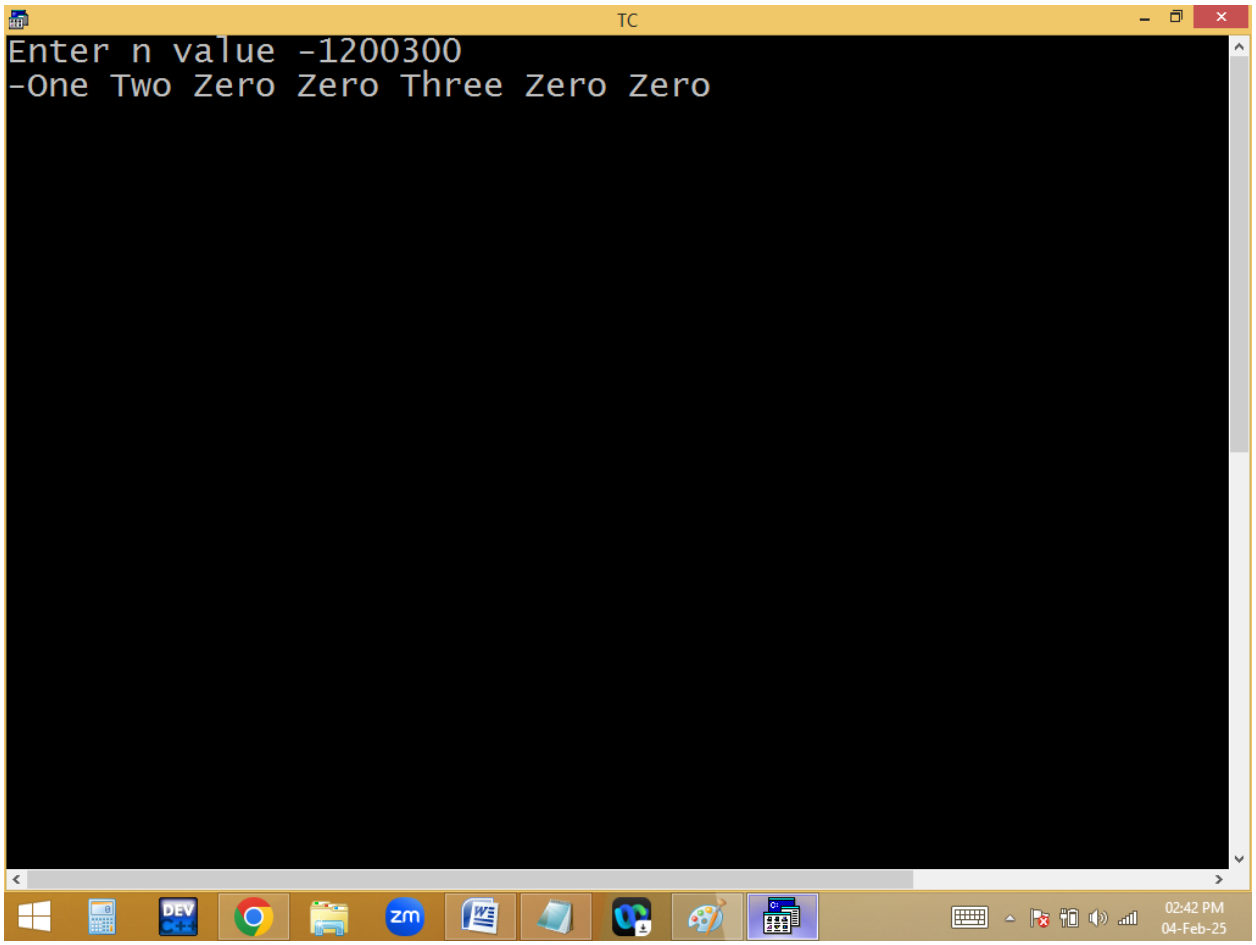
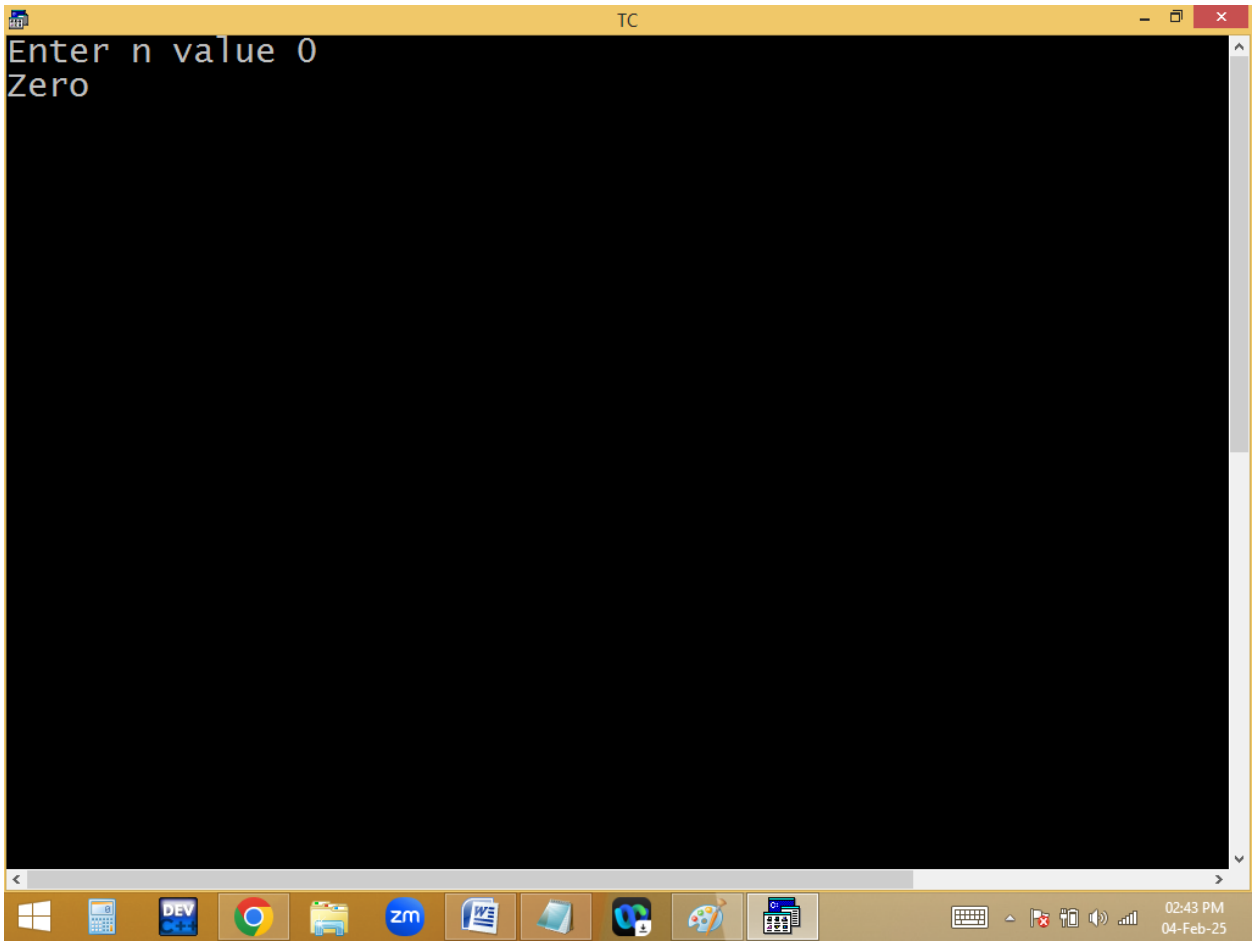
102 ➔ One Zero Two

```
#include<stdio.h>
#include<conio.h>
void main()
{
long m,n,rev=0; int r;
clrscr();
printf("Enter n value "); scanf("%ld",&n);
if(n<0)printf("-",n=-n); m=n;
do{r=n%10;rev=rev*10+r;n=n/10;}while(n!=0);/*rev no*/
do
{
switch(rev%10)
{
case 0: printf("Zero");break;
case 1: printf("One");break;
case 2: printf("Two");break;
case 3: printf("Three");break;
```

```
case 4: printf("Four");break;
case 5: printf("Five");break;
case 6: printf("Six");break;
case 7: printf("Seven");break;
case 8: printf("Eight");break;
case 9: printf("Nine");break;
}
printf(" "); rev=rev/10;
}while(rev!=0);
while(m%10==0&&m!=0)printf("Zero ",m/=10);
getch();
}
```


TC
Enter n value -1200300
-One Two Zero Zero Three Zero Zero
02:42 PM
04-Feb-25



n → rev

102 → 201 % 10 = 1 One

20 % 10 = 0 Zero

2 % 10 = 2 Two

```
do
{
switch( rev%10 )
{
case 0: p(zero);break;
case 1: p(one);b;
case 2: p(two);b;
case 9: p(nine);
}
rev=rev/10;
p(" ");
}while(rev!=0);
```

