UCS1512 Microprocessors Lab

Experiment 2: 16-bit Arithmetic Operations

Date: 27-08-2020

Name: Nivedhitha D

Register Number: 185001104

Aim

To write programs for performing 16-bit arithmetic operations in an 8086 microprocessor using MASM and DOSBox.

Algorithm

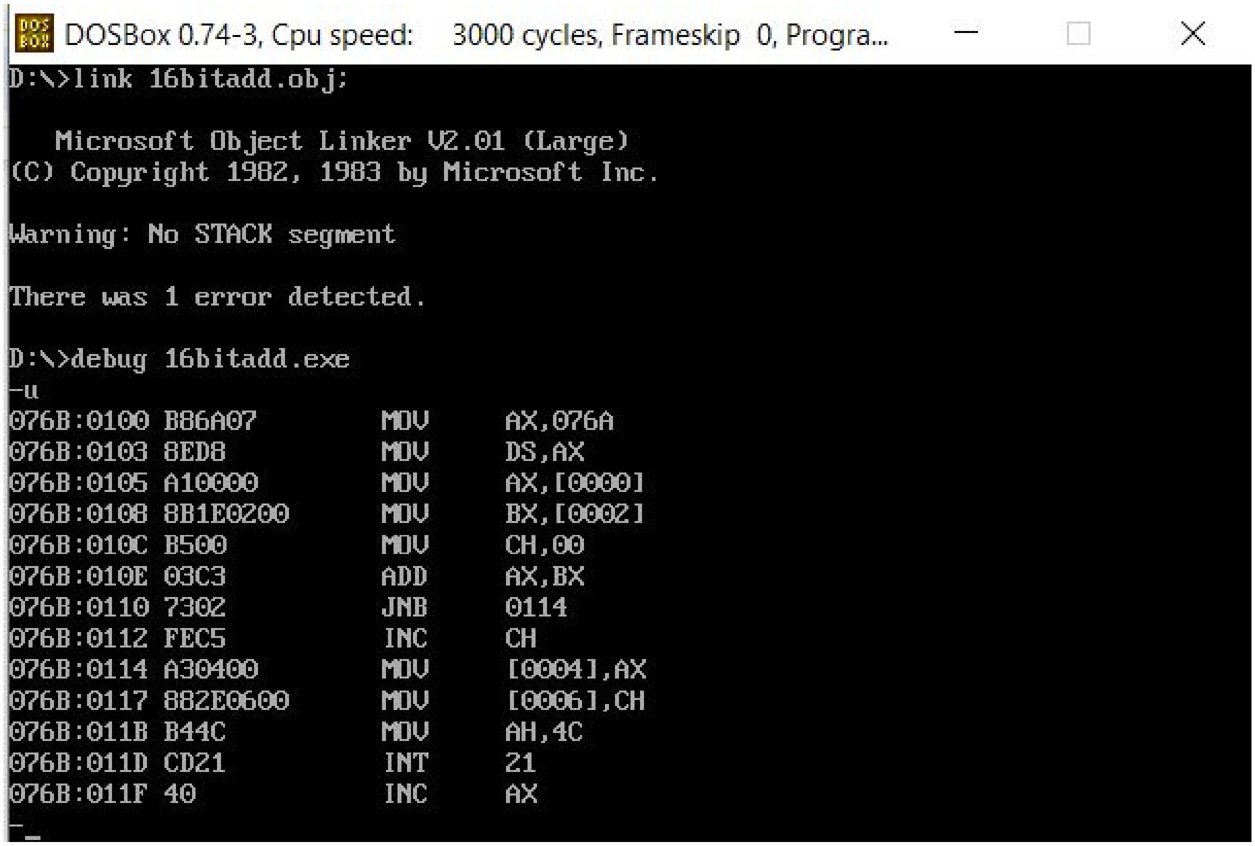
1. Define the values in the data segment and assign the initial values if required
2. Initialize the data segment register with a data segment address
3. Load the data into the appropriate registers and perform addition/ subtraction/ multiplication/ division and store the sum/ difference/product/quotient-remainder to the result address for display
4. Terminate the program

Program for adding two 16-bit numbers

Program Name: 16BITADD.ASM

Program	Comments
;Program for adding two 16-bit numbers ASSUME CS:CODE,DS:DATA	Naming the CS and DS for the program
DATA SEGMENT OPR1 DW 1111H OPR2 DW 2219H	Declaring and initializing the values for the two operands in the data segment
RESULT DW 0000H CARRY DB 00H DATA ENDS	The carry and result must be set to zero to obtain the correct result and get rid off garbage values
CODE SEGMENT ORG 0100H	Providing an offset value
START: MOV AX, DATA MOV DS, AX	Initializing the data segment register with the data segment address
MOV AX, OPR1 MOV BX, OPR2 MOV CH, 00H	Loading the data into the appropriate registers
ADD AX, BX	Performing addition: AX = AX + BX
JNC HERE	If the carry is zero, go-to label, HERE
INC CH	If carry has occurred, increment register, CH
HERE: MOV RESULT, AX MOV CARRY, CH	Loading the sum and carry into the appropriate locations for display
MOV AH, 4CH INT 21H CODE ENDS END START	Calling the DOS Function to enter the display screen using interrupt 21H and to terminate the program

Unassembled Code


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
D:\>link 16bitadd.obj;

   Microsoft Object Linker V2.01 (Large)
(C) Copyright 1982, 1983 by Microsoft Inc.

Warning: No STACK segment

There was 1 error detected.

