Lab-4

Arithmetic Operations – Part 2 – CS2523

Table of Contents

1. Introduction

The four basic arithmetic operations are addition, subtraction, multiplication and division. Apart from this we will cover increment and decrement operator as well in this lab.

1.1. Addition

Addition is the most basic operation of arithmetic. In its simplest form, addition combines two quantities into a single quantity, or *sum*. For example, say you have a group of 2 boxes and another group of 3 boxes. If you combine both groups together, you now have one group of 5 boxes. To represent this idea in mathematical terms:

2 Boxes + 3 Boxes = Will result in 5 Boxes

1.2. Subtraction

Subtraction is the opposite of addition. Instead of adding quantities together, we are removing one quantity from another to find the *difference* between the two. Continuing the previous example, say you start with a group of 5 boxes. If you then remove 3 boxes from that group, you are left with 2 boxes. In mathematical terms:

5 Boxes – 3 Boxes =Will result in 2 Boxes

1.3. Multiplication

Multiplication also combines multiple quantities into a single quantity, called the *product*. In fact, multiplication can be thought of as a consolidation of many additions. Specifically, the product of xx and yy is the result of xx added together yy times. For example, one way of counting four groups of two boxes is to add the groups together:

2+2+2+2=8

However, another way to count the boxes is to multiply the quantities:

2 * 4 = 8

Note that both methods give you the same result—8—but in many cases, particularly when you have large quantities or many groups, multiplying can be much faster.

1.4. Division

Division is the inverse of multiplication. Rather than multiplying quantities together to result in a larger value, you are splitting a quantity into a smaller value, called the *quotient*. Again, to return to the box example, splitting up a group of 8 boxes into 4 equal groups results in 4 groups of 2 boxes:

8 ÷ 4 = 2

2. The Basic Arithmetic Properties

2.1. Commutative Property

The commutative property describes equations in which the *order* of the numbers involved does not affect the result. Addition and multiplication are commutative operations:

- 2+3=3+2=52+3=3+2=5
- 5·2=2·5=105·2=2·5=10

Subtraction and division, however, are not commutative.

2.2. Associative Property

The associative property describes equations in which the *grouping* of the numbers involved does not affect the result. As with the commutative property, addition and multiplication are associative operations:

- (2+3)+6=2+(3+6) = 11
- (4·1)·2=4·(1·2) = 8

Once again, subtraction and division are not associative.

2.3. Distributive Property

The distributive property can be used when the sum of two quantities is then multiplied by a third quantity.

