

```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 33 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
long int n;
clrscr();
printf("Enter n value "); scanf("%ld",&n);
if(n<0)printf("-",n=-n);
do
{
if(n>9 && n%100<10) printf("0");
printf("%-4d",n%100);n/=100;}while(n!=0);
getch();
}

Enter n value 100000
00 00 10 _
```

12345678 ==> 78 56 34 12

1234%100=34

1234/100=12

Finding perfect no: sum of factors is equal to given no.

6 factors are $1 + 2 + 3 = 6$

28 factors are $1 + 2 + 4 + 7 + 14 = 28$

$$6\%1=0$$

$$6\%2=0$$

$$6\%3=0$$

$$6\%4=2$$

$$6\%5=1$$

$$6\%6=0$$

$$1+2+3=6$$

$$2 \div 1 = 2$$

~~$$2 \div 2 = 1$$~~

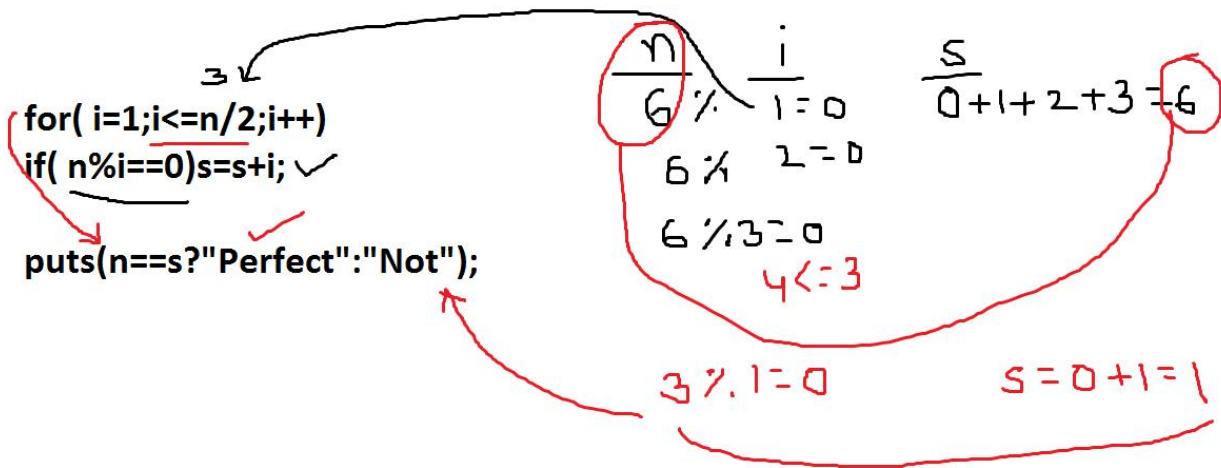
```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 44 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n, i, s=0;
clrscr();
printf("Enter n value "); scanf("%ld",&n);
for(i=1;i<=n/2;i++) if(n%i==0)s+=i;
puts(n==s?"Perfect no":"Not a Perfect no");
getch();
}
```

Enter n value 6
Perfect no

TC

```
TC
Enter n value 28
Perfect no
```

```
TC
Enter n value 4
Not a Perfect no
```



Finding prime/composite no: When a no is having 2 factors it is a prime / the no divisible with 1 and itself is called prime.