D:\>debug 16bitadd.exe
-u
076B:0100 B86A07        MOV     AX,076A
076B:0103 8ED8          MOV     DS,AX
076B:0105 A10000        MOV     AX,[0000]
076B:0108 8B1E0200      MOV     BX,[0002]
076B:010C B500          MOV     CH,00
076B:010E 03C3          ADD     AX,BX
076B:0110 7302          JNB     0114
076B:0112 FEC5          INC     CH
076B:0114 A30400        MOV     [0004],AX
076B:0117 882E0600      MOV     [0006],CH
076B:011B B44C          MOV     AH,4C
076B:011D CD21          INT     21
076B:011F 40            INC     AX
-_
```

Snapshot of Input and Output

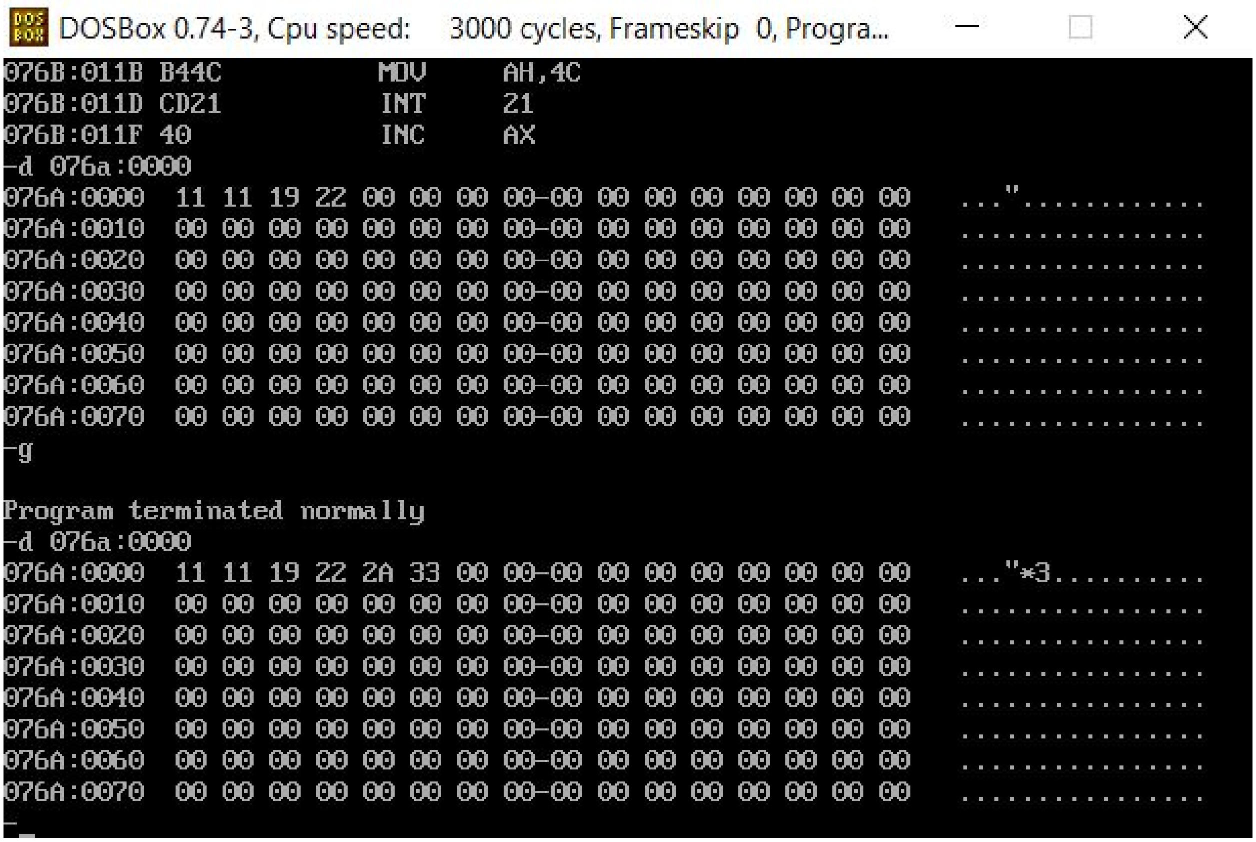
Input

Two 16-bit values

Output

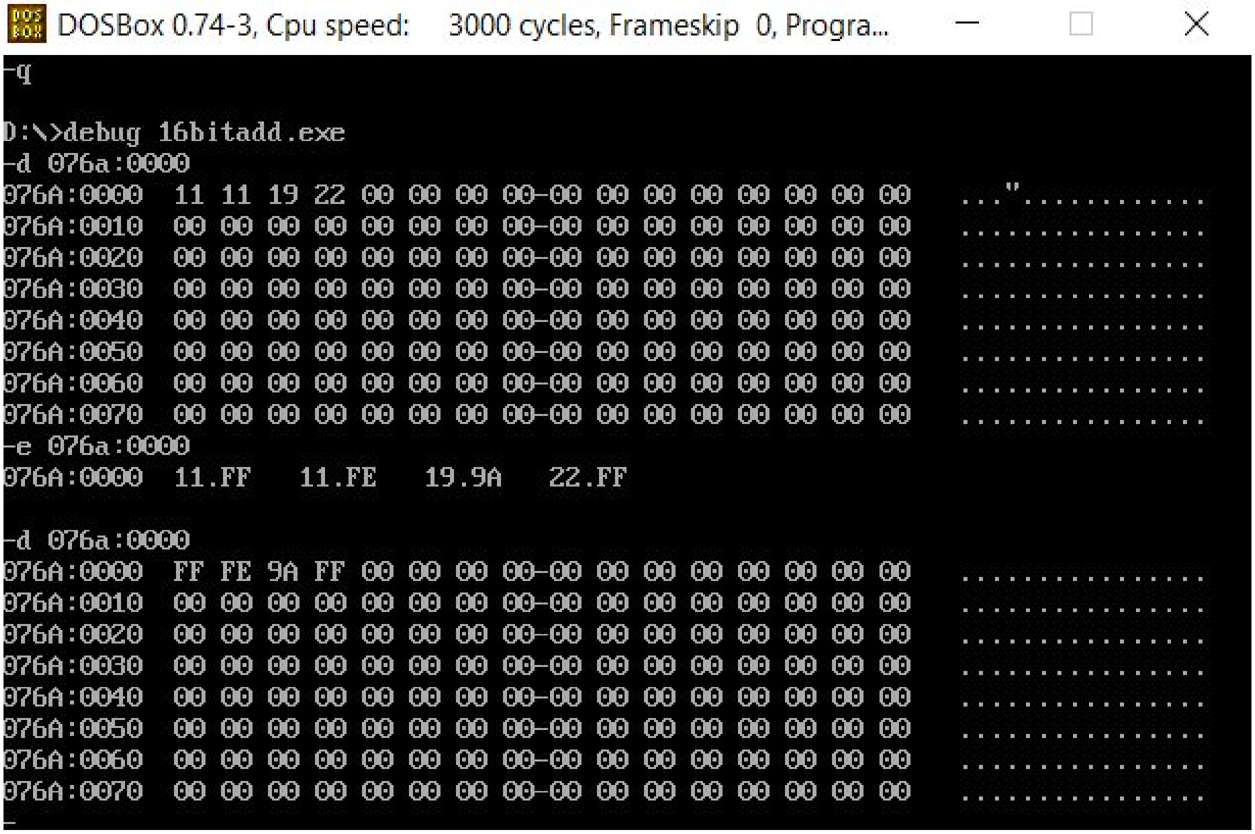
Sum in two memory locations and the carry in another location

Without carry

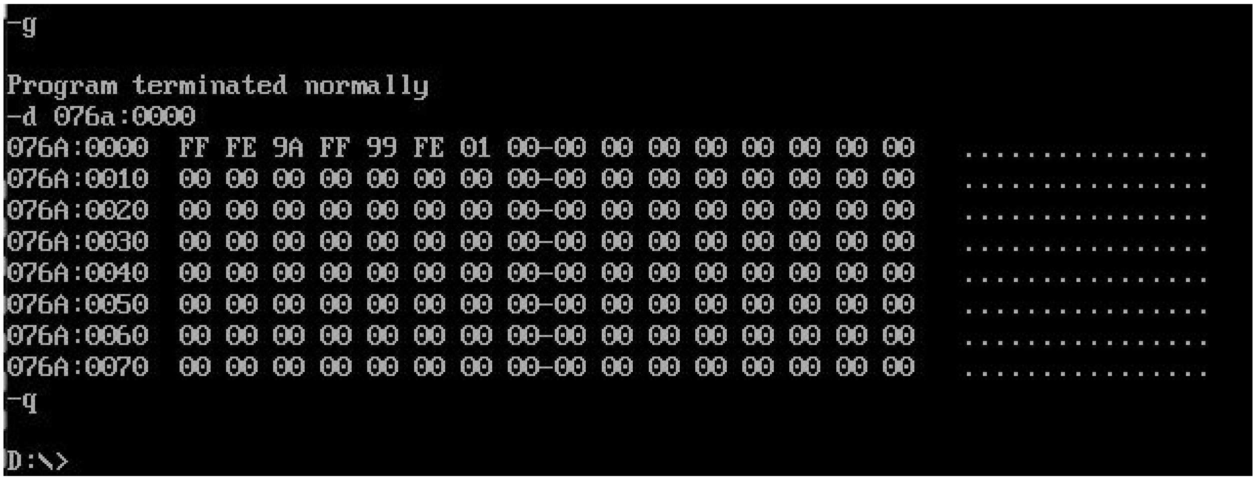

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
076B:011B B44C          MOV     AH,4C
076B:011D CD21          INT     21
076B:011F 40            INC     AX
-d 076a:0000
076A:0000  11 11 19 22 00 00 00 00-00 00 00 00 00 00 00 00   ...".............
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-g

Program terminated normally
-d 076a:0000
076A:0000  11 11 19 22 2A 33 00 00-00 00 00 00 00 00 00 00   ..."*3..........
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-
```

