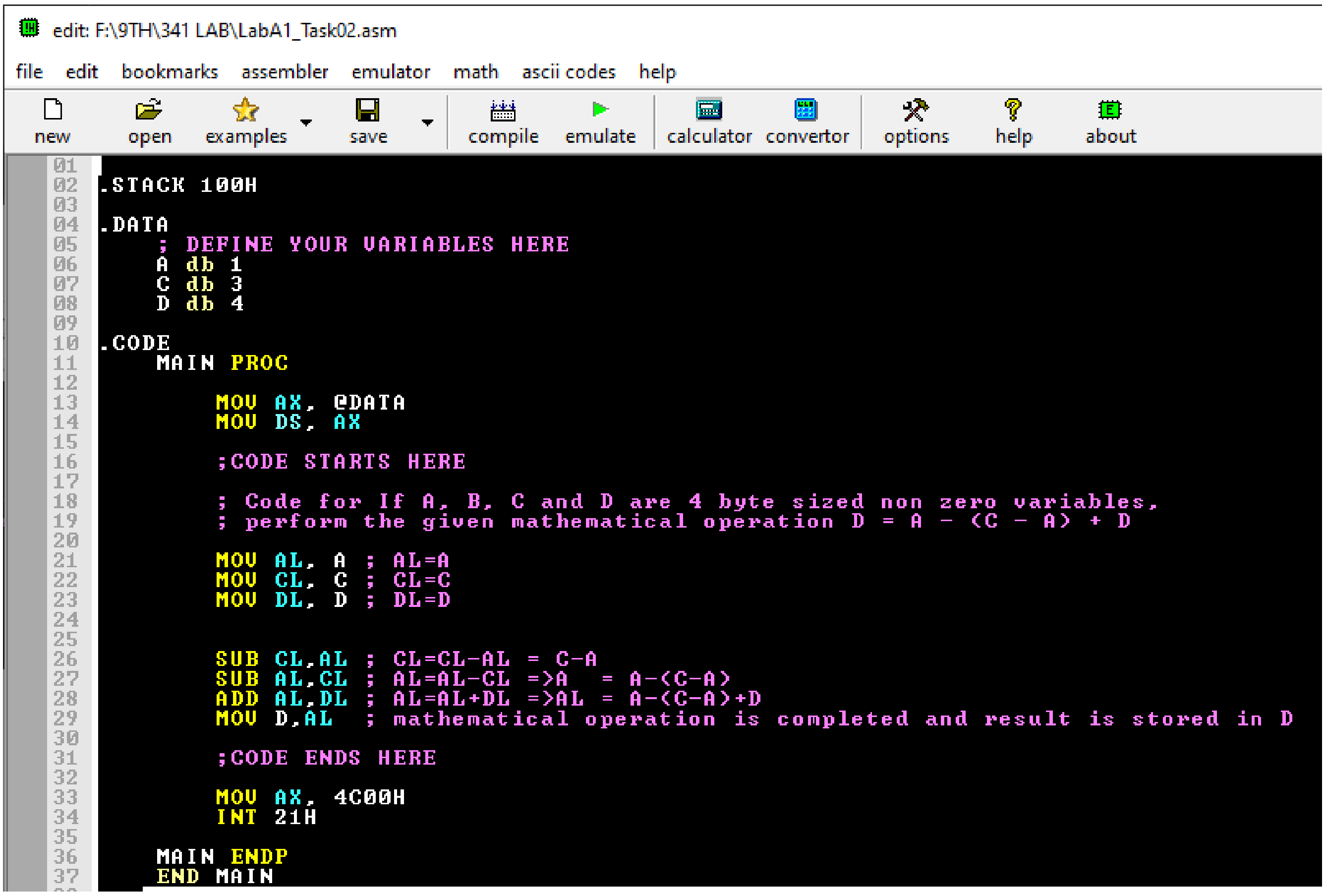
```
.MODEL SMALL

.STACK 100H

.DATA
    ; DEFINE YOUR VARIABLES HERE

.CODE
    MAIN PROC

        MOV AX, @DATA
        MOV DS, AX

        MOV AL, 3 ;AL=3
        MOV BL, 8 ;BL=8

        ; Code for Swap two numbers using ADD/SUB instructions only.

        ADD DL,BL ;DL=DL+8=0+8=8 Here, DL is used for storing BL register's value temporarily
        ADD BL,AL ;BL=BL+AL=8+3=11
        SUB BL,DL ;BL=BL-DL=11-8=3 So,BL=3
        SUB AL,AL ;AL=AL-AL=3-3=0
        ADD AL,DL ;AL=AL+DL=0+8=8
                  ;SWAP complete as now, AL=8

        ; MY CODE ENDS HERE

        MOV AX, 4C00H
        INT 21H

    MAIN ENDP
    END MAIN
```

```
.MODEL SMALL
.STACK 100H
.DATA
   ; DEFINE YOUR VARIABLES HERE

.CODE
   MAIN PROC
      MOV AX, @DATA
      MOV DS, AX

      ;CODE STARTS HERE

      MOV AL, 3 ;AL=3
      MOV BL, 8 ;BL=8

      ; Code for Swap two numbers using ADD/SUB instructions only.

      ADD DL,BL ;DL=DL+8=0+8=8 Here, DL is used for storing BL register's value temporarily
      ADD BL,AL ;BL=BL+AL=8+3=11
      SUB BL,DL ;BL=BL-DL=11-8=3 So,BL=3
      SUB AL,AL ;AL=AL-AL=3-3=0
      ADD AL,DL ;AL=AL+DL=0+8=8
                ;SWAP complete as now, AL=8

      ; MY CODE ENDS HERE

      MOV AX, 4C00H
      INT 21H

   MAIN ENDP
   END MAIN
```

