BCD Addition and Subtraction

Expt No:	7	**Name:**	Shivanirudh S G
Date :	09/10/2020	**Reg No:**	185001146

Aim:

To perform BCD addition and subtraction operations in 8086.

BCD Addition

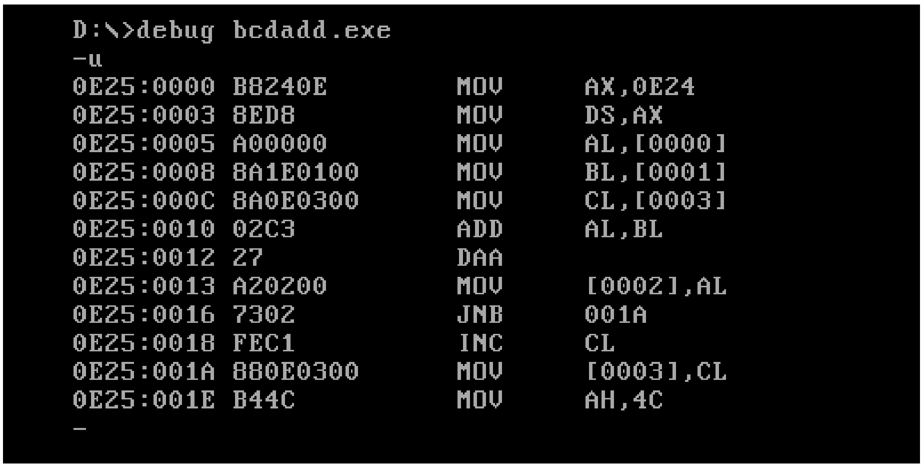
Algorithm:

- Move the data segment to the AX register and then move it to the DS register.
- Move value of num1 to AL, num2 to BL, carry to CL registers.
- Add AL and BL using ADD AL, BL.
- Perform Decimal Adjust After Addition using DAA instruction.
- Move value of AL to ans.
- Jump to label HERE if no carry.
- Increment value of CL.
- Move value of CL to carry, under label HERE.

Program:

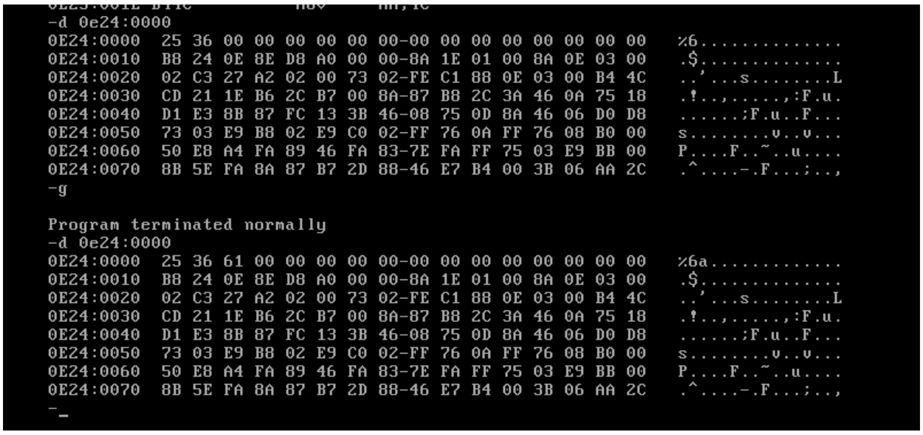
Program	**Comments**
assume cs:code, ds:data	Declare code and data segments
data segment	Start of data segment
num1 db 25H	Define byte num1 with value 25
num2 db 36H	Define byte num2 with value 36
ans db ?	Define byte ans for result
carry db 00H	Define byte carry with value 00
data ends	End of data segment
code segment	Start of code segment
start: mov ax, data	Move data to AX register
mov ds, ax	Move contents of AX register to DS register
mov al, num1	Move value of num1 to AL register
mov bl, num2	Move value of num2 to BL register
mov cl, carry	Move value of carry to CL register
add al, bl	AL = AL + BL
daa	Decimal Adjust after Addition
mov ans, al	Move value of AL register into ans
jnc here	Jump to label HERE if no carry
inc cl	Increment value of CL
here: mov carry, cl	Move value of CL register into carry
mov ah, 4ch	To request interrupt
int 21h	Request interrupt routine
code ends	End of code segment
end start	

Unassembled code:



```
D:\>debug bcdadd.exe
-u
0E25:0000 B8240E        MOV     AX,0E24
0E25:0003 8ED8          MOV     DS,AX
0E25:0005 A00000        MOV     AL,[0000]
0E25:0008 8A1E0100      MOV     BL,[0001]
0E25:000C 8A0E0300      MOV     CL,[0003]
0E25:0010 02C3          ADD     AL,BL
0E25:0012 27            DAA
0E25:0013 A20200        MOV     [0002],AL
0E25:0016 7302          JNB     001A
0E25:0018 FEC1          INC     CL
0E25:001A 880E0300      MOV     [0003],CL
0E25:001E B44C          MOV     AH,4C
-
```