With carry


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
-q

D:\>debug 16bitadd.exe
-d 076a:0000
076A:0000  11 11 19 22 00 00 00 00-00 00 00 00 00 00 00 00   ..."............
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-e 076a:0000
076A:0000  11.FF   11.FE   19.9A   22.FF

-d 076a:0000
076A:0000  FF FE 9A FF 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-
```



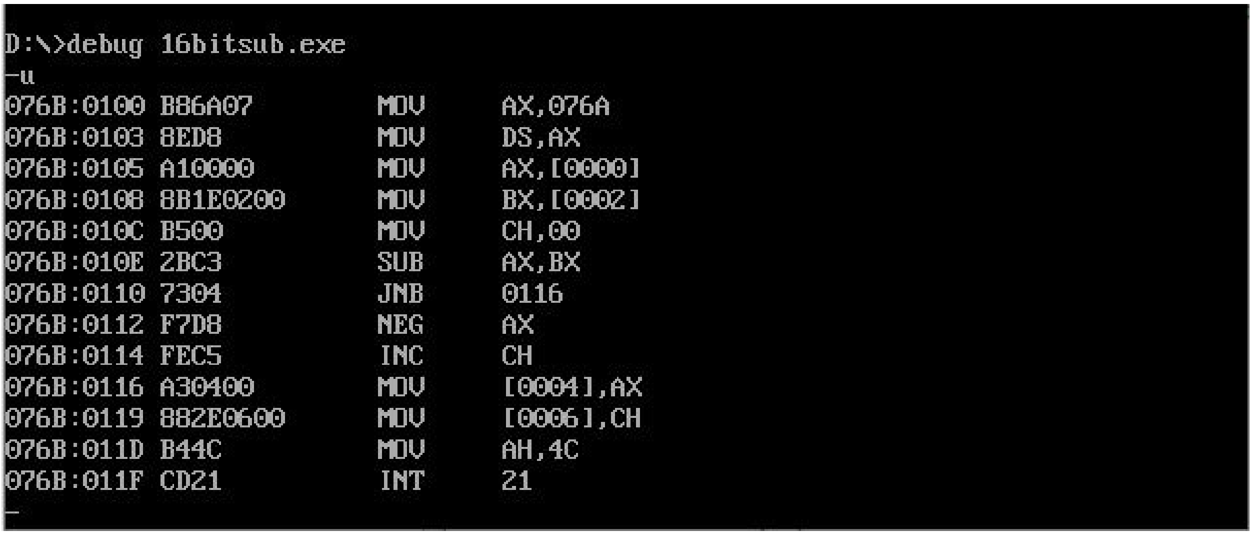
```
-g

Program terminated normally
-d 076a:0000
076A:0000  FF FE 9A FF 99 FE 01 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-q

D:\>
```

Program for subtracting two 16-bit numbers

Program Name: 16BITSUB.ASM

Program	**Comments**
;Program for subtracting two 16-bit numbers ASSUME CS:CODE,DS:DATA	Naming the CS and DS for the program
DATA SEGMENT OPR1 DW 1211H OPR2 DW 3599H	Declaring and initializing the values for the two operands in the data segment
DIFF DW 0000H SIGN DB 00H DATA ENDS	The difference and sign must be set to zero to obtain the correct result and get rid off garbage values
CODE SEGMENT ORG 0100H	Providing an offset value
START: MOV AX, DATA MOV DS, AX	Initializing the data segment register with the data segment address
MOV AX, OPR1 MOV BX, OPR2 MOV CH, 00H	Loading the data into the appropriate registers
SUB AX, BX	Performing subtraction: AX = AX - BX
JNC HERE	If the carry is zero, go-to label, HERE
NEG AX INC CH	If carry has occurred, increment register, CH to denote sign & 2’s complement of difference
HERE: MOV DIFF, AX MOV SIGN, CH	Loading the difference and sign into the appropriate locations for display
MOV AH, 4CH INT 21H CODE ENDS END START	Calling the DOS Function to enter the display screen using interrupt 21H and to terminate the program

Unassembled Code