Task 02

If A, B, C and D are 4 byte sized non zero variables, perform the given mathematical operation

D = A - (C - A) + D


edit: F:\9TH\341 LAB\LabA1_Task02.asm
file edit bookmarks assembler emulator math ascii codes help
new
open
examples
save
compile
emulate
calculator
convertor
options
help
about

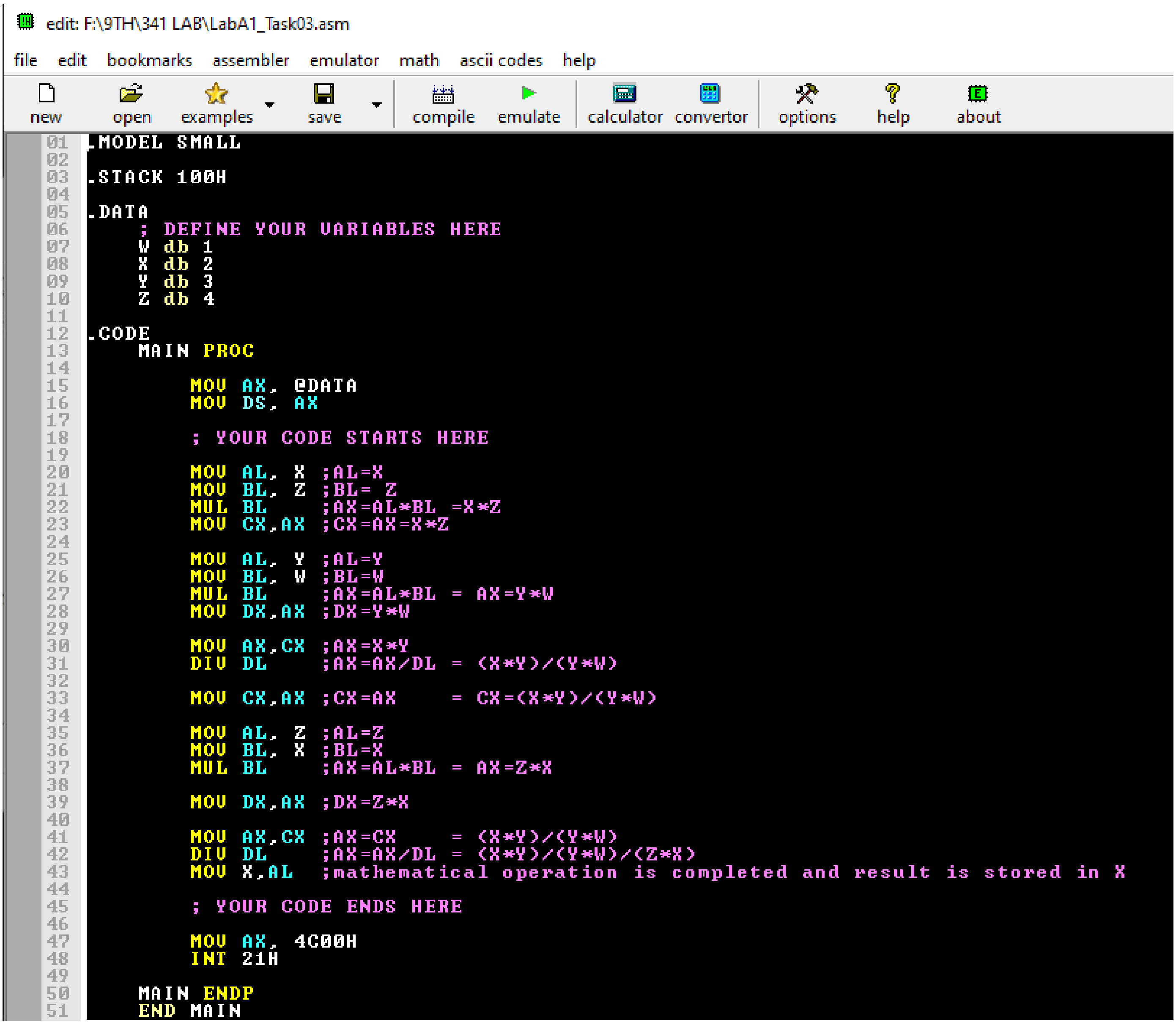
```

.STACK 100H

.DATA
    ; DEFINE YOUR VARIABLES HERE
    A db 1
    C db 3
    D db 4

.CODE
    MAIN PROC

        MOV AX, @DATA
        MOV DS, AX

        ;CODE STARTS HERE

        ; Code for If A, B, C and D are 4 byte sized non zero variables,
        ; perform the given mathematical operation D = A - (C - A) + D

        MOV AL, A ; AL=A
        MOV CL, C ; CL=C
        MOV DL, D ; DL=D


        SUB CL,AL ; CL=CL-AL = C-A
        SUB AL,CL ; AL=AL-CL =>A   = A-(C-A)
        ADD AL,DL ; AL=AL+DL =>AL = A-(C-A)+D
        MOV D,AL  ; mathematical operation is completed and result is stored in D

        ;CODE ENDS HERE

        MOV AX, 4C00H
        INT 21H

    MAIN ENDP
    END MAIN
```

```
.MODEL SMALL
.STACK 100H
 .DATA
   ; DEFINE YOUR VARIABLES HERE
   A db 1
   C db 3
   D db 4
.CODE
   MAIN PROC

      MOV AX, @DATA
      MOV DS, AX

      ;CODE STARTS HERE

      ; Code for If A, B, C and D are 4 byte sized non zero variables,
      ; perform the given mathematical operation D = A - (C - A) + D

      MOV AL, A ; AL=A
      MOV CL, C ; CL=C
      MOV DL, D ; DL=D

      SUB CL,AL ; CL=CL-AL = C-A
      SUB AL,CL ; AL=AL-CL =>A  = A-(C-A)
      ADD AL,DL ; AL=AL+DL =>AL = A-(C-A)+D
      MOV D,AL  ; mathematical operation is completed and result is stored in D


      ;CODE ENDS HERE
      MOV AX, 4C00H
      INT 21H

   MAIN ENDP
   END MAIN
```

Task 03

If W, X, Y and Z are 4 byte sized non zero variables, then perform the given mathematical operation

X = (X * Z) / (Y * W) / (Z * X)

