Name: **Tushar Nankani** Roll No: **1902112** Batch: **C23**

**Microprocessor: Experiment 9**

**Aim**: Mixed Language program to find the GCD/LCM of two numbers.

**Theory:**

Mixed-language programming is the process of building programs in which the source code is written in two or more languages.

It allows you to:

• Call existing code that is written in another language

• Use procedures that may be difficult to implement in a particular language

• Gain advantages in processing speeds

Mixed-language programming is possible between Intel Fortran and other languages.

Although other languages (such as assembly language) are discussed,

the primary focus of this section is programming using Intel Fortran and C/C++ .

Mixed language programming between these two languages is relatively

straightforward because each language implements functions, subroutines, and

procedures in approximately the same way.

**What is GCD?**

In mathematics, the greatest common divisor (GCD) of two or more integers,

which are not all zero, is the largest positive integer that divides each of the integers.

For eg.

36 = 2 X 2 X 3 X 3

60 = 2 X 2 X 3 X 5

Multiplication of common factors: 2 X 2 X 3 = 12

**What is LCM ?**

The least common multiple (LCM) of two numbers is the smallest number (not zero)

that is a multiple of both

For eg:

15 = 5 X 3

25 = 5 X 5

Union of all the factors = 5 X 5 X 3

= 75

**Algorithm:**

**GCD**

* Load value d1 in ax and d2 in bx
* Call the gcd function
* If value in bx is zero
* Then set the value of gcd ( CX ) as ax
* Else set the value of ax as bx and value of bx as ax % bx
* Call the gcd function recursively
* Load the value of cx into ax
* Call the print function to print the gcd of two numbers

**LCM**

* Start
* Store first number(num1) in a register
* Store second number(num2) in another register
* Initialize a counter register(Rd) to 01h
* Compare both the values num1 and num2
* If num1 = num2 : Store num1 or num2 as result and jump to step 8
* If num1 < num2 : Swap the register values so that num1 > num2
* Multiply num2 and Rd and divide the product with num1
* Check the reminder
* If reminder is zero then store product obtained from multiplication in step 6 as result and jump to step 8
* Else increment Rd and repeat steps 6 and 7
* Stop

**CODE:**

**#include<stdio.h>**

**#include<conio.h>**

**void main()**

**{**

**int gcd,a,b;**

**clrscr();**

**printf("enter two numbers\n");**

**scanf("%d%d",&a,&b);**

**asm mov ax,a;**

**asm mov bx,b;**

**l1:**

**asm cmp ax,bx;**

**asm jz l;**

**asm jc m;**

**asm jmp n;**

**m:**

**asm sub bx,ax;**

**asm loop l1;**

**n:**

**asm sub ax,bx;**

**asm loop l1;**

**l:**

**asm mov gcd,ax;**

**printf("GCD of the two numbers is %d\n",gcd);**

**int a1,b1,temp;**

**printf("Please enter first number: ");**

**scanf("%d",&a1);**

**printf("Please enter second number: ");**

**scanf("%d",&b1);**

**temp = a1;**

**asm mov ax,a1**

**asm mov bx,b1**

**bck:**

**asm cmp ax,0000h**

**asm jz ex**

**asm cmp bx,0000h**

**asm jz ex**

**asm div bl**

**asm cmp ah,00h**

**asm jz ex**

**temp = temp + a1;**

**asm mov ax,temp**

**asm mov bx,b1**

**asm jmp bck**

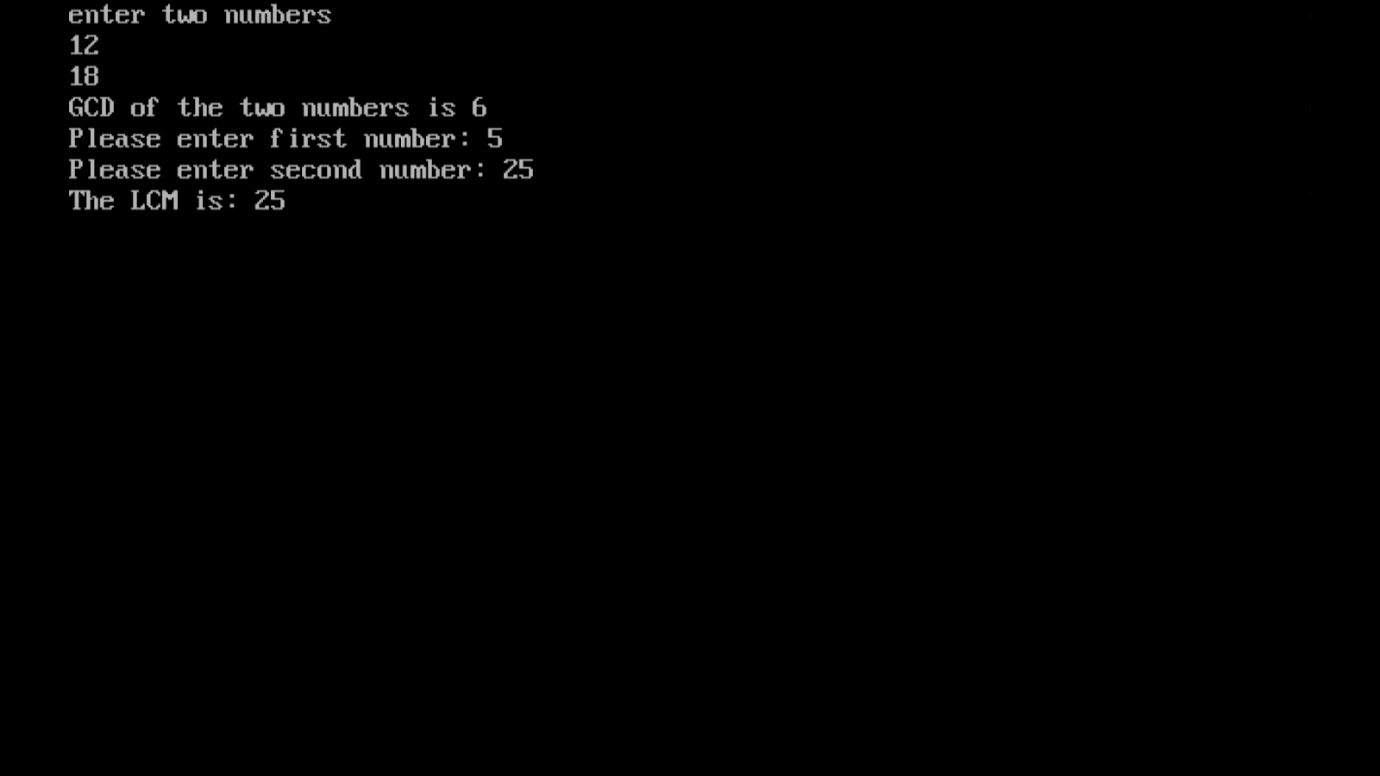
**ex:**

**printf("The LCM is: %d\n",temp);**

**getch();**

**}**

**OUTPUT:**

****