```
D:\>debug 16bitsub.exe
-u
076B:0100 B86A07        MOV     AX,076A
076B:0103 8ED8          MOV     DS,AX
076B:0105 A10000        MOV     AX,[0000]
076B:0108 8B1E0200      MOV     BX,[0002]
076B:010C B500          MOV     CH,00
076B:010E 2BC3          SUB     AX,BX
076B:0110 7304          JNB     0116
076B:0112 F7D8          NEG     AX
076B:0114 FEC5          INC     CH
076B:0116 A30400        MOV     [0004],AX
076B:0119 882E0600      MOV     [0006],CH
076B:011D B44C          MOV     AH,4C
076B:011F CD21          INT     21
-
```

Snapshot of Input and Output

Input

Two 16-bit values

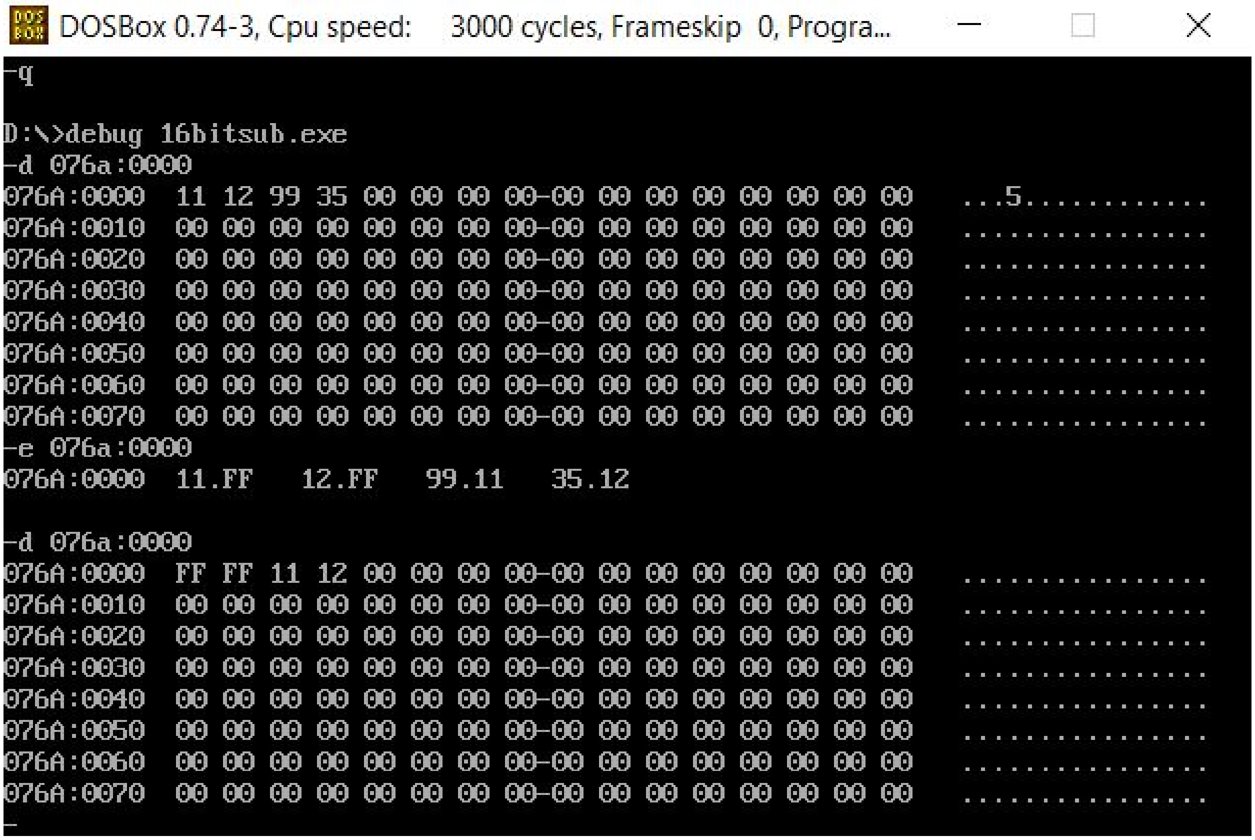
Output

Difference in two memory locations and the indication of the sign in one

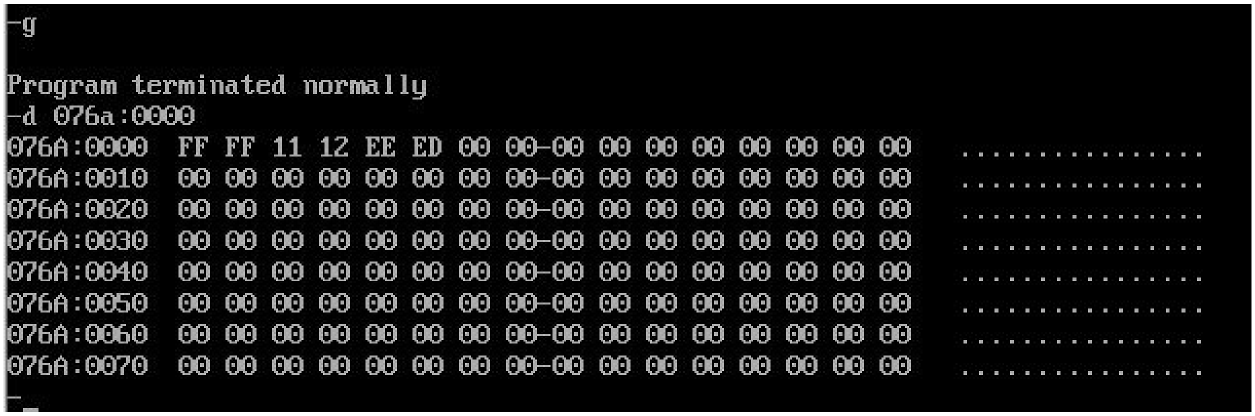
(eg: FF-FE = 01, indication of sign is 00 i.e. positive

FE – FF = 01, indication of sign is 01 i.e. negative)

Positive Sign

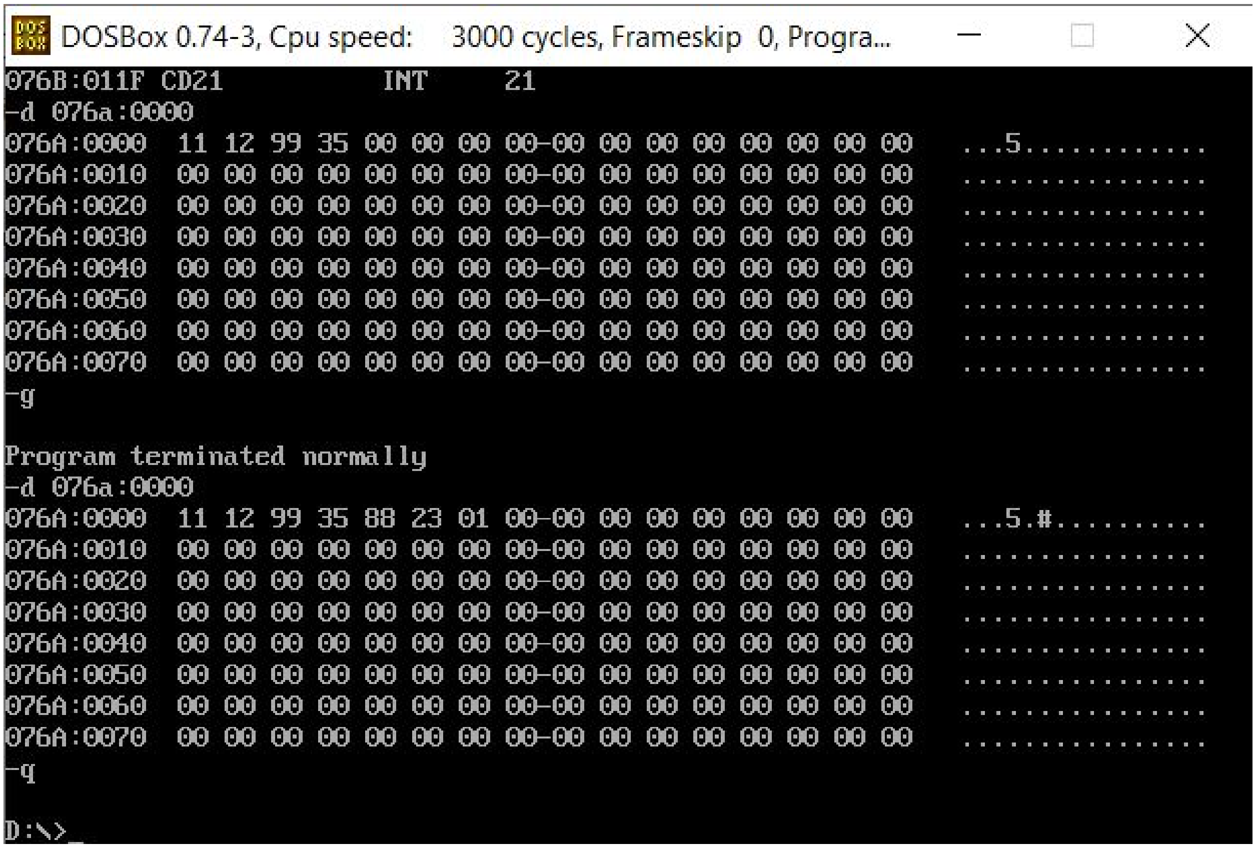

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
-q

D:\>debug 16bitsub.exe
-d 076a:0000
076A:0000  11 12 99 35 00 00 00 00-00 00 00 00 00 00 00 00   ...5............
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-e 076a:0000
076A:0000  11.FF   12.FF   99.11   35.12

-d 076a:0000
076A:0000  FF FF 11 12 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-
```



```
-g

Program terminated normally
-d 076a:0000
076A:0000  FF FF 11 12 EE ED 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-_
```