m

100 % 10 = 0

n → rev

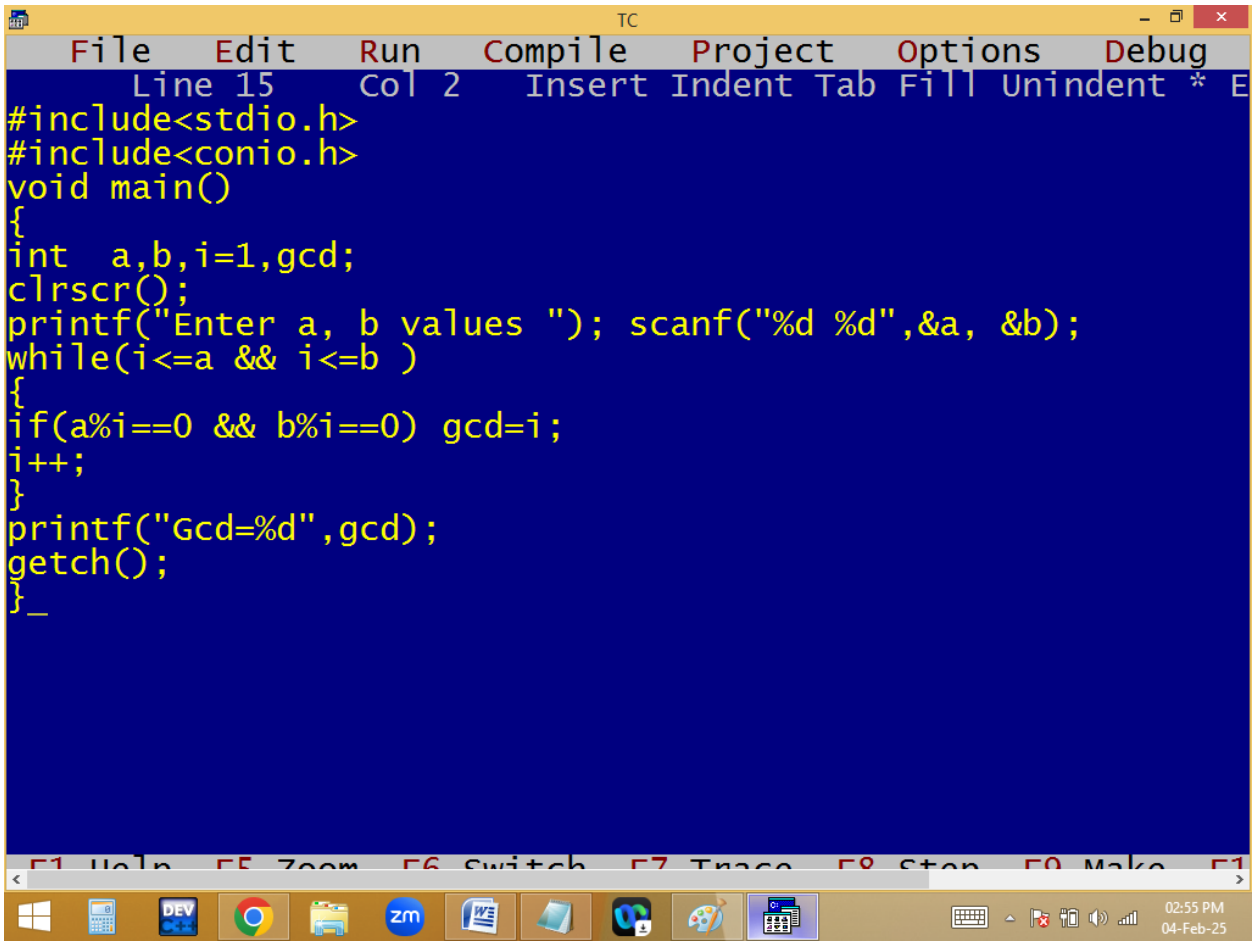
100 → 1 — One Zero Zero

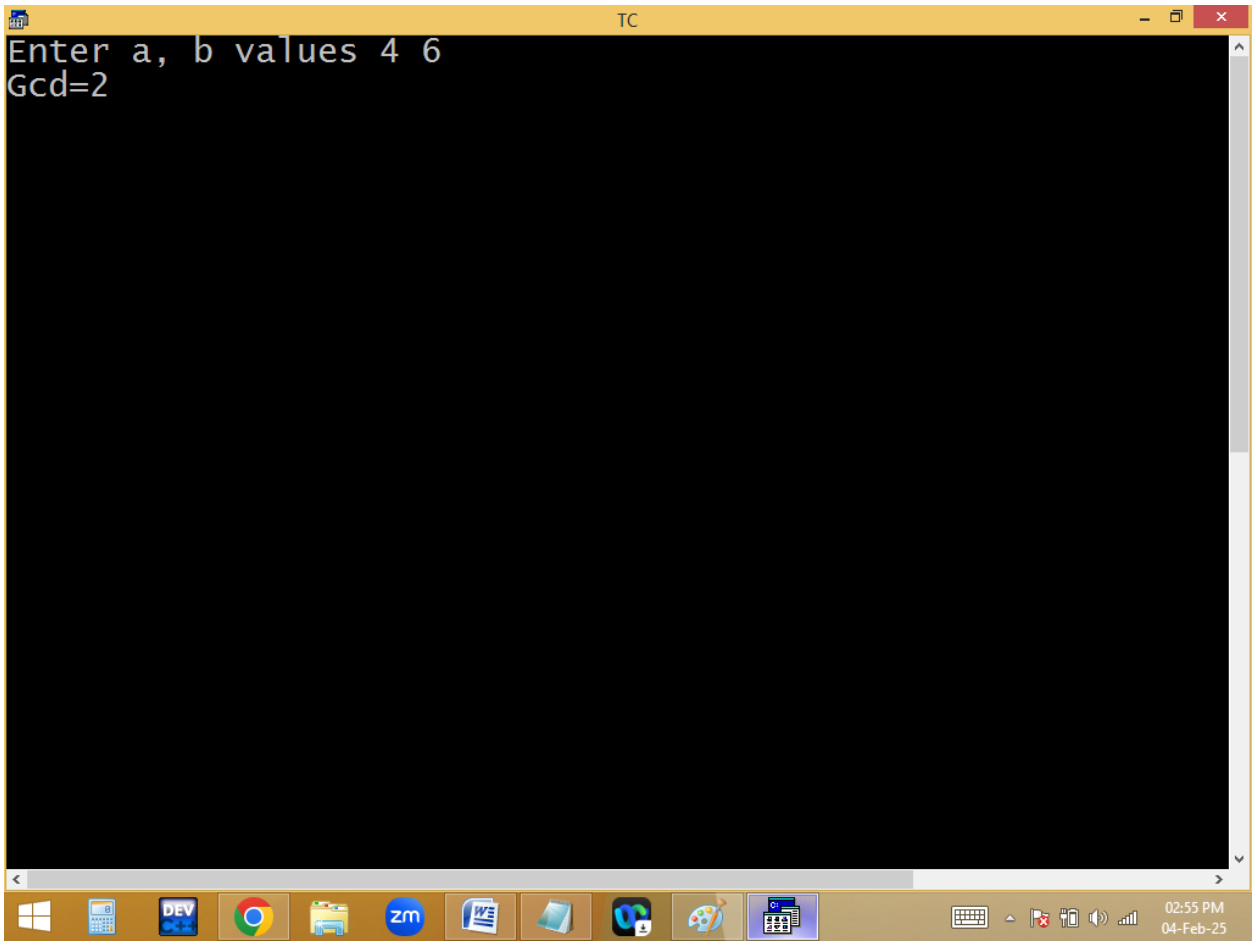
while(m%10==0 && m!=0) p("Zero", m=m/10);

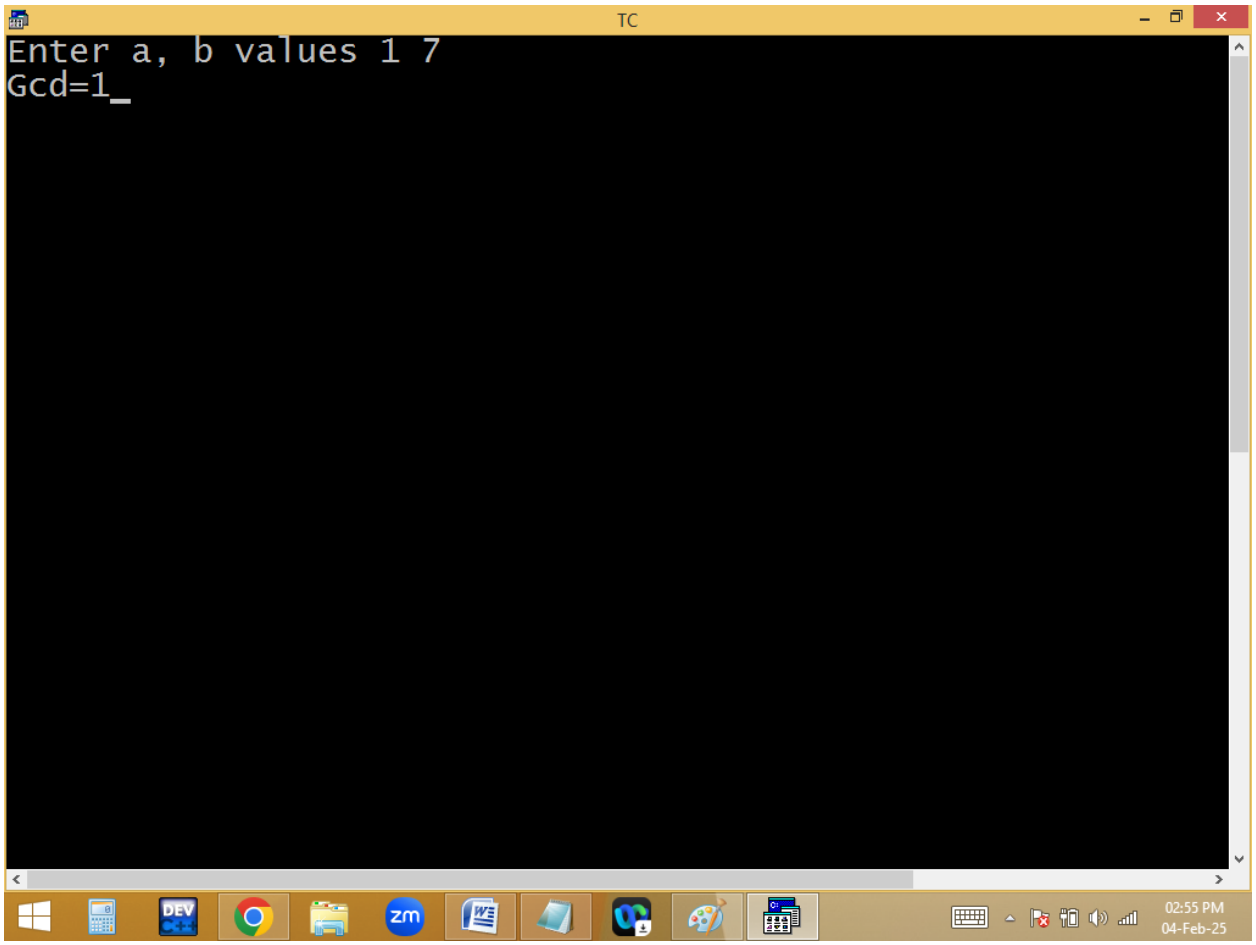
Finding gcd / hcf of given two no's:

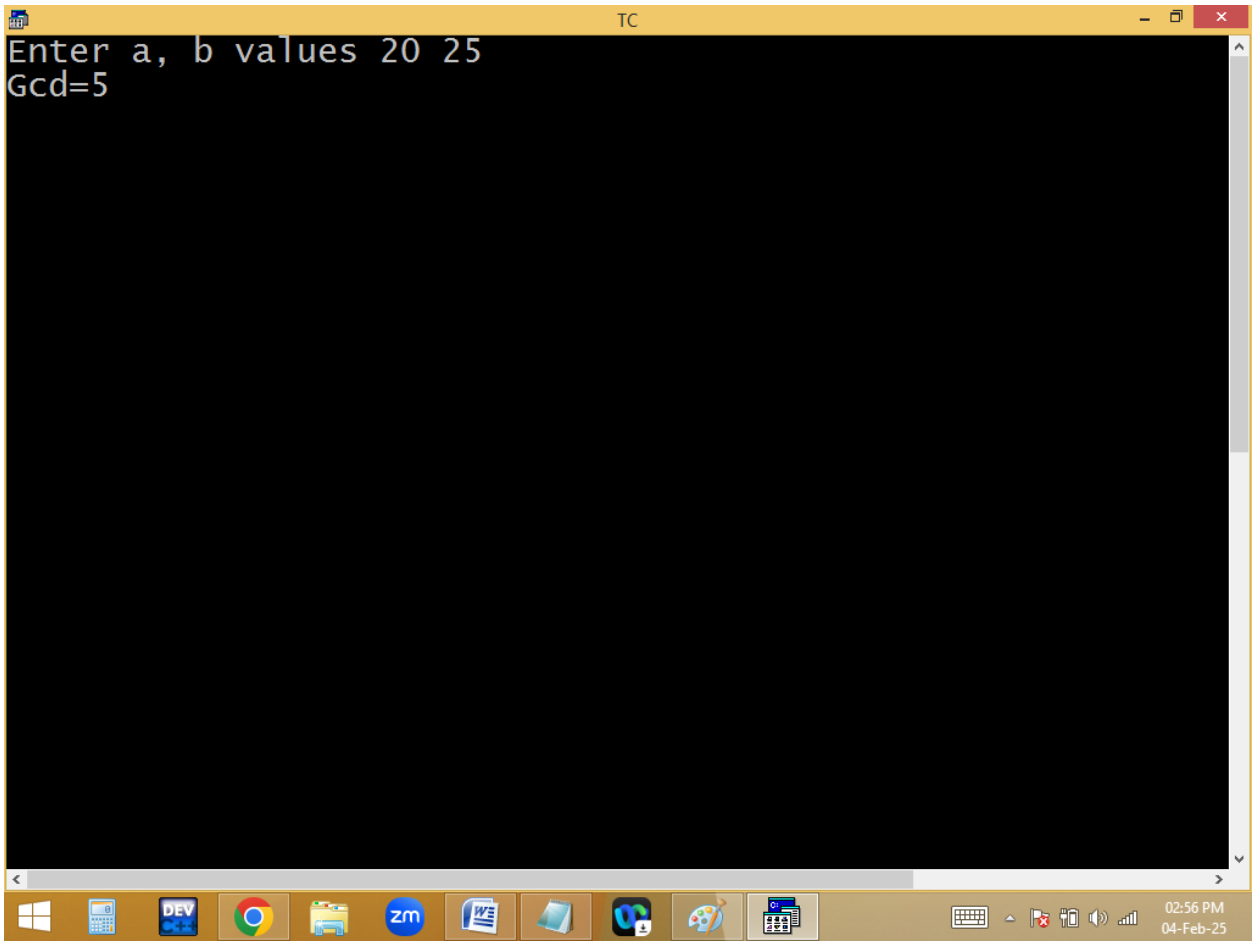
4 factors 1 2 4

6 factors 1 2 3 6


TC
File Edit Run Compile Project Options Debug
Line 15 Col 2 Insert Indent Tab Fill Unindent * E
#include<stdio.h>
#include<conio.h>
void main()
{
int a,b,i=1,gcd;
clrscr();
printf("Enter a, b values "); scanf("%d %d",&a, &b);
while(i<=a && i<=b)
{
if(a%i==0 && b%i==0) gcd=i;
i++;
}
printf("Gcd=%d",gcd);
getch();
}_
02:55 PM
04-Feb-25


TC
Enter a, b values 4 6
Gcd=2
02:55 PM
04-Feb-25


TC
Enter a, b values 1 7
Gcd=1
02:55 PM
04-Feb-25



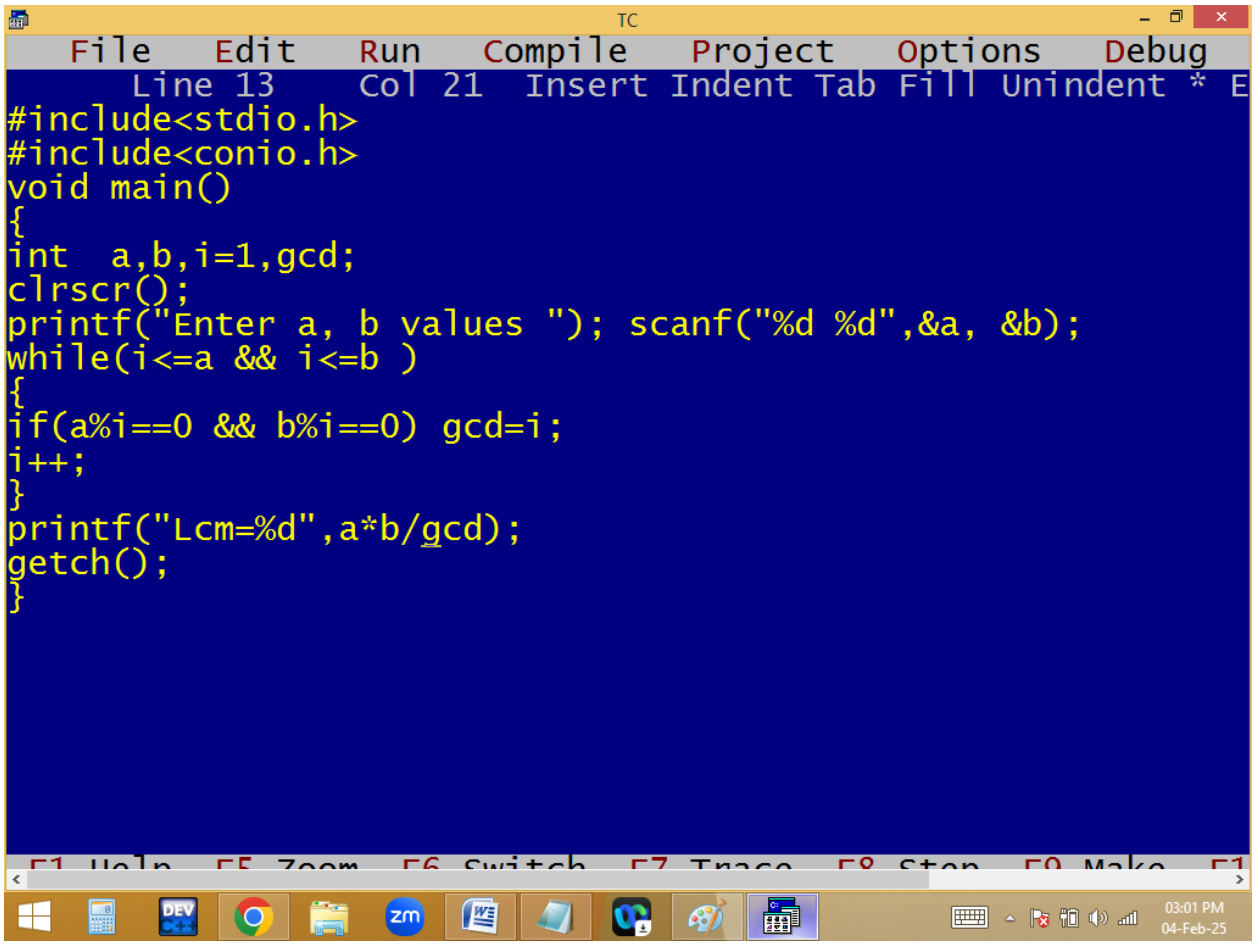

```
while( i<=a && i<=b )
{
if(a%i==0 && b%i==0) gcd=i;
i++;
}
p(gcd);
```

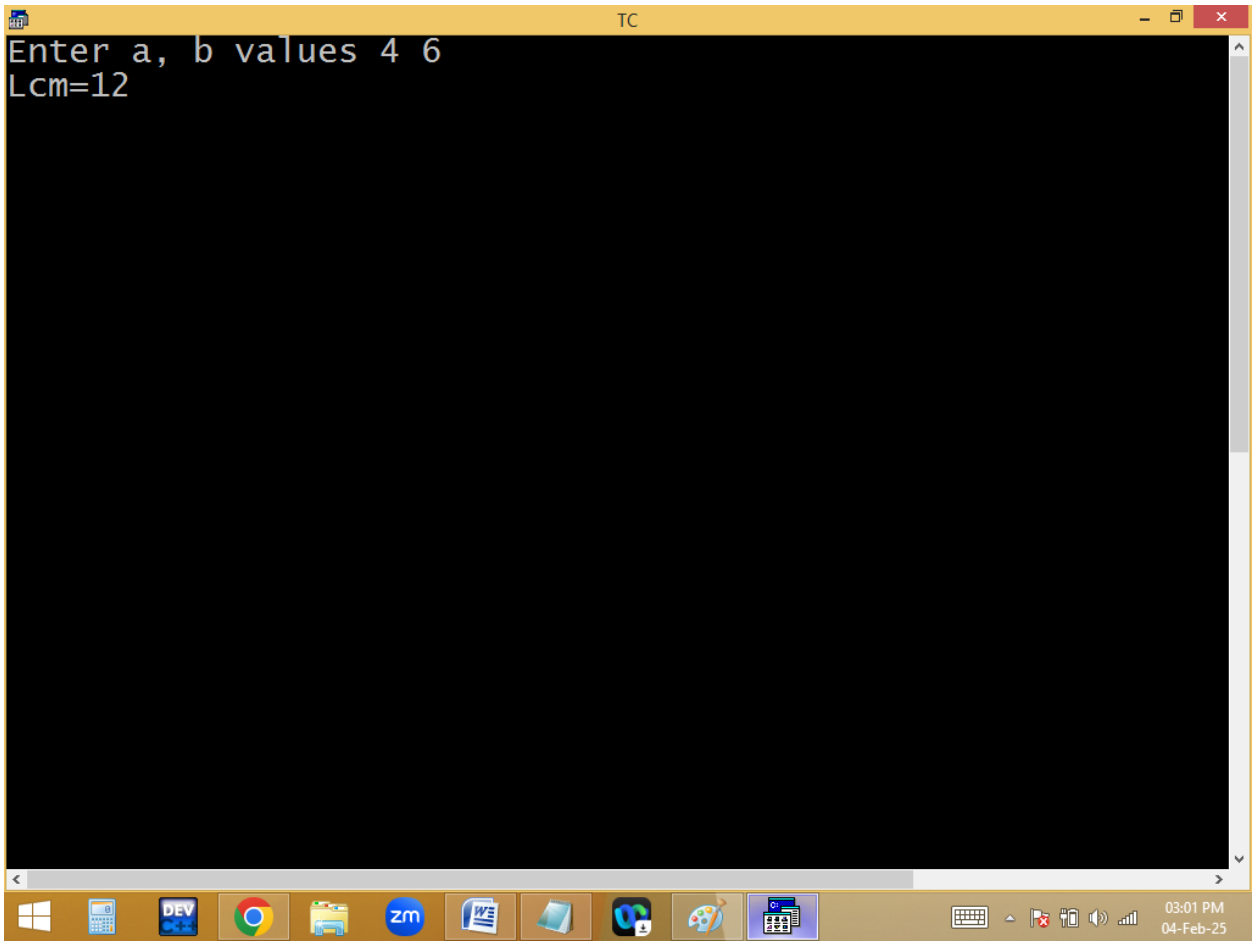
a	i	b	i	gcd
4	1	6	1	1
4	2	6	2	2
4	3			
4	4	6	4	
5				