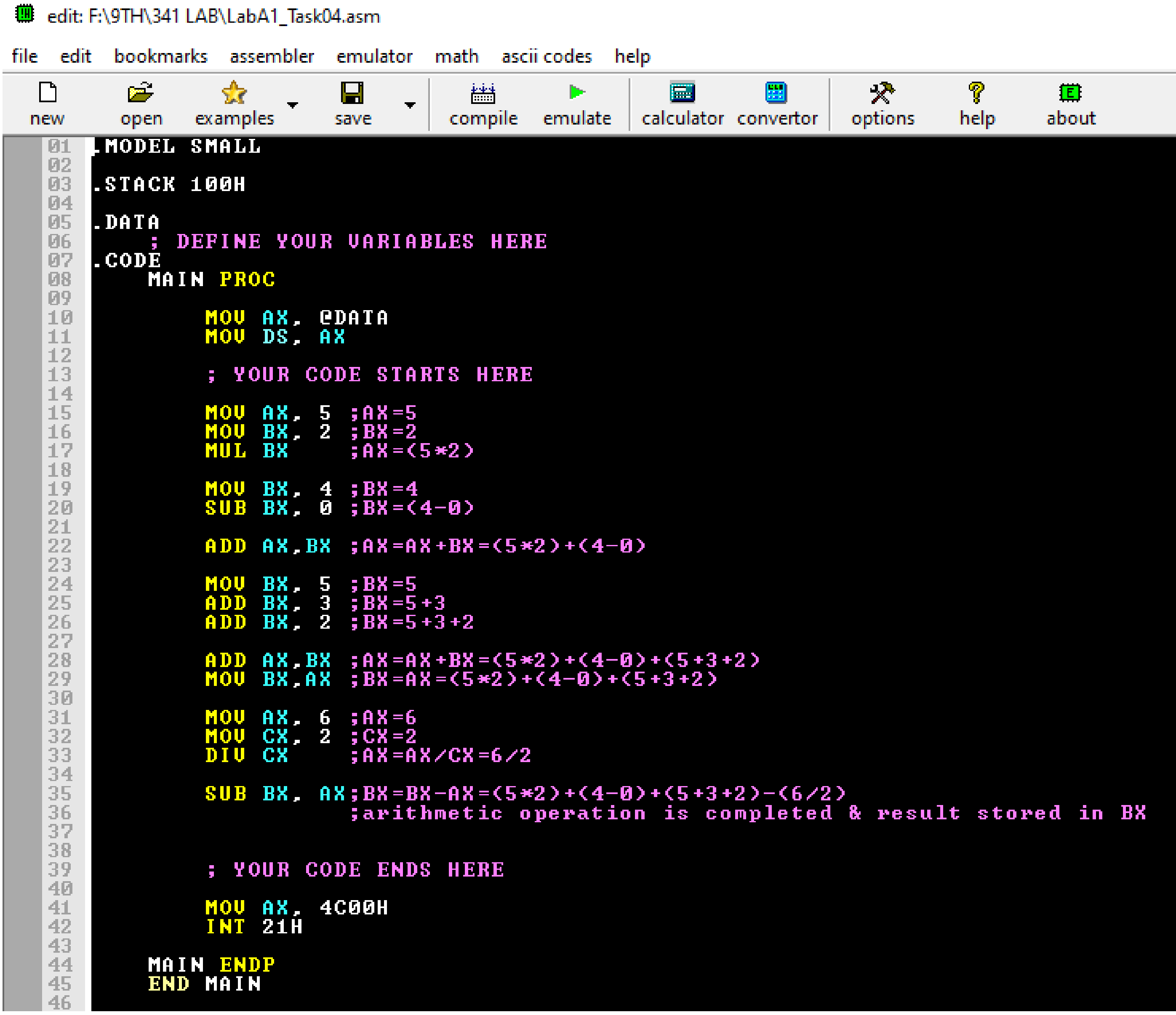


```
.MODEL SMALL

.STACK 100H

.DATA
    ; DEFINE YOUR VARIABLES HERE
    W db 1
    X db 2
    Y db 3
    Z db 4

.CODE
    MAIN PROC

        MOV AX, @DATA
        MOV DS, AX

        ; YOUR CODE STARTS HERE

        MOV AL, X ;AL=X
        MOV BL, Z ;BL= Z
        MUL BL    ;AX=AL*BL  =X*Z
        MOV CX,AX ;CX=AX=X*Z

        MOV AL, Y ;AL=Y
        MOV BL, W ;BL=W
        MUL BL    ;AX=AL*BL  = AX=Y*W
        MOV DX,AX ;DX=Y*W

        MOV AX,CX ;AX=X*Y
        DIV DL    ;AX=AX/DL  = (X*Y)/(Y*W)

        MOV CX,AX ;CX=AX     = CX=(X*Y)/(Y*W)

        MOV AL, Z ;AL=Z
        MOV BL, X ;BL=X
        MUL BL    ;AX=AL*BL  = AX=Z*X

        MOV DX,AX ;DX=Z*X

        MOV AX,CX ;AX=CX     = (X*Y)/(Y*W)
        DIV DL    ;AX=AX/DL  = (X*Y)/(Y*W)/(Z*X)
        MOV X,AL  ;mathematical operation is completed and result is stored in X

        ; YOUR CODE ENDS HERE

        MOV AX, 4C00H