- (2+4)·3=2·3+4·3=18

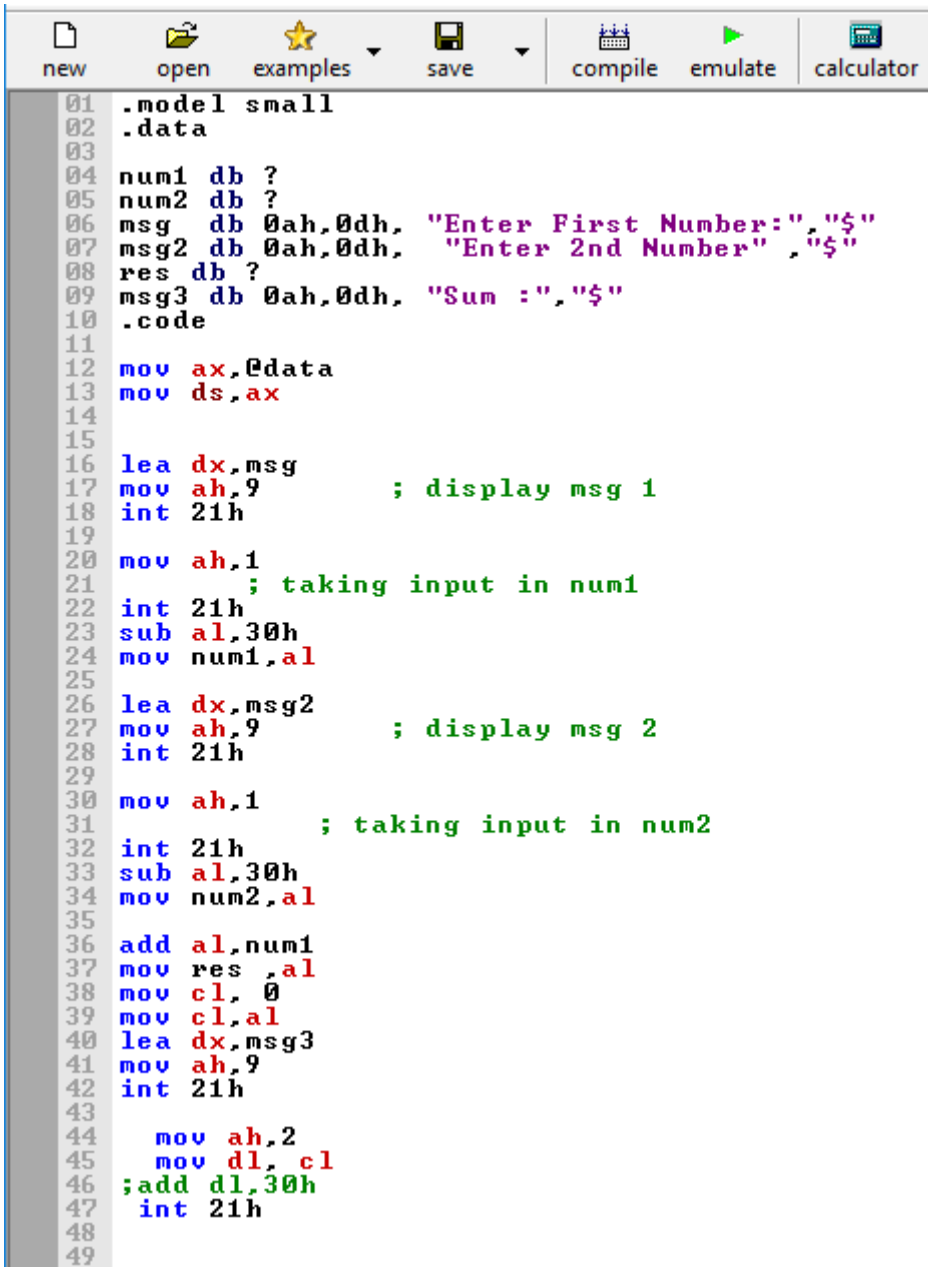
3. Objective

- To get basic understanding of arithmetic operation in Assembly Language.
- Arithmetic properties
- To get an understanding of number representation in ascii format.

.

4. Walkthrough Task

The above program shows the input of two numbers and add them and store the result in res variable.




```
.model small
.data

num1 db ?
num2 db ?
msg  db 0ah,0dh, "Enter First Number:","$"
msg2 db 0ah,0dh,  "Enter 2nd Number" ,"$"
res db ?
msg3 db 0ah,0dh, "Sum :","$"
.code

mov ax,@data
mov ds,ax


lea dx,msg
mov ah,9        ; display msg 1
int 21h

mov ah,1
        ; taking input in num1
int 21h
sub al,30h
mov num1,al

lea dx,msg2
mov ah,9        ; display msg 2
int 21h

mov ah,1
            ; taking input in num2
int 21h
sub al,30h
mov num2,al

add al,num1
mov res ,al
mov cl, 0
mov cl,al
lea dx,msg3
mov ah,9
int 21h

  mov ah,2
  mov dl, cl
;add dl,30h
 int 21h
```

5. Procedure& Tools

Procedure and tools installation discussed in first lab (Lab01).

5.1. Tools

- Open emu8086.
- Make new project.
- Enter your code.
- Click on Emulate.
- Click on Run.

6. Practice Tasks

This section will provide more practice exercises which you need to finish during the lab. You need to finish the tasks in the required time.

6.1. Practice Task 1 [Expected time = 15mins]

Write an assembly program to solve the following equation
Res=A+B+(C+D)+E+(F+G)
Whereas the value of parameters is as follow

- A=1
- B=2
- C=3
- D=4
- E=5
- G=6

At the end the result of the equation should be stored in **Res** variable.

6.2. Practice Task 2 [Expected time = 15mins]

Write an assembly program to solve the following equation
Res=A-B-(C-D)-E-(F-G)
Whereas the value of parameters is as follow

- A=6
- B=5
- C=4
- D=3
- E=2
- G=1

At the end the result of the equation should be stored in **Res** variable.

6.3. Practice Task 3 [Expected time = 15mins]

Write an assembly program to solve the following equation
Res=A+B-(C+D)-E+(F-G)
Whereas the value of parameters is as follow

- A=3
- B=2
- C=2
- D=1
- E=1
- G=3

At the end the result of the equation should be stored in **Res** variable.

6.4. Practice Task 4 [Expected time = 15mins]

Write an assembly program to solve the following equation

Res=A-B-(C+D)+E-(F-G)

Whereas the value of parameters is as follow

- A=4
- B=1
- C=3
- D=2
- E=4
- G=3

At the end the result of the equation should be stored in Res variable.

7. Out comes

After completing this lab, student will be able to compile and run basic Assembly arithmetic operation with their associative properties.

8. Evaluation Task (Unseen) [Expected time = 30mins for tasks]

The lab instructor will give you unseen task depending upon the progress of the class.

9. Evaluation criteria

The evaluation criteria for this lab will be based on the completion of the following tasks. Each task is assigned the marks percentage which will be evaluated by the instructor in the lab whether the student has finished the complete/partial task(s).

Table: Evaluation of the Lab

Sr. No.	Task No	Description	Marks
1		Understanding of Problem	20
2		Program Logic	20
3		Program Implementation	20
4		Program Correctness	10
5		Use of Tool	10
6		Viva	20