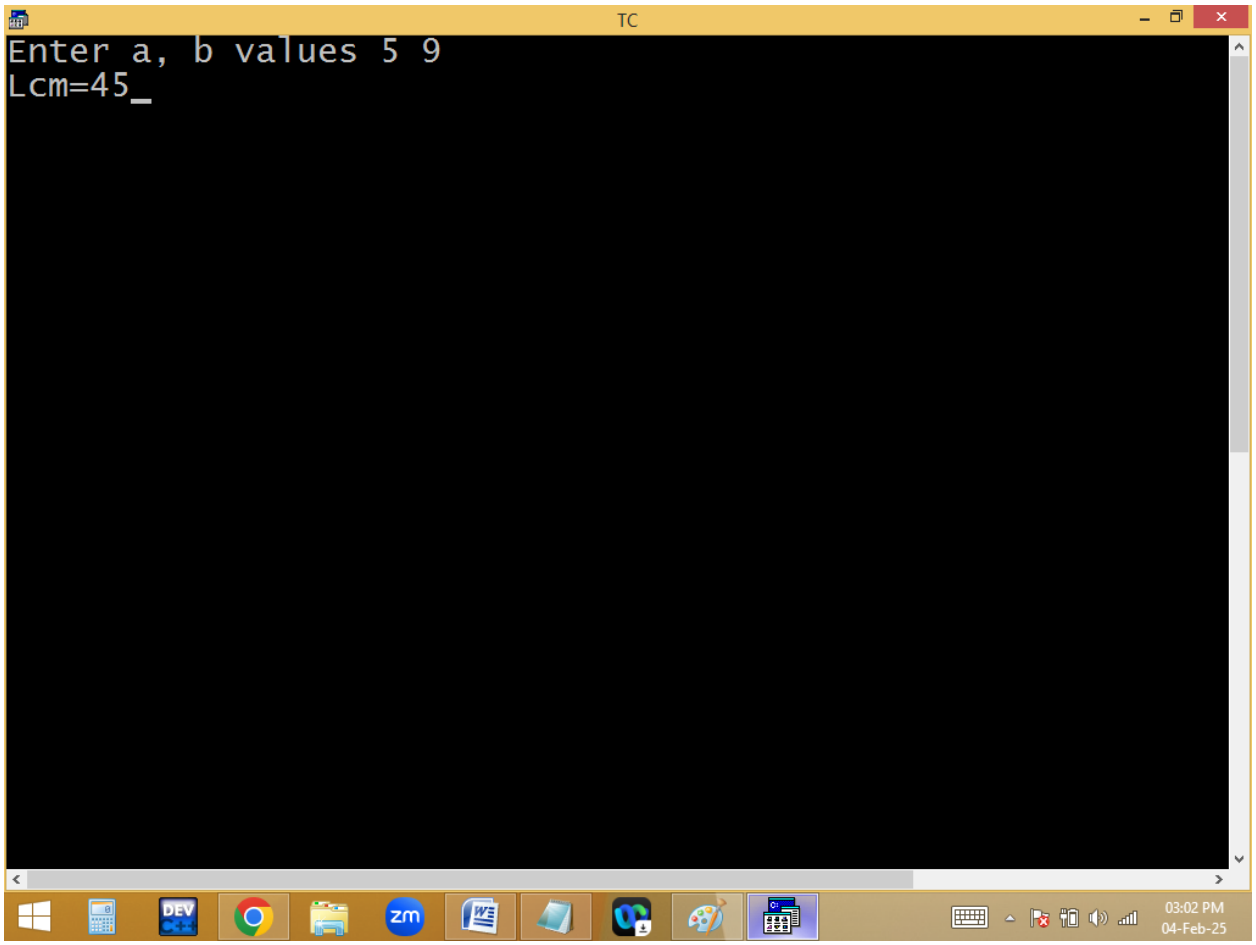
5<=4
1 <=4 <=6

Finding lcm of given two no's:

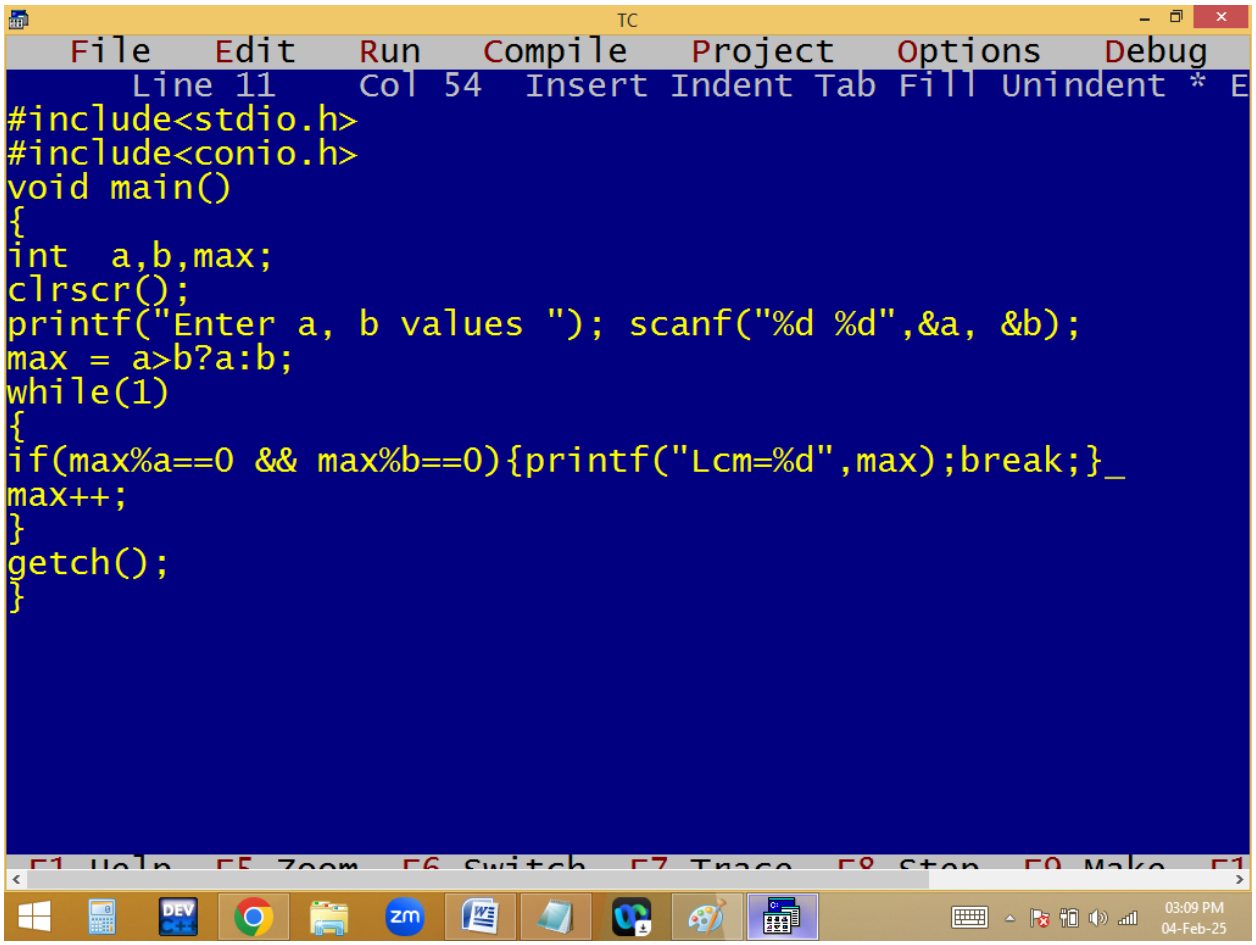
Using gcd:

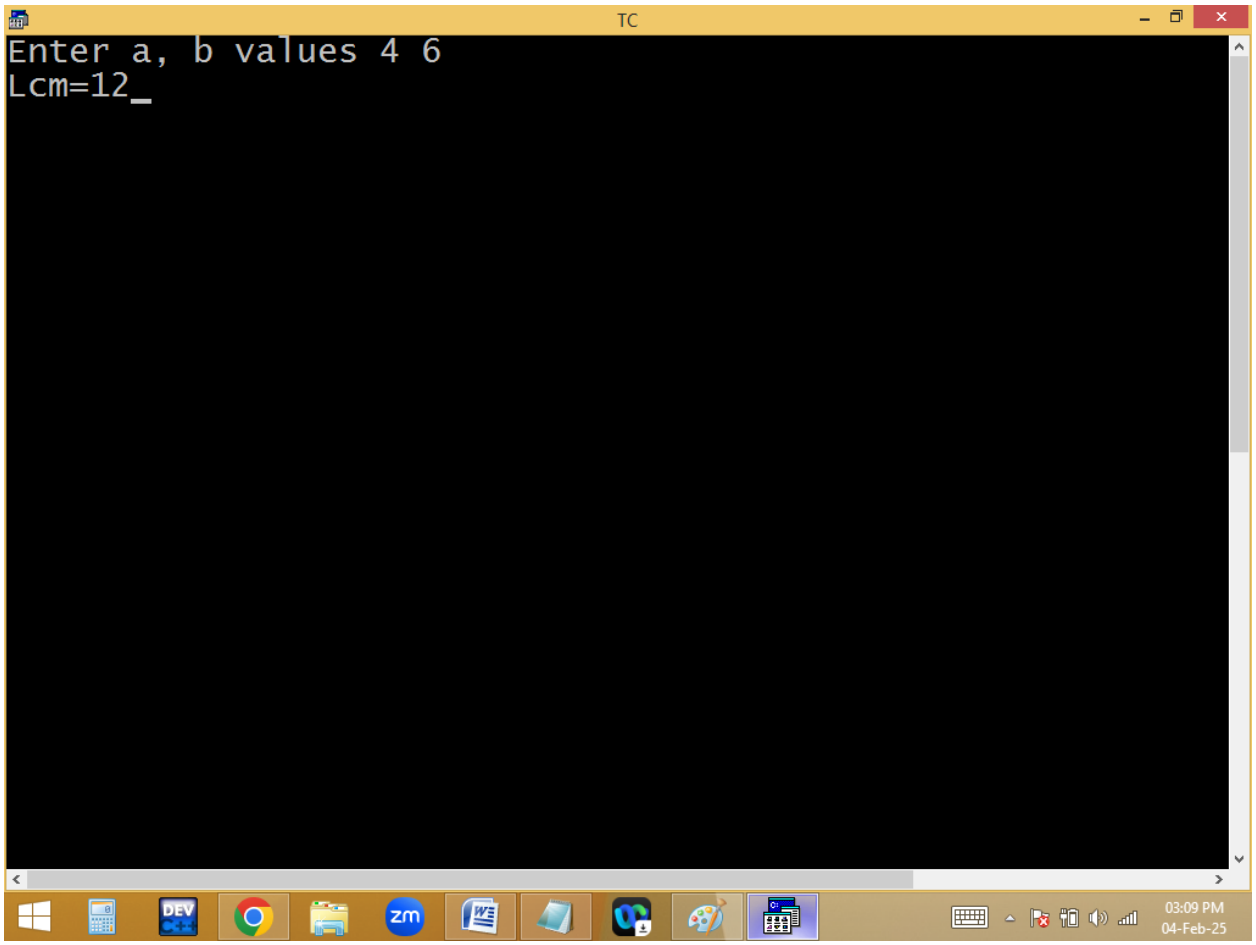


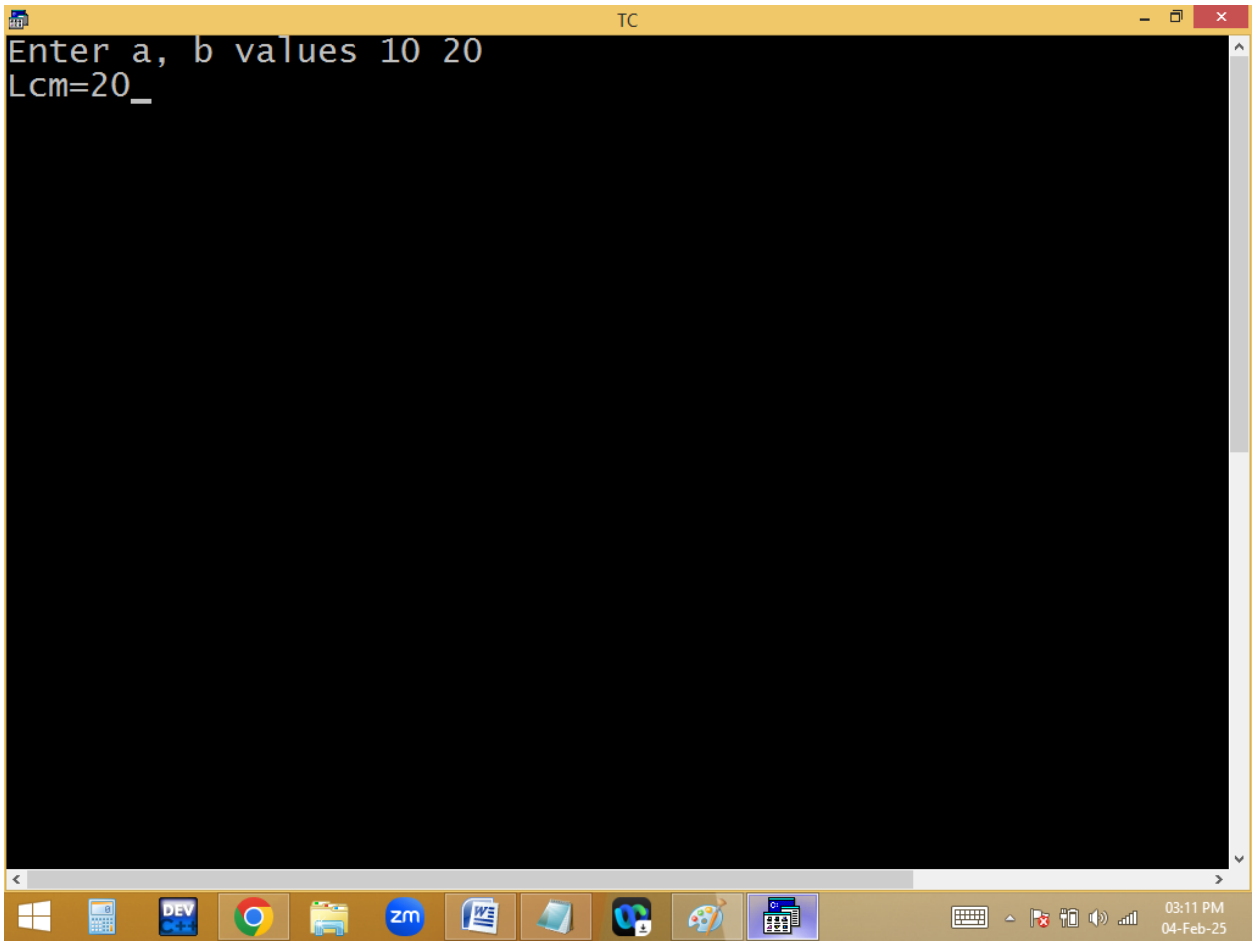

TC
Enter a, b values 4 6
Lcm=12
03:01 PM
04-Feb-25



Method2:


TC
File Edit Run Compile Project Options Debug
Line 11 Col 54 Insert Indent Tab Fill Unindent * E
#include<stdio.h>
#include<conio.h>
void main()
{
int a,b,max;
clrscr();
printf("Enter a, b values "); scanf("%d %d",&a, &b);
max = a>b?a:b;
while(1)
{
if(max%a==0 && max%b==0){printf("Lcm=%d",max);break;}_
max++;
}
getch();
}
03:09 PM
04-Feb-25


TC
Enter a, b values 4 6
Lcm=12
03:09 PM
04-Feb-25




```
max=a>b?a:b;

while(1)
{
if(max%a==0 && max%b==0)
{
p("lcm=%d",max);break;
}
max++;
}
```

max	a	max	b
6 %	4 : 2	6	6
7 %	4 = 3		
8 %	4 : 0	8 % 6 = 2	
9 %	4 = 1		
10 % 4 = 2			
11 % 4 = 3			
12 % 4 = 0		12 % 6 = 0	

for loop:

for loop:

It is an entry control loop.

for is a keyword.

It is also used to repeat a program several times based on a condition.

When compared with while and do while, for loop is looking to be smart. In for it is compulsory to maintain two semicolons. For works without condition also and default condition is always 1 i.e. true.

Generally for loop is having 3 expressions.

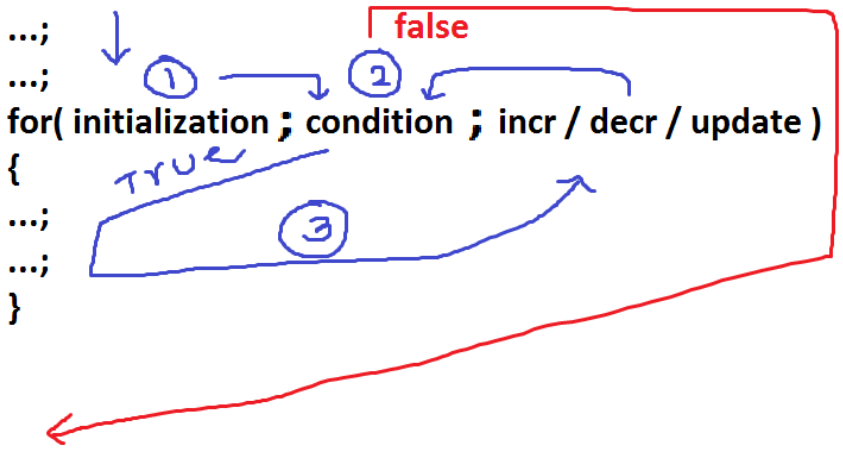
1. **Initialization**
2. **Test condition / expression**
3. **Increment/decrement / updation**

At first entry of for loop the initialization part is executed and later the test condition is checked. If the condition is true then the for block statements are executed. After completion of the block, the increment or decrement part is executed. Later once again the test condition is evaluated. If it is true then once again for block statements are executed. Like this the process is continued until the condition becomes false. Here the initialization part is executed only once, at the time of loop beginning.

It is mandatory to maintain 2 semicolon (;) in a for loop.