2 factors are 1 and 2 \Rightarrow 2 factors \leftarrow prime

3 factors are 1 and 3 \Rightarrow 2 factors \leftarrow prime

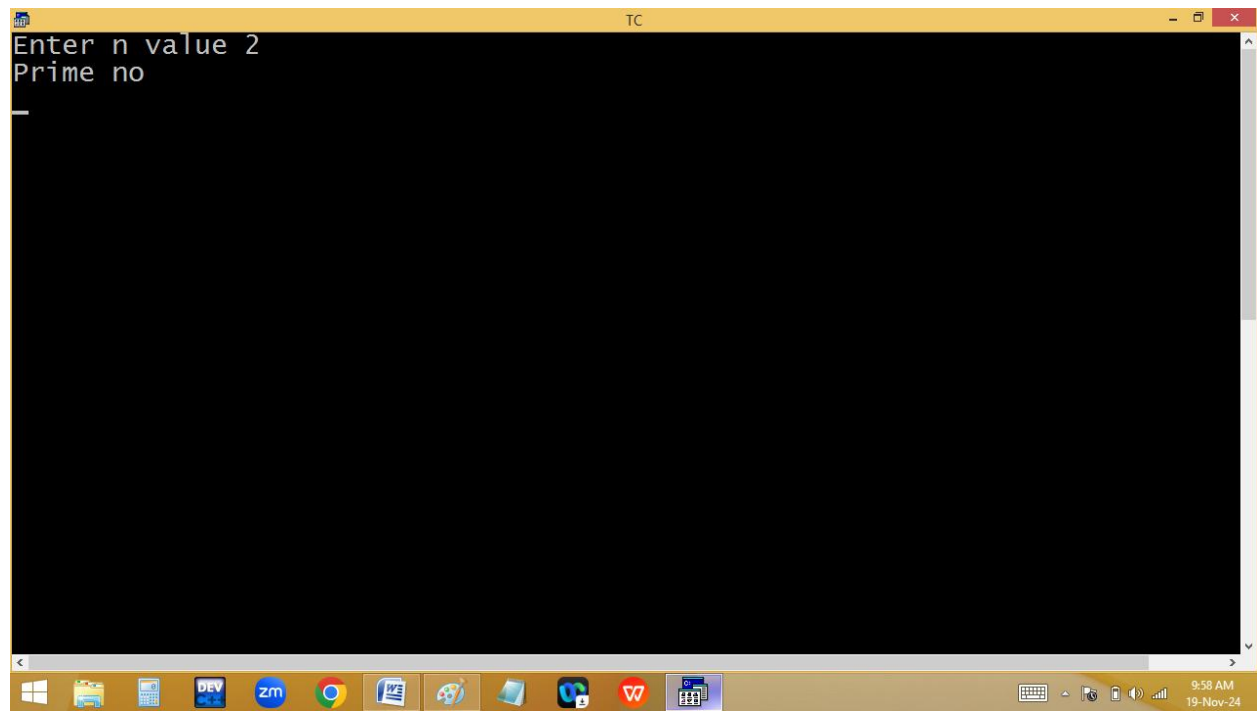
4 factors are 1, 2, 4 \Rightarrow 3 factors \leftarrow composite no

1 divisible with 1 and itself also \Rightarrow not a prime/composite no

The image shows two windows from the Turbo C++ (TC) IDE. The top window is the code editor, displaying a C program that checks if a number is prime or composite. The code includes `<stdio.h>` and `<conio.h>`, and uses `scanf` to get input, `printf` for prompts, and `puts` for output. A `for` loop checks divisibility from 1 to `n`. The bottom window is the output console, showing the program's execution with the input '1' and the output 'Not a prime/composite no'.

```
File Edit Run Compile Project Options Debug Break/watch
Line 13 Col 2 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n, i, c=0;
clrscr();
printf("Enter n value "); scanf("%d",&n);
if(n==1)puts("Not a prime/composite no");
else
{
for(i=1;i<=n;i++) if(n%i==0)c++;
puts(c==2?"Prime no":"Composite no");
}
getch();
}
```

Enter n value 1
Not a prime/composite no




```
TC
Enter n value 3
Prime no
```

```
TC
Enter n value 4
Composite no
```

for(i=1; i<=n; i++)
 if(n%i==0)c++;

puts(c==2?"Prime":"Com");

| n | i | c |
|---|---|---|
| 2 | 1 | 0 |
| 2 | 2 | 1 |

3 ← 2

| | | | |
|---|---|---|---|
| 4 | 1 | 0 | 1 |
| 4 | 2 | 0 | 2 |
| 4 | 4 | 0 | 3 |