        INT 21H

    MAIN ENDP
    END MAIN
```

```
.MODEL SMALL
.STACK 100H
.DATA
   ; DEFINE YOUR VARIABLES HERE
   W db 1
   X db 2
   Y db 3
   Z db 4

.CODE
   MAIN PROC

      MOV AX, @DATA
      MOV DS, AX
       ; YOUR CODE STARTS HERE

      MOV AL, X ;AL=X
      MOV BL, Z ;BL= Z
      MUL BL    ;AX=AL*BL =X*Z
      MOV CX,AX ;CX=AX=X*Z

      MOV AL, Y ;AL=Y
      MOV BL, W ;BL=W
      MUL BL    ;AX=AL*BL = AX=Y*W
      MOV DX,AX ;DX=Y*W

      MOV AX,CX ;AX=X*Y
      DIV DL    ;AX=AX/DL = (X*Y)/(Y*W)

      MOV CX,AX ;CX=AX    = CX=(X*Y)/(Y*W)

      MOV AL, Z ;AL=Z
      MOV BL, X ;BL=X
      MUL BL    ;AX=AL*BL = AX=Z*X

      MOV DX,AX ;DX=Z*X

      MOV AX,CX ;AX=CX    = (X*Y)/(Y*W)
      DIV DL    ;AX=AX/DL = (X*Y)/(Y*W)/(Z*X)
      MOV X,AL  ;mathematical operation is completed and result is stored in X

      ; YOUR CODE ENDS HERE
      MOV AX, 4C00H
      INT 21H

   MAIN ENDP
   END MAIN
```