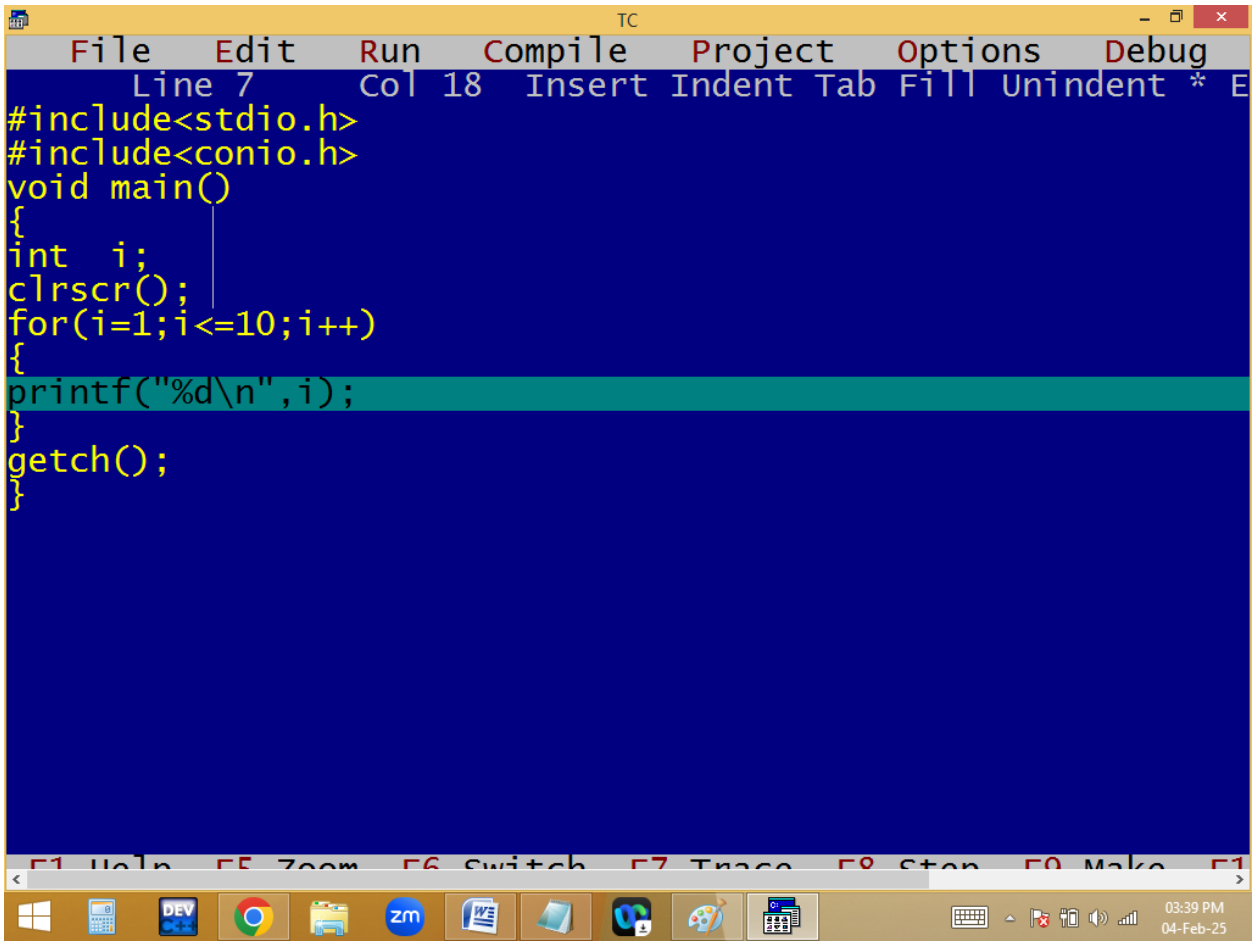
If the for loop is having more than three expressions, it is mandatory to separate the expressions with , separator.

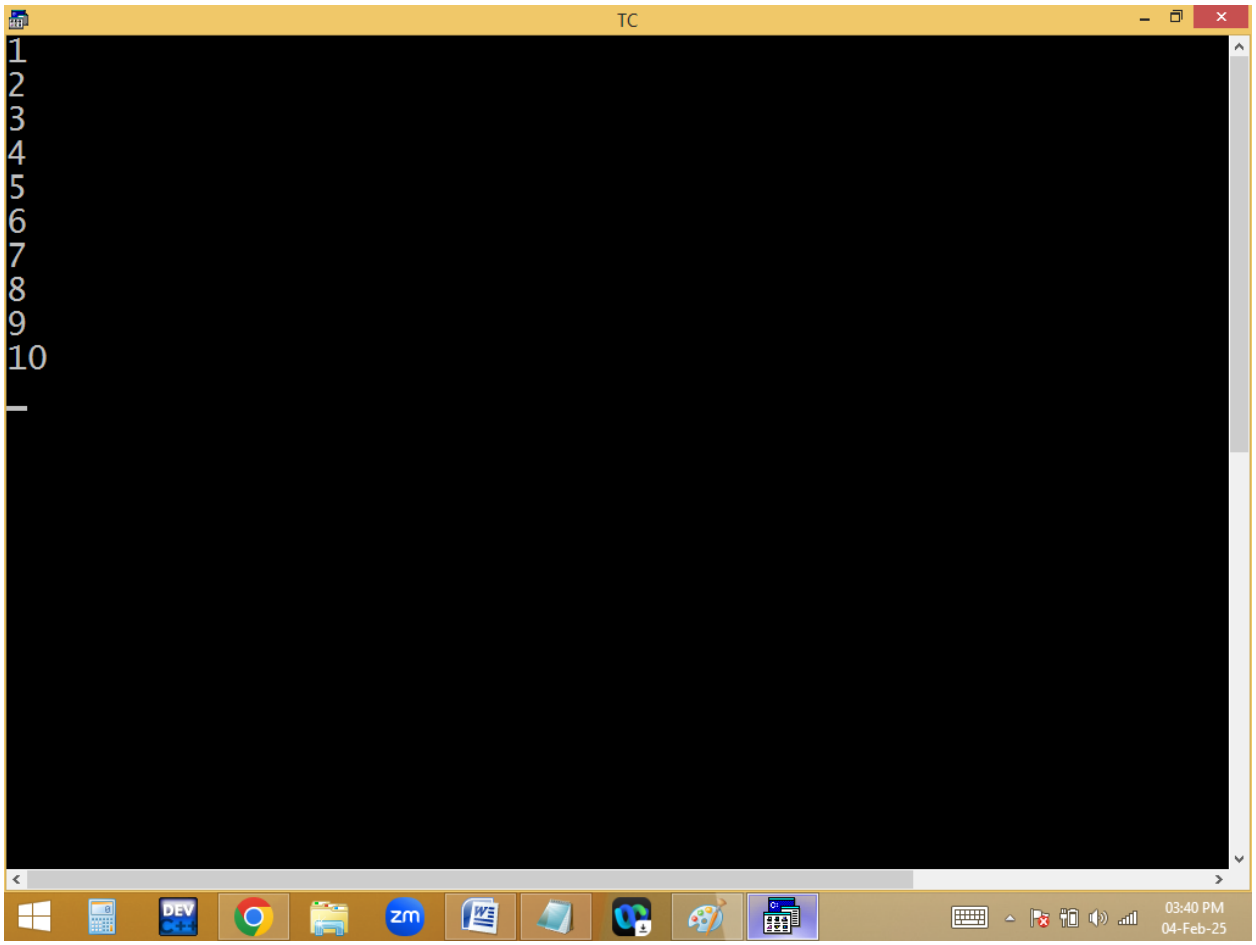
If the for loop is having less than three expressions, then leave the expressions with empty semicolon.




```
for(exp ; exp ; exp )
{
}

for(exp, exp ; exp ; exp,exp)
{
}

for( ;  exp  ; )
{
}

for( ;  ; )
{
}
```

Printing 1..10 no’s with for:


TC
File Edit Run Compile Project Options Debug
Line 7 Col 18 Insert Indent Tab Fill Unindent * E
#include<stdio.h>
#include<conio.h>
void main()
{
int i;
clrscr();
for(i=1;i<=10;i++)
{
printf("%d\n",i);
}
getch();
}
03:39 PM
04-Feb-25



Printing given table:

Example 9th table

9*1=9

9*2=18

9*3=27

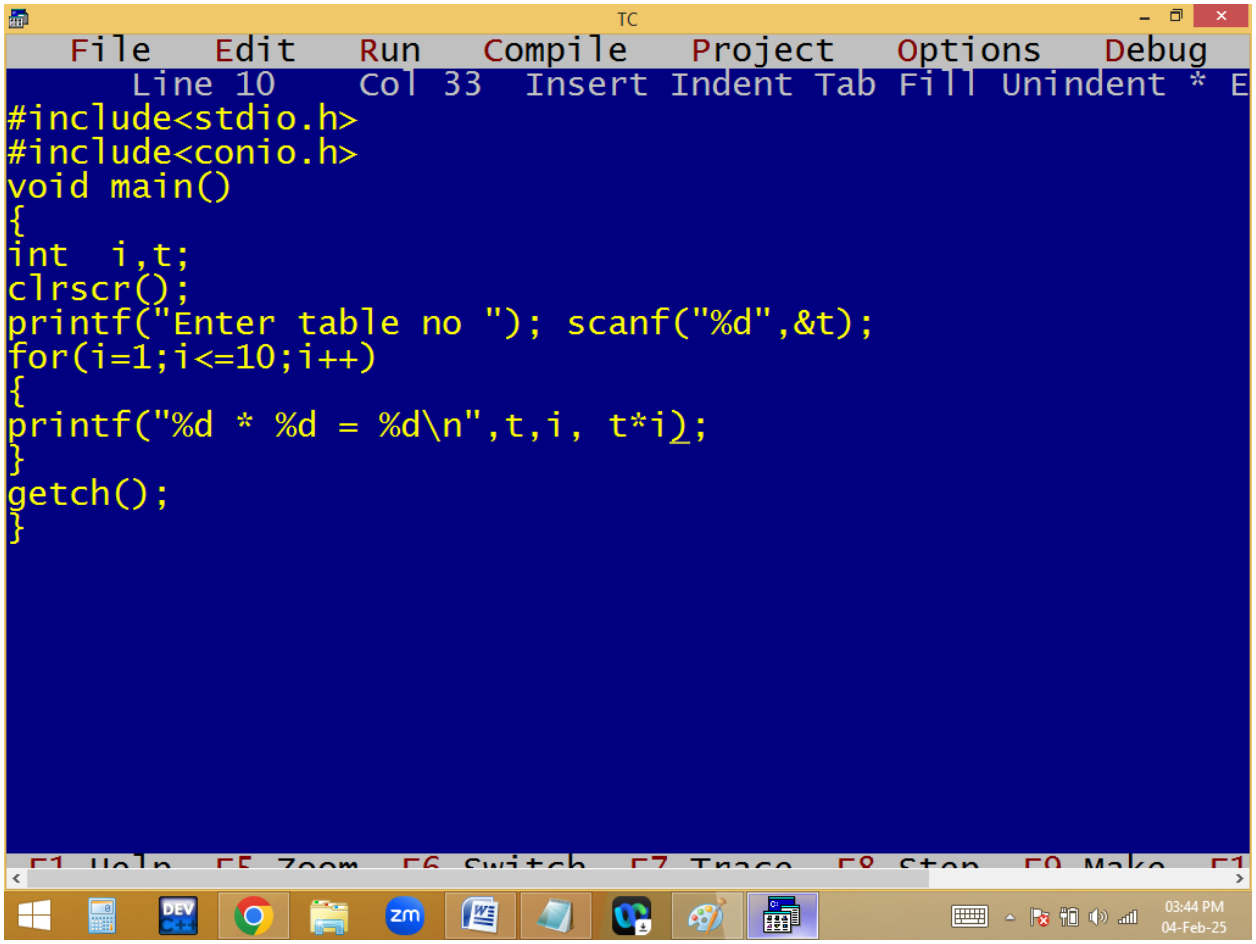
9*4=36

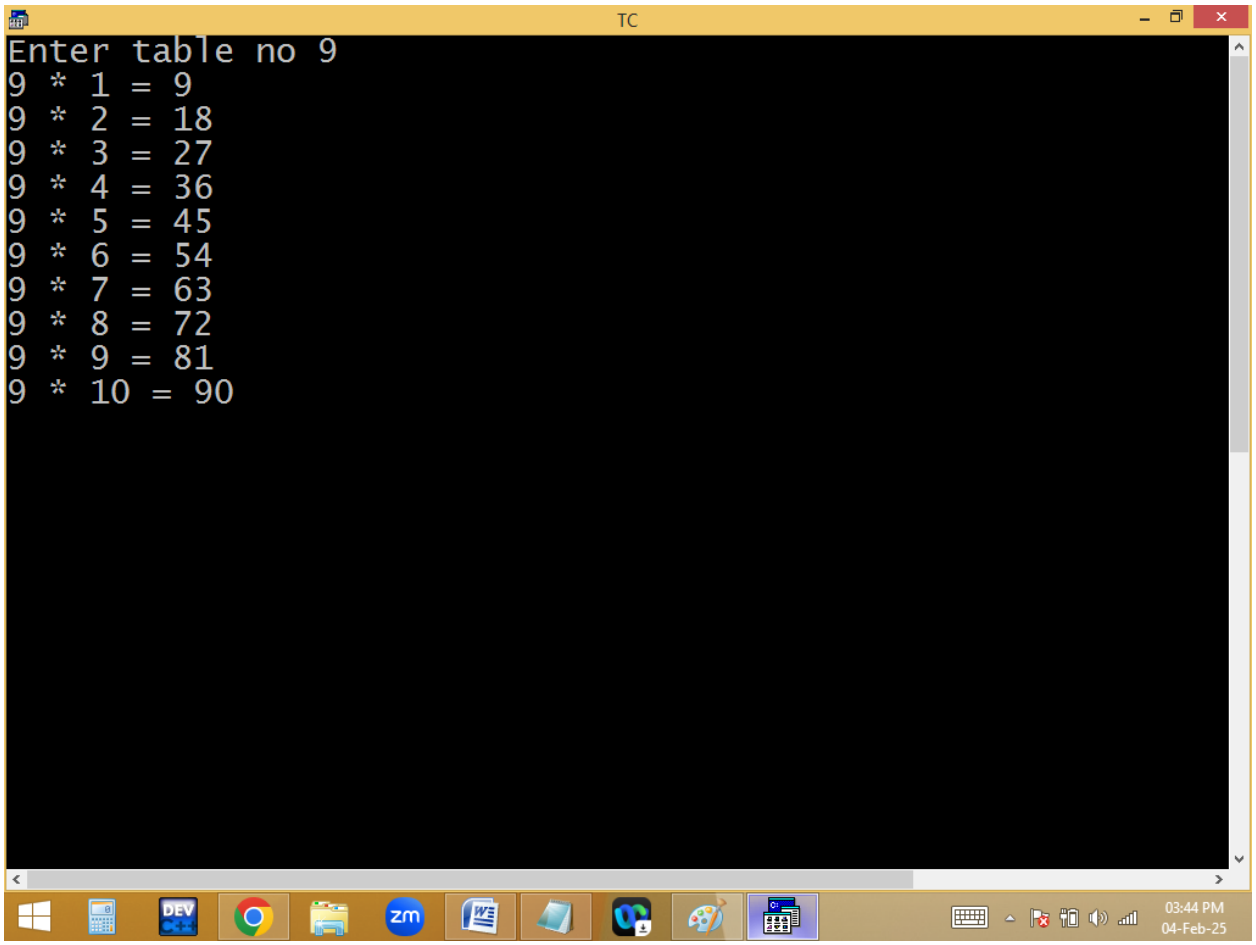
9*5=45

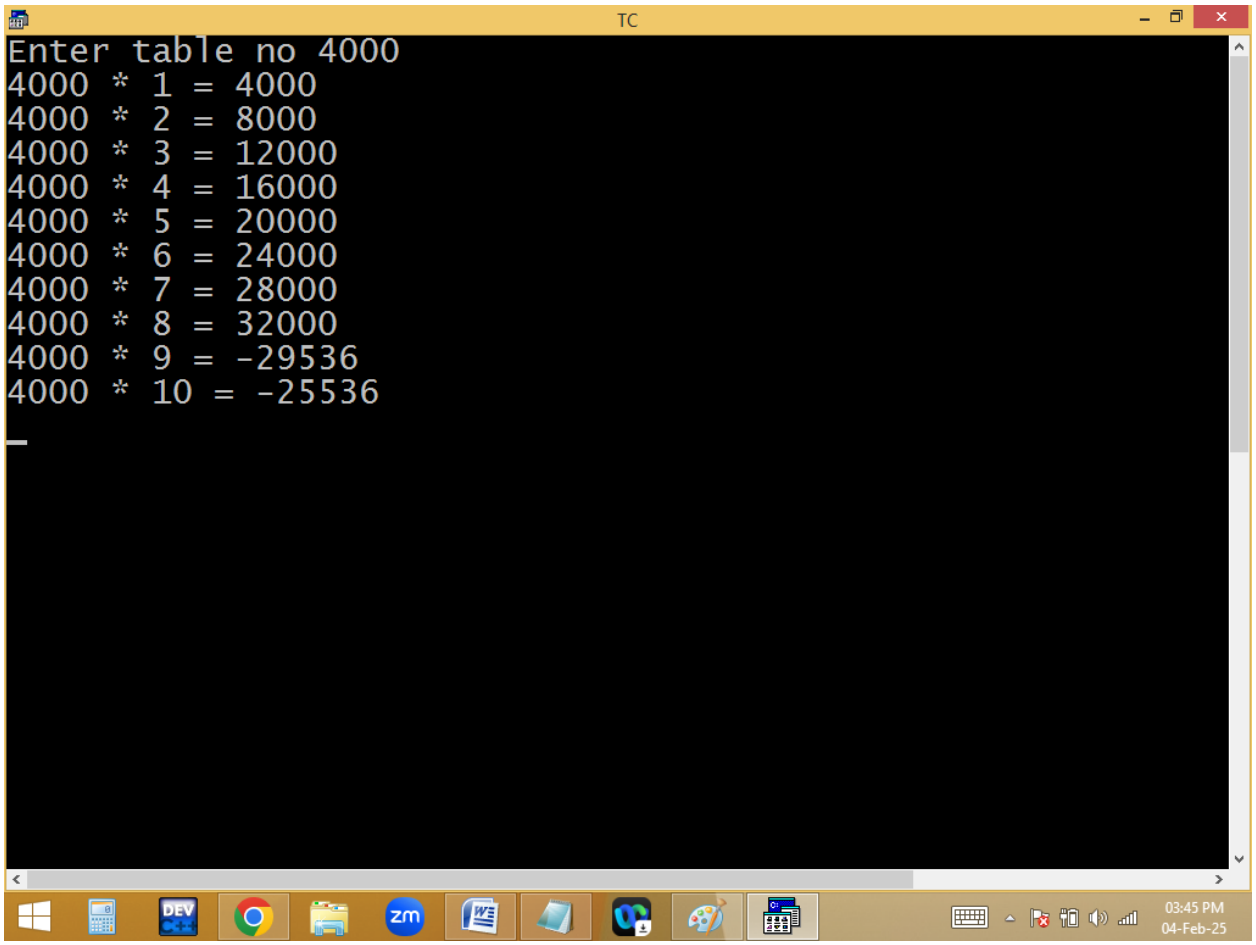
..