Input and Output:



```
0E25:001E B44C          MOV     AH,4C
-d 0e24:0000
0E24:0000  25 36 00 00 00 00 00 00-00 00 00 00 00 00 00 00   %6..............
0E24:0010  B8 24 0E 8E D8 A0 00 00-8A 1E 01 00 8A 0E 03 00   .$..............
0E24:0020  02 C3 27 A2 02 00 73 02-FE C1 88 0E 03 00 B4 4C   ..'...s........L
0E24:0030  CD 21 1E B6 2C B7 00 8A-87 B8 2C 3A 46 0A 75 18   .!..,.....,:F.u.
0E24:0040  D1 E3 8B 87 FC 13 3B 46-08 75 0D 8A 46 06 D0 D8   ......;F.u..F...
0E24:0050  73 03 E9 B8 02 E9 C0 02-FF 76 0A FF 76 08 B0 00   s........v..v...
0E24:0060  50 E8 A4 FA 89 46 FA 83-7E FA FF 75 03 E9 BB 00   P....F..~..u....
0E24:0070  8B 5E FA 8A 87 B7 2D 88-46 E7 B4 00 3B 06 AA 2C   .^....-.F...;..,
-g

Program terminated normally
-d 0e24:0000
0E24:0000  25 36 61 00 00 00 00 00-00 00 00 00 00 00 00 00   %6a.............
0E24:0010  B8 24 0E 8E D8 A0 00 00-8A 1E 01 00 8A 0E 03 00   .$..............
0E24:0020  02 C3 27 A2 02 00 73 02-FE C1 88 0E 03 00 B4 4C   ..'...s........L
0E24:0030  CD 21 1E B6 2C B7 00 8A-87 B8 2C 3A 46 0A 75 18   .!..,.....,:F.u.
0E24:0040  D1 E3 8B 87 FC 13 3B 46-08 75 0D 8A 46 06 D0 D8   ......;F.u..F...
0E24:0050  73 03 E9 B8 02 E9 C0 02-FF 76 0A FF 76 08 B0 00   s........v..v...
0E24:0060  50 E8 A4 FA 89 46 FA 83-7E FA FF 75 03 E9 BB 00   P....F..~..u....
0E24:0070  8B 5E FA 8A 87 B7 2D 88-46 E7 B4 00 3B 06 AA 2C   .^....-.F...;..,
-_
```

Figure 1: **Input:** num1: 25, num2: 36; **Output:** ans: 61, carry: 0

BCD Subtraction

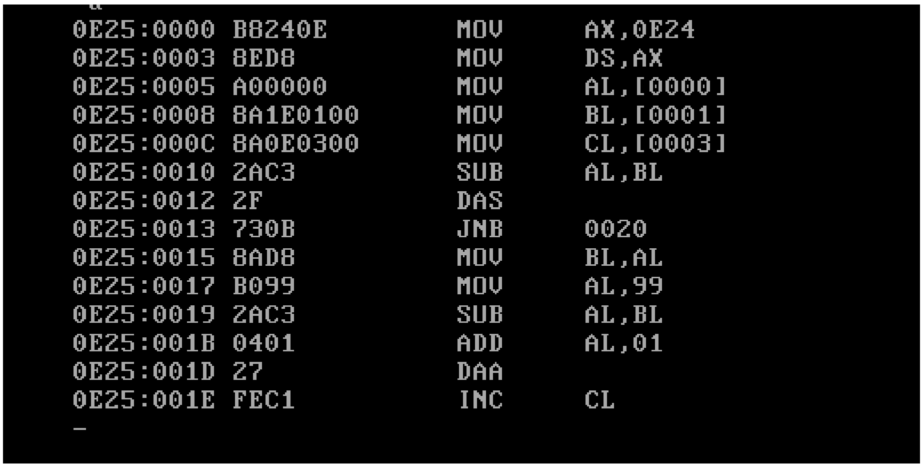
Algorithm:

- Move the data segment to the AX register and then move it to the DS register.
- Move value of num1 to AL, num2 to BL, sign to CL registers.
- Subtract AL and BL using SUB AL, BL.
- Perform Decimal Adjust After Subtraction using DAS instruction.
- Jump to label HERE if no carry.
- Move value in AL to BL register, 99H to AL register.
- Subtract AL and BL using SUB AL, BL.
- Add 1 to AL using ADD AL, 01H.
- Perform Decimal Adjust after Addition using DAA instruction.
- Increment value of CL.
- Move value of CL to sign, under label HERE.
- Move value of AL to ans.