Negative Sign


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

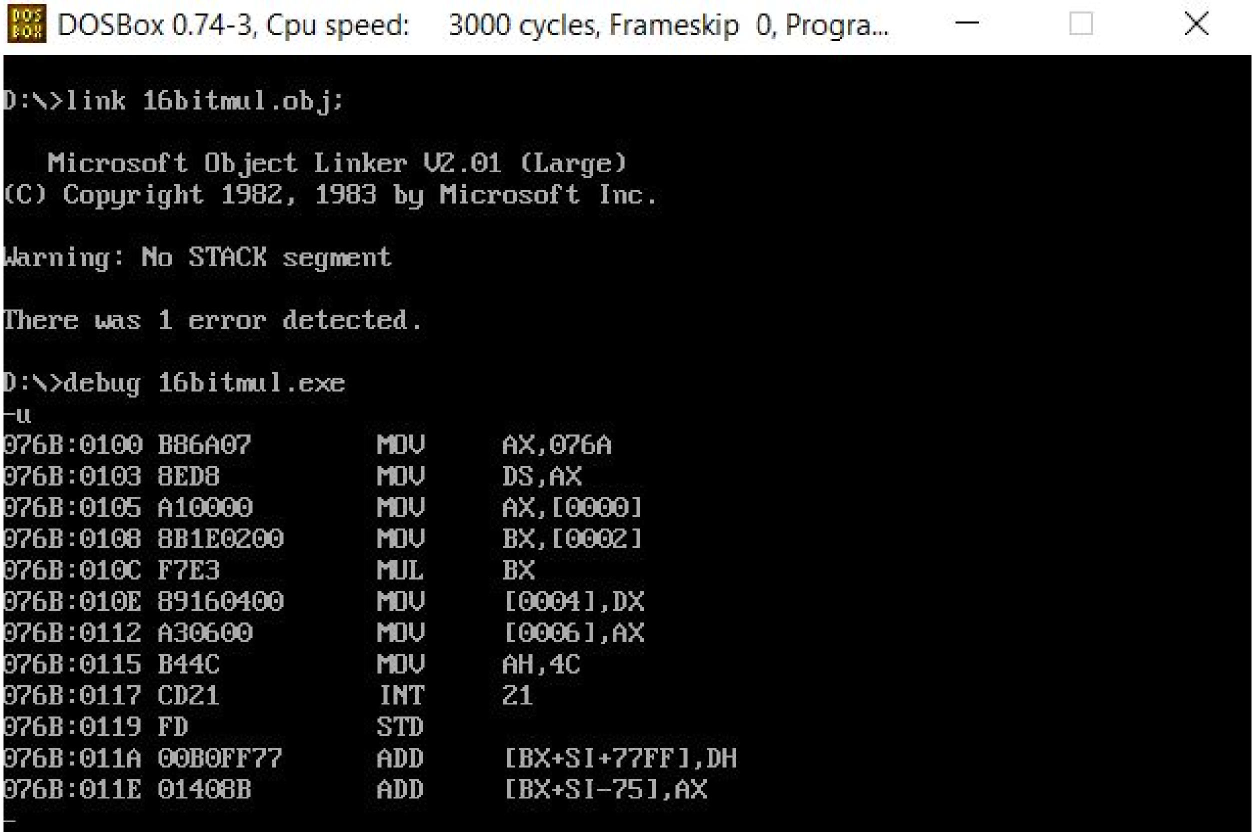
```
076B:011F CD21          INT     21
-d 076a:0000
076A:0000  11 12 99 35 00 00 00 00-00 00 00 00 00 00 00 00   ...5............
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-g

Program terminated normally
-d 076a:0000
076A:0000  11 12 99 35 88 23 01 00-00 00 00 00 00 00 00 00   ...5.#..........
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-q

D:\>_
```

Program for multiplying two 16-bit numbers

Program Name: 16BITMUL.ASM

Program	Comments
;Program for multiplying two 16-bit numbers ASSUME CS:CODE,DS:DATA	Naming the CS and DS for the program
DATA SEGMENT OPR1 DW 1111H OPR2 DW 4444H	Declaring and initializing the values for the two operands in the data segment
PRODH DW 0000H PRODL DW 0000H DATA ENDS	The higher and lower order bits of the product must be set to zero to obtain the correct result and get rid off garbage values
CODE SEGMENT ORG 0100H	Providing an offset value
START: MOV AX, DATA MOV DS, AX	Initializing the data segment register with the data segment address
MOV AX, OPR1 MOV BX, OPR2	Loading the data into the appropriate registers
MUL BX	Performing multiplication: DXAX = AX * BX
MOV PRODH, DX MOV PRODL, AX	Since the product is stored in the DX & AX register, the higher and lower order bits are extracted
MOV AH, 4CH INT 21H CODE ENDS END START	Calling the DOS Function to enter the display screen using interrupt 21H and to terminate the program

Unassembled Code


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
D:\>link 16bitmul.obj;

   Microsoft Object Linker V2.01 (Large)
(C) Copyright 1982, 1983 by Microsoft Inc.