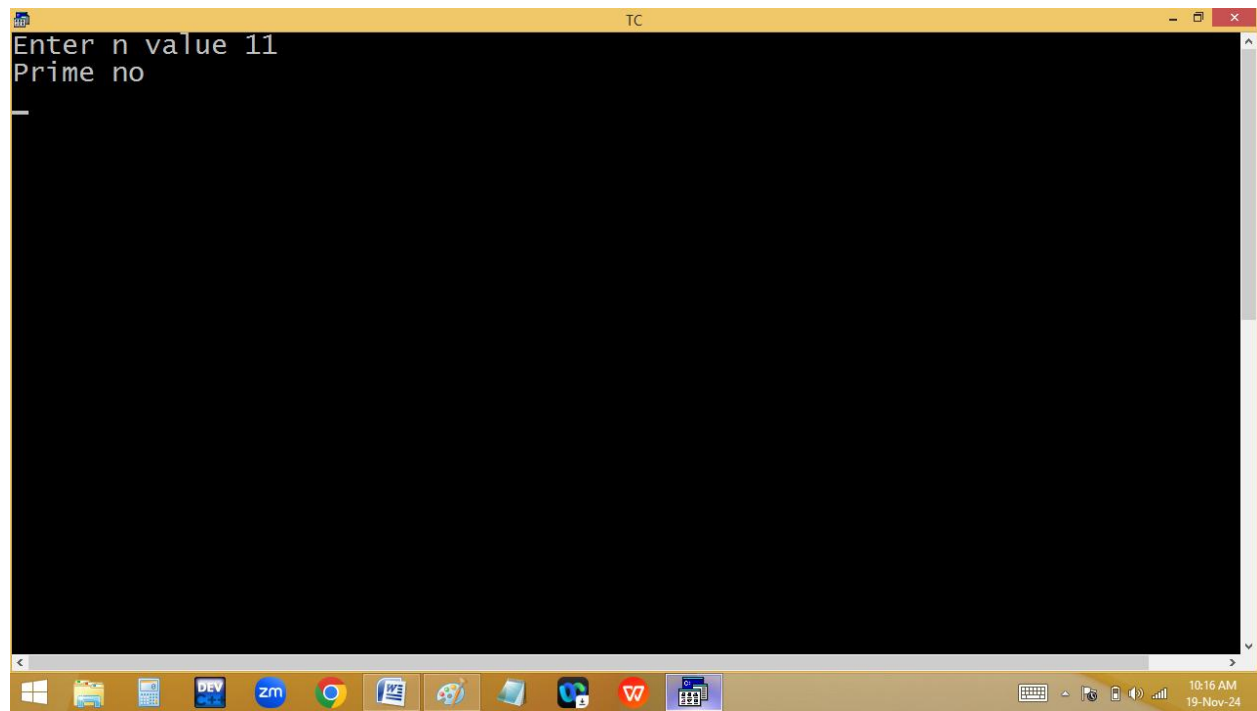
Method2:

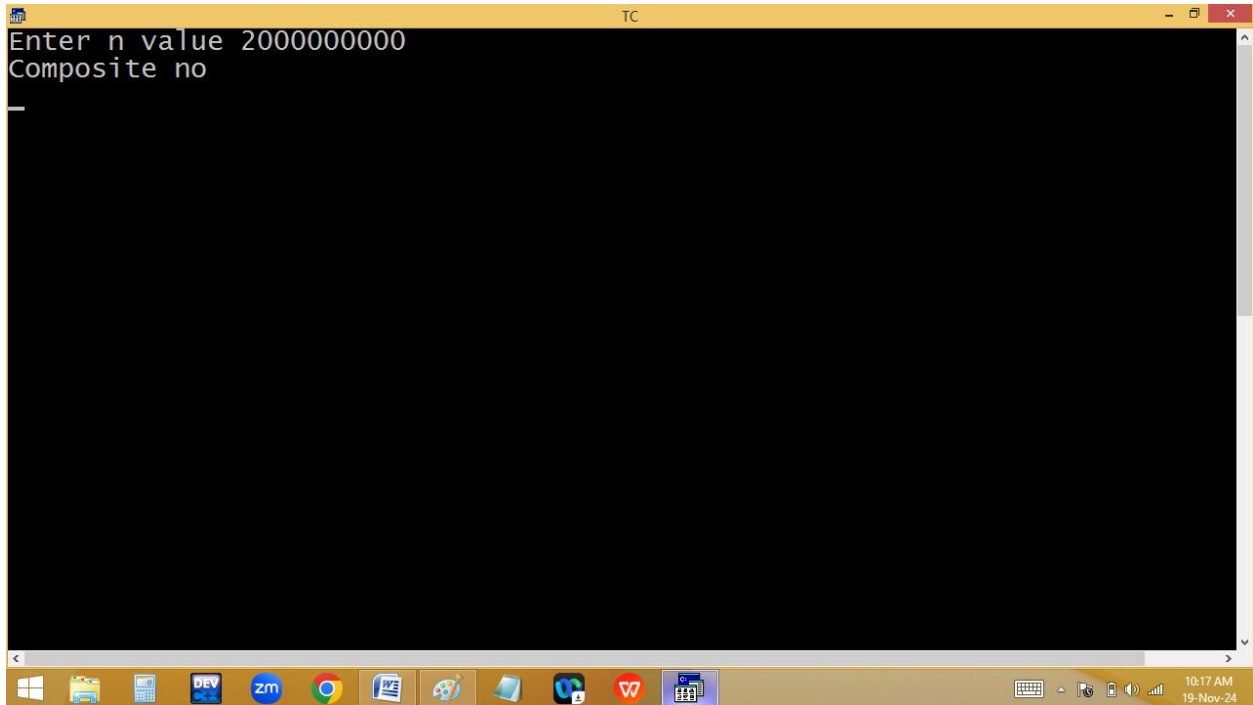
```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 16 Col 2 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
long int n, i;
clrscr();
printf("Enter n value "); scanf("%ld",&n);
if(n==1)puts("Not a prime/composite no");
else
{
for(i=2;i<=n/2;i++)
{
if(n%i==0){puts("Composite no");getch();return;}
}
puts("Prime no");
}
getch();
}
```

Enter n value 1
Not a prime/composite no

```
TC
Enter n value 2
Prime no
```

```
TC
Enter n value 4
Composite no
```





10 - ~~1~~ 2 5 ~~10~~
 20 - ~~1~~ 2 4 5 10 ~~20~~
 100 - ~~1~~ 2 4 5 10 20 25 50 ~~100~~
 5 - ~~1~~ ~~5~~

```

for( i=2; i<=n/2;i++)
{
    if( n%i==0 ) {p(com); return;}
}
p(prime);
  
```

$$\frac{n}{10} \% 2 = 0$$

| | | | | |
|----|---|---|---|---|
| 11 | / | 2 | = | 1 |
| 11 | / | 3 | = | 2 |
| 11 | / | 4 | = | 3 |
| 11 | / | 5 | = | 1 |

$$6 \leq 11 / 2 = 5$$

Fibonacci series:

$n=5 \rightarrow 0 \ 1 \ 1 \ 2 \ 3$

```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 15 Col 1 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n, i, f1=0, f2=1, f3;
clrscr();
printf("Enter n value "); scanf("%d",&n);
for(i=1;i<=n;i++)
{
printf("%4d",f1);
f3=f1+f2;
f1=f2;
f2=f3;
}
getch();
}
```