Task 04

Perform the following arithmetic operation: **(5 * 2) + (4 – 0) + (5 + 3 + 2) – (6 / 2)**



```
.MODEL SMALL

.STACK 100H

.DATA
    ; DEFINE YOUR VARIABLES HERE
.CODE
    MAIN PROC

        MOV AX, @DATA
        MOV DS, AX

        ; YOUR CODE STARTS HERE

        MOV AX, 5 ;AX=5
        MOV BX, 2 ;BX=2
        MUL BX    ;AX=(5*2)

        MOV BX, 4 ;BX=4
        SUB BX, 0 ;BX=(4-0)

        ADD AX,BX ;AX=AX+BX=(5*2)+(4-0)

        MOV BX, 5 ;BX=5
        ADD BX, 3 ;BX=5+3
        ADD BX, 2 ;BX=5+3+2

        ADD AX,BX ;AX=AX+BX=(5*2)+(4-0)+(5+3+2)
        MOV BX,AX ;BX=AX=(5*2)+(4-0)+(5+3+2)

        MOV AX, 6 ;AX=6
        MOV CX, 2 ;CX=2
        DIV CX    ;AX=AX/CX=6/2

        SUB BX, AX;BX=BX-AX=(5*2)+(4-0)+(5+3+2)-(6/2)
                  ;arithmetic operation is completed & result stored in BX


        ; YOUR CODE ENDS HERE

        MOV AX, 4C00H
        INT 21H

    MAIN ENDP
    END MAIN

```

```
.MODEL SMALL
.STACK 100H
.DATA
   ; DEFINE YOUR VARIABLES HERE
.CODE
   MAIN PROC
      MOV AX, @DATA
      MOV DS, AX
       ; YOUR CODE STARTS HERE

      MOV AX, 5 ;AX=5
      MOV BX, 2 ;BX=2
      MUL BX    ;AX=(5*2)
      MOV BX, 4 ;BX=4
      SUB BX, 0 ;BX=(4-0)
       ADD AX,BX ;AX=AX+BX=(5*2)+(4-0)
      MOV BX, 5 ;BX=5
      ADD BX, 3 ;BX=5+3
      ADD BX, 2 ;BX=5+3+2
      ADD AX,BX ;AX=AX+BX=(5*2)+(4-0)+(5+3+2)
      MOV BX,AX ;BX=AX=(5*2)+(4-0)+(5+3+2)
      MOV AX, 6 ;AX=6
      MOV CX, 2 ;CX=2
      DIV CX    ;AX=AX/CX=6/2

      SUB BX, AX;BX=BX-AX=(5*2)+(4-0)+(5+3+2)-(6/2)
             ;arithmetic operation is completed & result stored in BX
      ; YOUR CODE ENDS HERE
      MOV AX, 4C00H
      INT 21H

   MAIN ENDP
   END MAIN
```