Warning: No STACK segment

There was 1 error detected.

D:\>debug 16bitmul.exe
-u
076B:0100 B86A07        MOV     AX,076A
076B:0103 8ED8          MOV     DS,AX
076B:0105 A10000        MOV     AX,[0000]
076B:0108 8B1E0200      MOV     BX,[0002]
076B:010C F7E3          MUL     BX
076B:010E 89160400      MOV     [0004],DX
076B:0112 A30600        MOV     [0006],AX
076B:0115 B44C          MOV     AH,4C
076B:0117 CD21          INT     21
076B:0119 FD            STD
076B:011A 00B0FF77      ADD     [BX+SI+77FF],DH
076B:011E 01408B        ADD     [BX+SI-75],AX
-
```

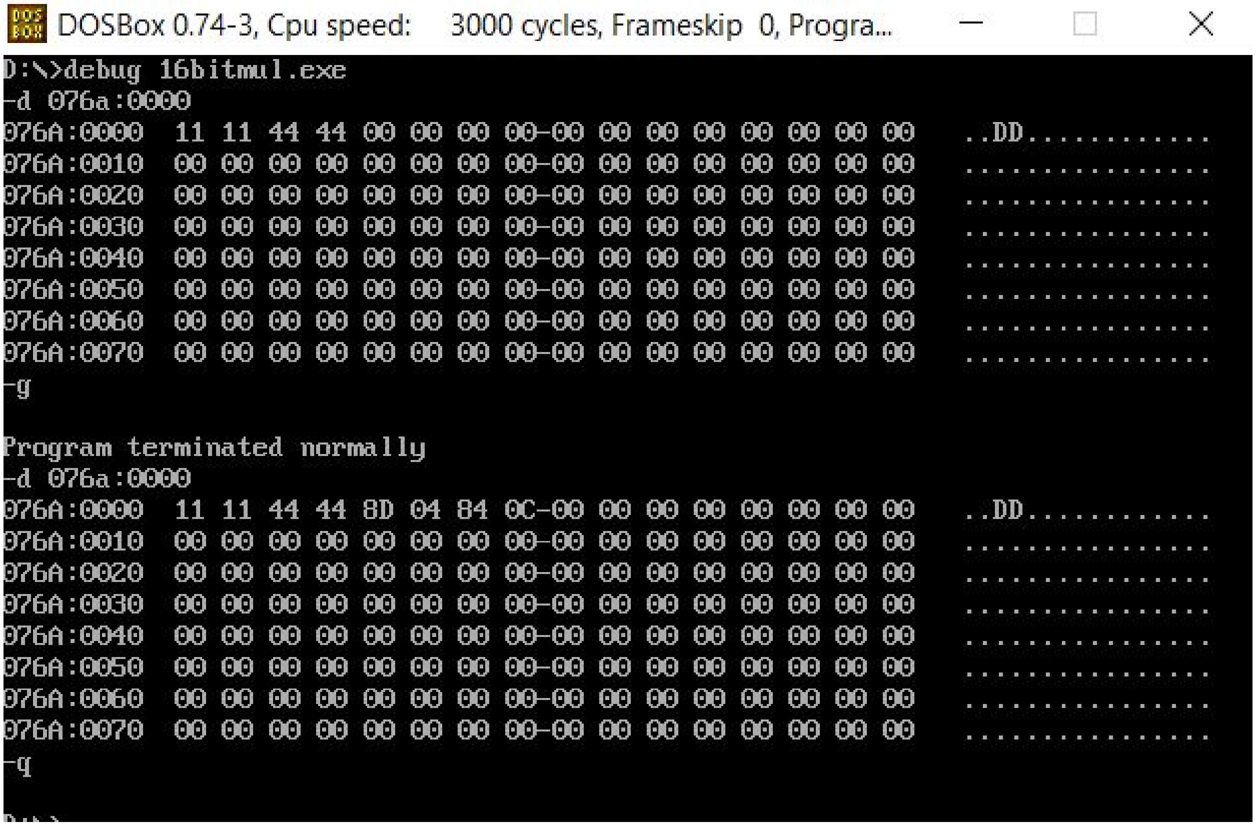
Snapshot of Input and Output

Input

Two 16-bit values

Output

Product in 32 bits


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

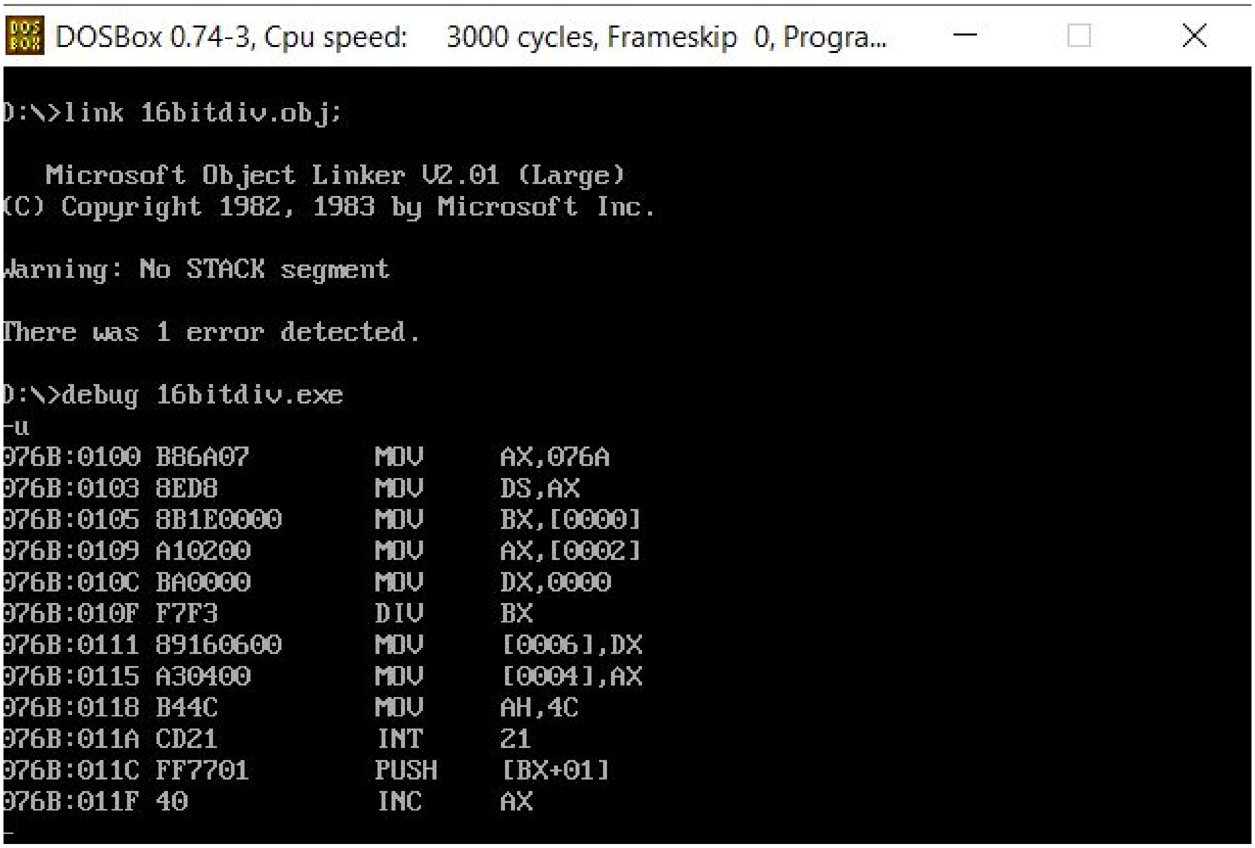
```
D:\>debug 16bitmul.exe
-d 076a:0000
076A:0000  11 11 44 44 00 00 00 00-00 00 00 00 00 00 00 00   ..DD............
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-g