Enter n value 5
0 1 1 2 3_

```

Enter n value 10
0 1 1 2 3 5 8 13 21 34_

```

```

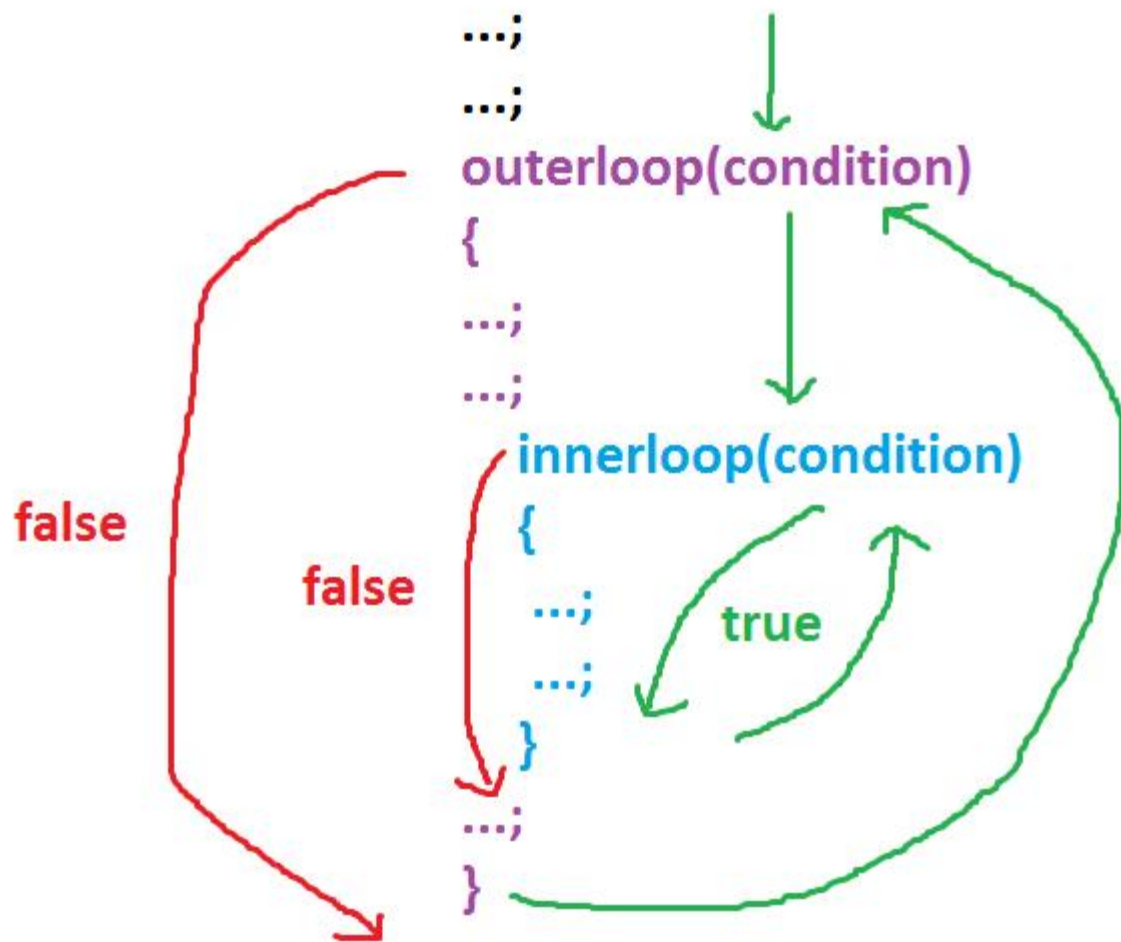
for( i=1; i<=n; i++ )
{
  p(f1); ✓
  f3=f1+f2;
  f1=f2; f2=f3;
}

```

| $i \leq n$ | f_1 | f_2 | f_3 |
|------------|-------|-------|-------|
| 1 | ✓ 0 | + | 1 = 1 |
| 5 | ✓ 1 | + | 1 = 2 |
| | ✓ 1 | + | 2 = 3 |
| | ✓ 2 | + | 3 = 5 |
| | ✓ 3 | | 5 |