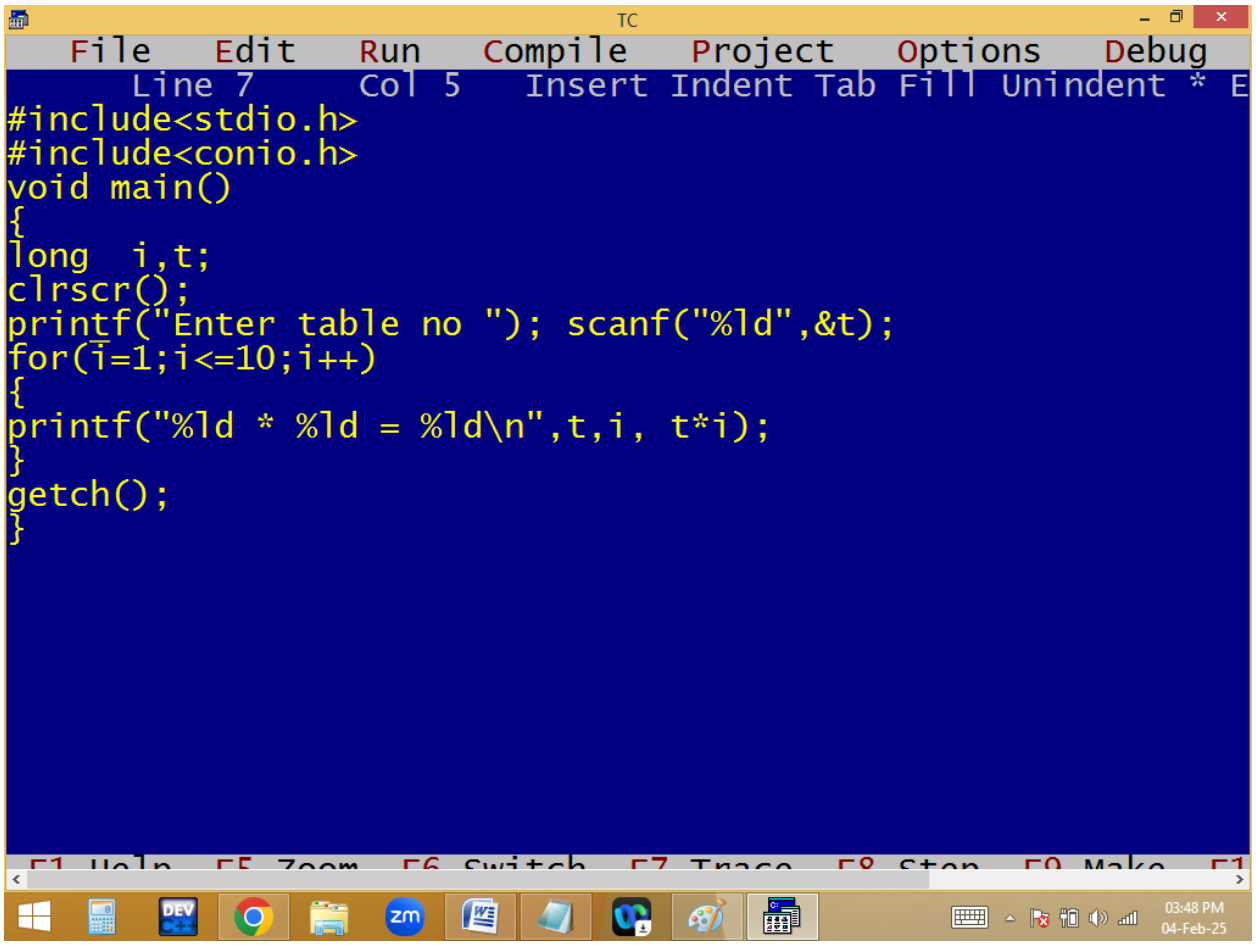
..

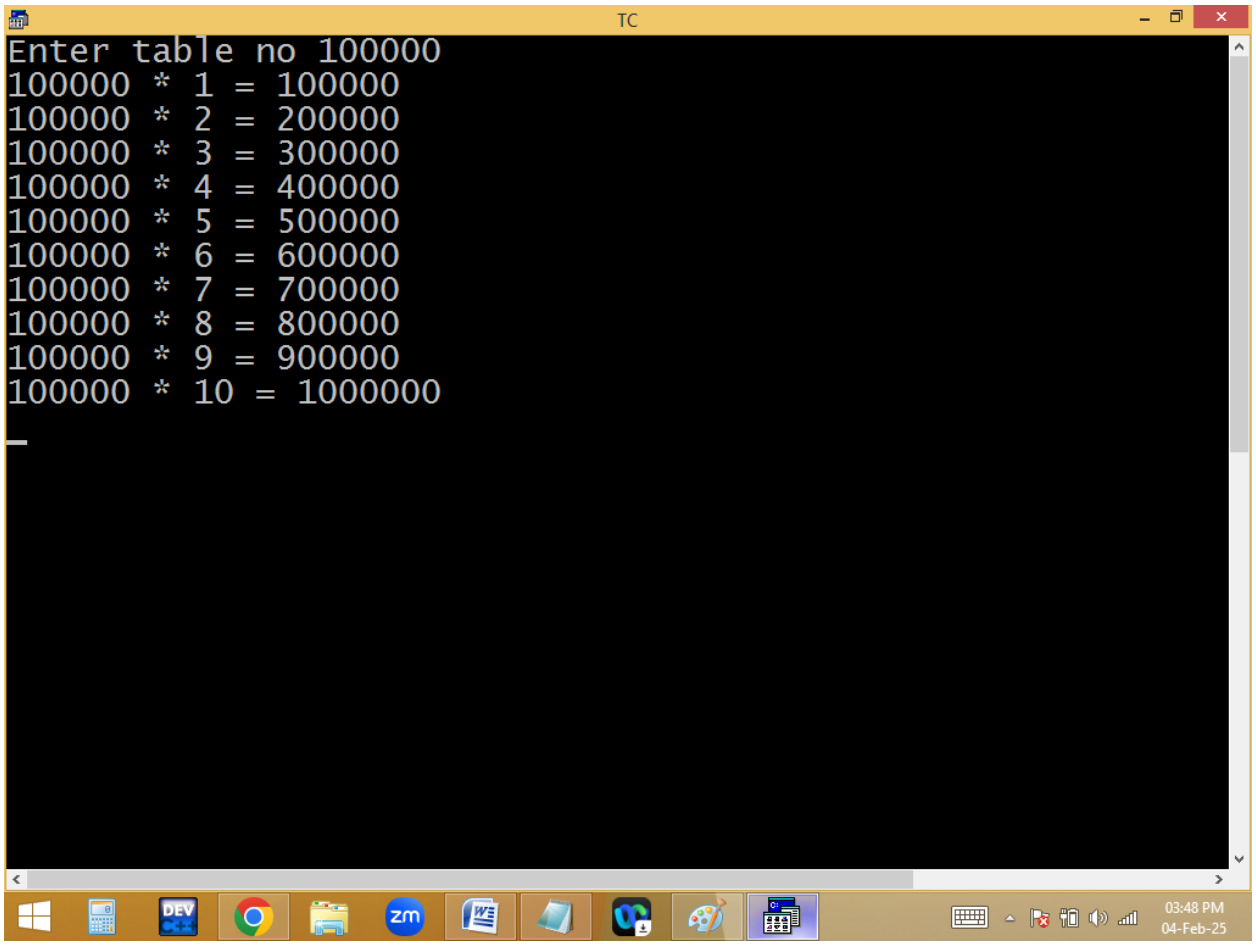
9*10=90

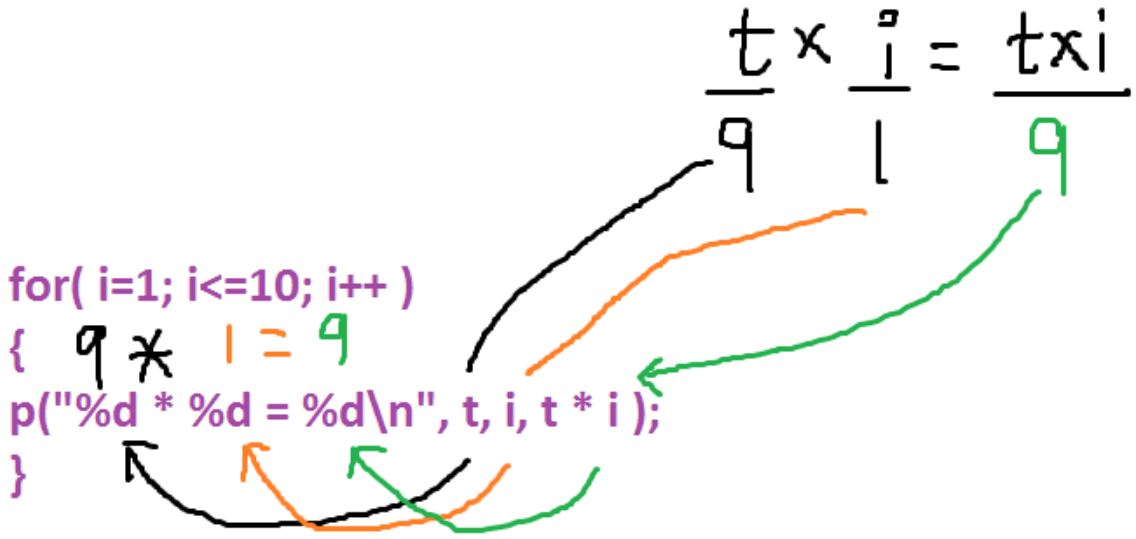

TC
File Edit Run Compile Project Options Debug
Line 10 Col 33 Insert Indent Tab Fill Unindent * E
#include<stdio.h>
#include<conio.h>
void main()
{
int i,t;
clrscr();
printf("Enter table no "); scanf("%d",&t);
for(i=1;i<=10;i++)
{
printf("%d * %d = %d\n",t,i, t*i);
}
getch();
}
03:44 PM
04-Feb-25


TC
Enter table no 9
9 * 1 = 9
9 * 2 = 18
9 * 3 = 27
9 * 4 = 36
9 * 5 = 45
9 * 6 = 54
9 * 7 = 63
9 * 8 = 72
9 * 9 = 81
9 * 10 = 90
03:44 PM
04-Feb-25


TC
Enter table no 4000
4000 * 1 = 4000
4000 * 2 = 8000
4000 * 3 = 12000
4000 * 4 = 16000
4000 * 5 = 20000
4000 * 6 = 24000
4000 * 7 = 28000
4000 * 8 = 32000
4000 * 9 = -29536
4000 * 10 = -25536
03:45 PM
04-Feb-25


TC
File Edit Run Compile Project Options Debug
Line 7 Col 5 Insert Indent Tab Fill Unindent * E
#include<stdio.h>
#include<conio.h>
void main()
{
long i,t;
clrscr();
printf("Enter table no "); scanf("%ld",&t);
for(i=1;i<=10;i++)
{
printf("%ld * %ld = %ld\n",t,i, t*i);
}
getch();
}
03:48 PM
04-Feb-25


TC
Enter table no 100000
100000 * 1 = 100000
100000 * 2 = 200000
100000 * 3 = 300000
100000 * 4 = 400000
100000 * 5 = 500000
100000 * 6 = 600000
100000 * 7 = 700000
100000 * 8 = 800000
100000 * 9 = 900000
100000 * 10 = 1000000
03:48 PM
04-Feb-25


t x i = txi
9 1 9
for(i=1; i<=10; i++)
{ 9 * 1 = 9
p("%d * %d = %d\n", t, i, t * i);
}

Finding Armstrong no

1 is a single digit no → $1^1=1$

2 is a single digit no → $2^1=2$

9 is a single digit no → $9^1=9$

153 is a three digit no → $1^{3+}5^{3+}3^3=1+125+27=153$

370, 371, 407, 1634, 8208,…

$1634 + 1^4 + 6^4 + 3^4 + 4^4 = 1634$