Program terminated normally
-d 076a:0000
076A:0000  11 11 44 44 8D 04 84 0C-00 00 00 00 00 00 00 00   ..DD............
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-q

D:\>
```

Program for dividing two 16-bit numbers

Program Name: 16BITDIV.ASM

Program	**Comments**
;Program for dividing two 16-bit numbers ASSUME CS:CODE,DS:DATA	Naming the CS and DS for the program
DATA SEGMENT OPR1 DW 1111H OPR2 DW 9999H	Declaring and initializing the values for the two operands in the data segment
QUOTIENT DW 0000H REMINDER DW 0000H DATA ENDS	The quotient and remainder must be set to zero to obtain the correct result and get rid off garbage values
CODE SEGMENT ORG 0100H	Providing an offset value
START: MOV AX, DATA MOV DS, AX	Initializing the data segment register with the data segment address
MOV BX, OPR1 MOV AX, OPR2	Loading the data into the appropriate registers
MOV DX, 0000H	Setting the DX bits to zero
DIV BX	Performing division: AX = DXAX / BX
MOV REMINDER, DX MOV QUOTIENT, AX	Since the quotient and remainder are stored in the AX register and DX register respectively, they need to be extracted
MOV AH, 4CH INT 21H CODE ENDS END START	Calling the DOS Function to enter the display screen using interrupt 21H

Unassembled Code


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
:\>link 16bitdiv.obj;

   Microsoft Object Linker V2.01 (Large)
C) Copyright 1982, 1983 by Microsoft Inc.

arning: No STACK segment

here was 1 error detected.

:\>debug 16bitdiv.exe
u
76B:0100 B86A07         MOV     AX,076A
76B:0103 8ED8           MOV     DS,AX
76B:0105 8B1E0000       MOV     BX,[0000]
76B:0109 A10200         MOV     AX,[0002]
76B:010C BA0000         MOV     DX,0000
76B:010F F7F3           DIV     BX
76B:0111 89160600       MOV     [0006],DX
76B:0115 A30400         MOV     [0004],AX
76B:0118 B44C           MOV     AH,4C
76B:011A CD21           INT     21
76B:011C FF7701         PUSH    [BX+01]
76B:011F 40             INC     AX
-
```

Snapshot of Input and Output

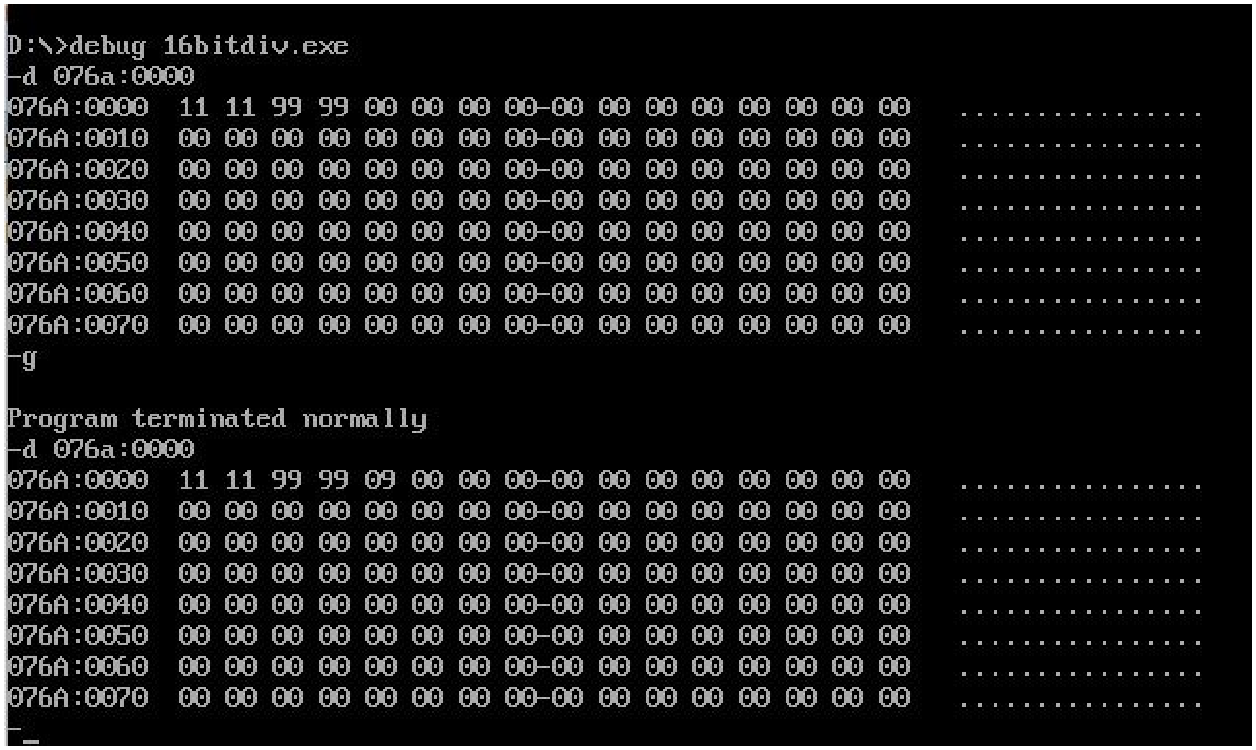
Input

Two 16-bit values(**Note:** No dedicated instruction available in 8086 to perform 16 bit / 16 bit)