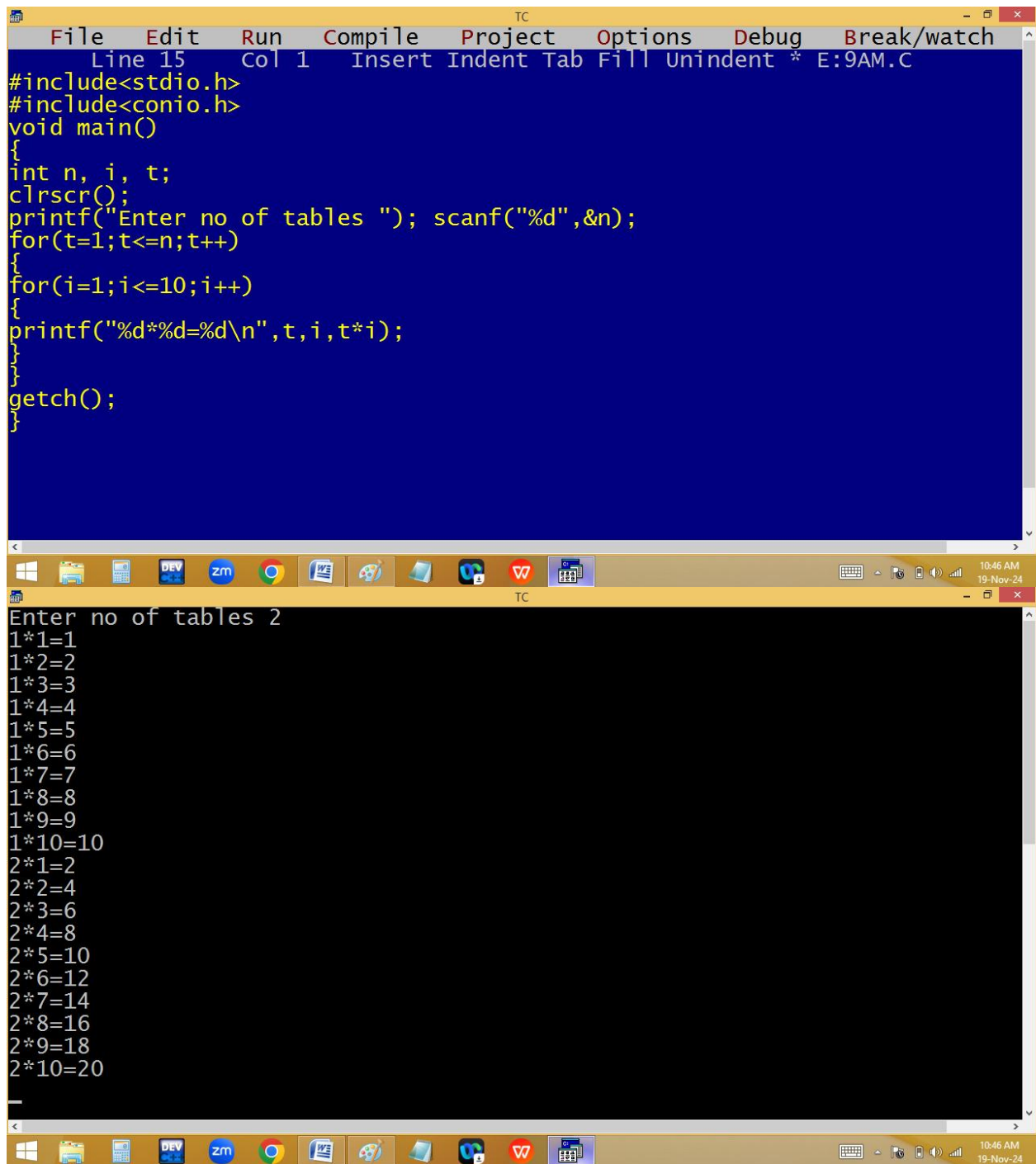
0 1 1 2 3

Nested loops: Loop within loop



Printing n tables:

3 tables



```
File Edit Run Compile Project Options Debug Break/watch
Line 15 Col 1 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n, i, t;
clrscr();
printf("Enter no of tables "); scanf("%d",&n);
for(t=1;t<=n;t++)
{
for(i=1;i<=10;i++)
{
printf("%d*%d=%d\n",t,i,t*i);
}
}
getch();
}
```

Enter no of tables 2

1*1=1
1*2=2
1*3=3
1*4=4
1*5=5
1*6=6
1*7=7
1*8=8
1*9=9
1*10=10
2*1=2
2*2=4
2*3=6
2*4=8
2*5=10
2*6=12
2*7=14
2*8=16
2*9=18
2*10=20

Home work: Tables side by side