Program:

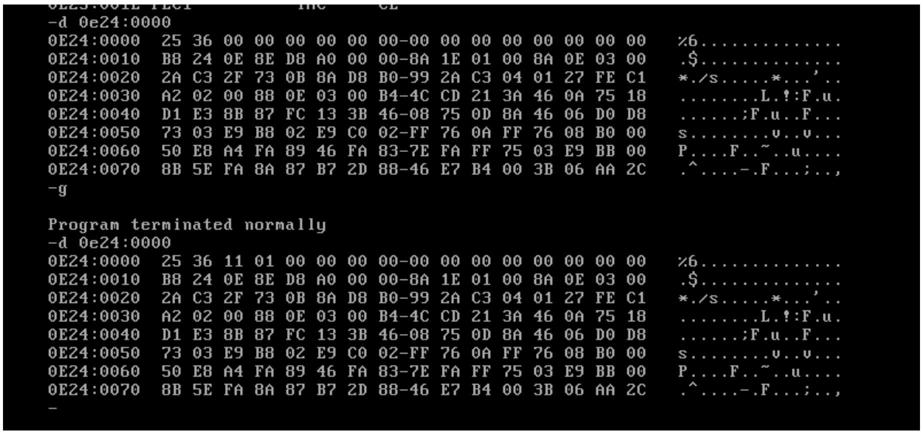
Program	**Comments**
assume cs:code, ds:data	Declare code and data segments
data segment	Start of data segment
num1 db 25H	Define byte num1 with value 25
num2 db 36H	Define byte num2 with value 36
ans db ?	Define byte ans for result
sign db 00H	Define byte sign with value 00
data ends	End of data segment
code segment	Start of code segment
start: mov ax, data	Move data to AX register
mov ds, ax	Move contents of AX register to DS register
mov al, num1	Move value of num1 to AL register
mov bl, num2	Move value of num2 to BL register
mov cl, sign	Move value of sign to CL register
sub al, bl	AL = AL - BL
das	Decimal Adjust after Subtraction
jnc here	Jump to label HERE if CF = 0
mov bl, al	Move value of AL to BL register
mov al, 99H	Move hex value 99H to AL register
sub al, bl	AL = AL - BL
add al, 01H	AL = AL + 1
daa	Decimal Adjust after Addition
inc cl	Increment value of CL
here: mov ans, al	Move value of AL to ans
mov sign, cl	Move value of CL to sign
mov ah, 4ch	To request interrupt
int 21h	Request interrupt routine
code ends	End of code segment
end start	

Unassembled code:



```
-u
0E25:0000 B8240E        MOV     AX,0E24
0E25:0003 8ED8          MOV     DS,AX
0E25:0005 A00000        MOV     AL,[0000]
0E25:0008 8A1E0100      MOV     BL,[0001]
0E25:000C 8A0E0300      MOV     CL,[0003]
0E25:0010 2AC3          SUB     AL,BL
0E25:0012 2F            DAS
0E25:0013 730B          JNB     0020
0E25:0015 8AD8          MOV     BL,AL
0E25:0017 B099          MOV     AL,99
0E25:0019 2AC3          SUB     AL,BL
0E25:001B 0401          ADD     AL,01
0E25:001D 27            DAA
0E25:001E FEC1          INC     CL
-
```

Input and Output:



```
0E25:001E FEC1          INC     CL
-d 0e24:0000
0E24:0000  25 36 00 00 00 00 00 00-00 00 00 00 00 00 00 00   %6..............
0E24:0010  B8 24 0E 8E D8 A0 00 00-8A 1E 01 00 8A 0E 03 00   .$..............
0E24:0020  2A C3 2F 73 0B 8A D8 B0-99 2A C3 04 01 27 FE C1   *./s.....*...'..
0E24:0030  A2 02 00 88 0E 03 00 B4-4C CD 21 3A 46 0A 75 18   ........L.!:F.u.
0E24:0040  D1 E3 8B 87 FC 13 3B 46-08 75 0D 8A 46 06 D0 D8   ......;F.u..F...
0E24:0050  73 03 E9 B8 02 E9 C0 02-FF 76 0A FF 76 08 B0 00   s........v..v...
0E24:0060  50 E8 A4 FA 89 46 FA 83-7E FA FF 75 03 E9 BB 00   P....F..~..u....
0E24:0070  8B 5E FA 8A 87 B7 2D 88-46 E7 B4 00 3B 06 AA 2C   .^....-.F...;..,
-g

Program terminated normally
-d 0e24:0000
0E24:0000  25 36 11 01 00 00 00 00-00 00 00 00 00 00 00 00   %6..............
0E24:0010  B8 24 0E 8E D8 A0 00 00-8A 1E 01 00 8A 0E 03 00   .$..............
0E24:0020  2A C3 2F 73 0B 8A D8 B0-99 2A C3 04 01 27 FE C1   *./s.....*...'..
0E24:0030  A2 02 00 88 0E 03 00 B4-4C CD 21 3A 46 0A 75 18   ........L.!:F.u.
0E24:0040  D1 E3 8B 87 FC 13 3B 46-08 75 0D 8A 46 06 D0 D8   ......;F.u..F...
0E24:0050  73 03 E9 B8 02 E9 C0 02-FF 76 0A FF 76 08 B0 00   s........v..v...
0E24:0060  50 E8 A4 FA 89 46 FA 83-7E FA FF 75 03 E9 BB 00   P....F..~..u....
0E24:0070  8B 5E FA 8A 87 B7 2D 88-46 E7 B4 00 3B 06 AA 2C   .^....-.F...;..,
-
```

Figure 2: **Input:** num1: 25, num2: 36; **Output:** ans: 11, sign: 1

Result:

The 8086 programs were written to perform BCD addition and subtraction operations, and the results observed.