Output

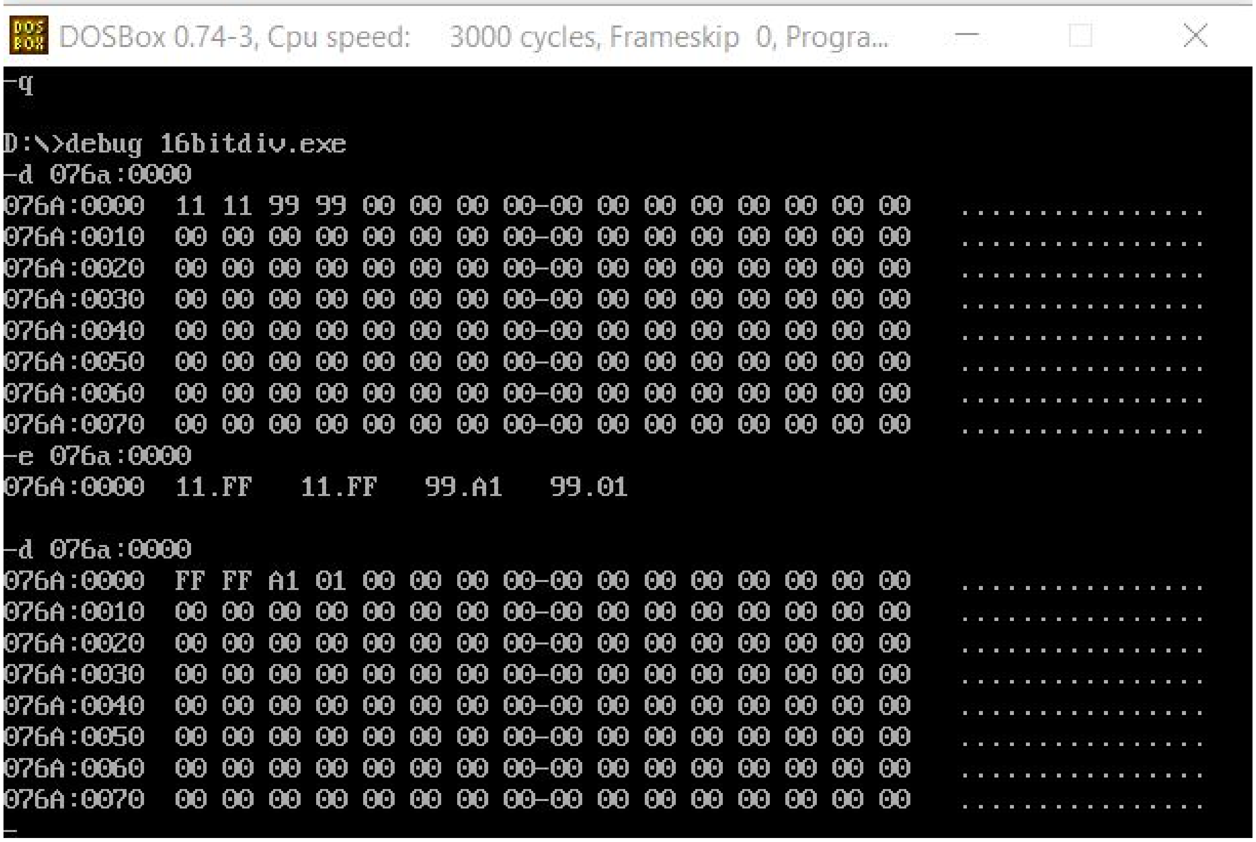
Quotient in two memory locations and the remainder in another two

Without reminder

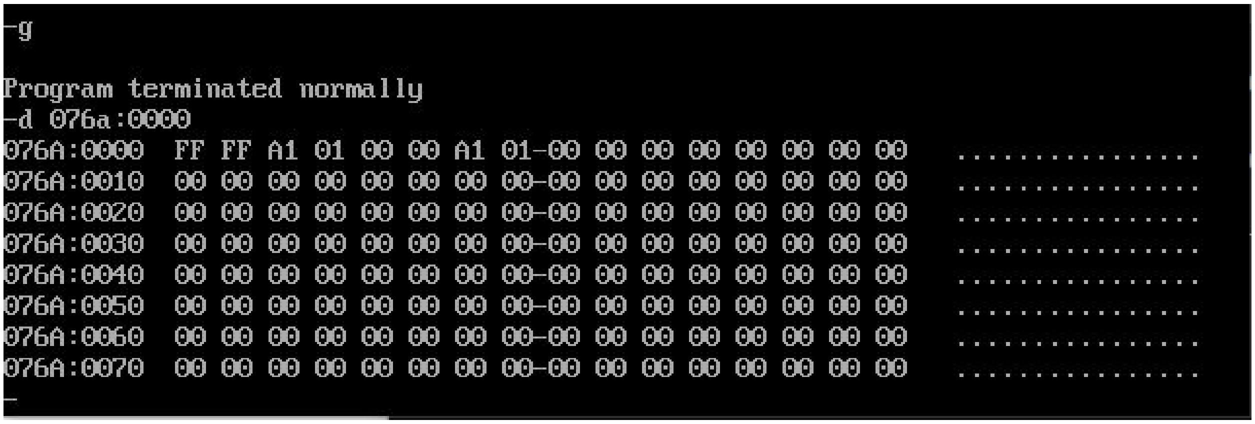


```
D:\>debug 16bitdiv.exe
-d 076a:0000
076A:0000  11 11 99 99 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-g

Program terminated normally
-d 076a:0000
076A:0000  11 11 99 99 09 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-_
```

With reminder


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
-q

D:\>debug 16bitdiv.exe
-d 076a:0000
076A:0000  11 11 99 99 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-e 076a:0000
076A:0000  11.FF   11.FF   99.A1   99.01

-d 076a:0000
076A:0000  FF FF A1 01 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-
```



```
-g

Program terminated normally
-d 076a:0000
076A:0000  FF FF A1 01 00 00 A1 01-00 00 00 00 00 00 00 00   ................
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0060  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
076A:0070  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00   ................
-
```

Result

Programs for performing 16-bit arithmetic operations in an 8086 microprocessor using MASM and DOSBox were implemented and the outputs were verified.