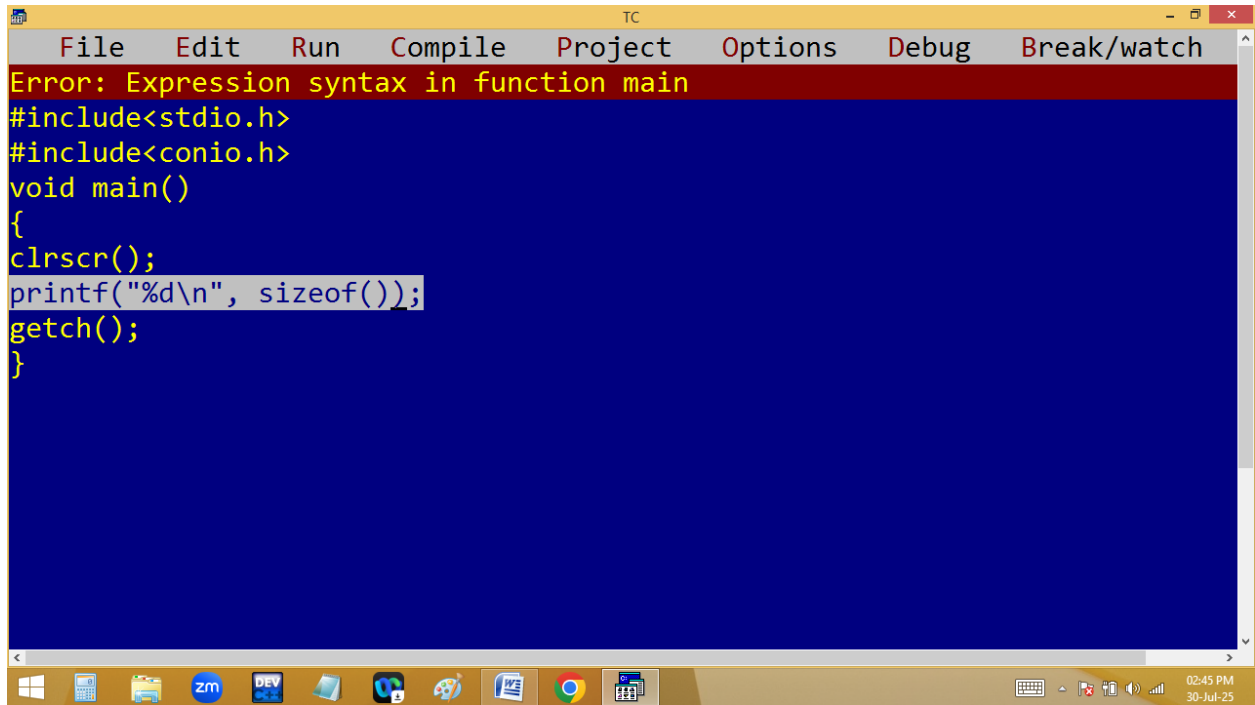


```
TC
#include<stdio.h>
#include<conio.h>
void main()
{
int a=999; clrscr();
printf("%d, %d\n",a,sizeof(++a));
printf("%d, %d\n",a, sizeof(a=555));
printf("%d, %d\n", sizeof("Kishore"), printf("Kishore"));
printf("%d, %d\n", sizeof("Kishore\\0"), printf("Kishore\\0"));
printf("Kishore addr %u\n","Kishore");
printf("%d, %d\n", sizeof("Kishore"+1), sizeof("Kishore")+1);
printf("%d, %d\n", sizeof("1.23"), sizeof(sizeof("Kishore")));
printf("%d, %d\n", sizeof(""), sizeof(" "));
getch();
}
/* sizeof() always returns the size only not the operation result
i.e. sizeof() never consider the expressions*/

999, 2
999, 2
Kishore8, 7
Kishore9, 7
Kishore addr 470
2, 9
5, 2
1, 2
_

TC
02:43 PM
30-Jul-25
```

A screenshot of the Turbo C++ (TC) IDE. The window title is 'TC'. The menu bar includes 'File', 'Edit', 'Run', 'Compile', 'Project', 'Options', 'Debug', and 'Break/watch'. A red error message banner at the top reads 'Error: Expression syntax in function main'. The code editor has a blue background and contains the following C code:

```
#include<stdio.h>
#include<conio.h>
void main()
{
clrscr();
printf("%d\n", sizeof());
getch();
}
```

The line 'printf("%d\n", sizeof());' is highlighted. The Windows taskbar is visible at the bottom with various icons and a system clock showing 02:45 PM on 30-Jul-25.

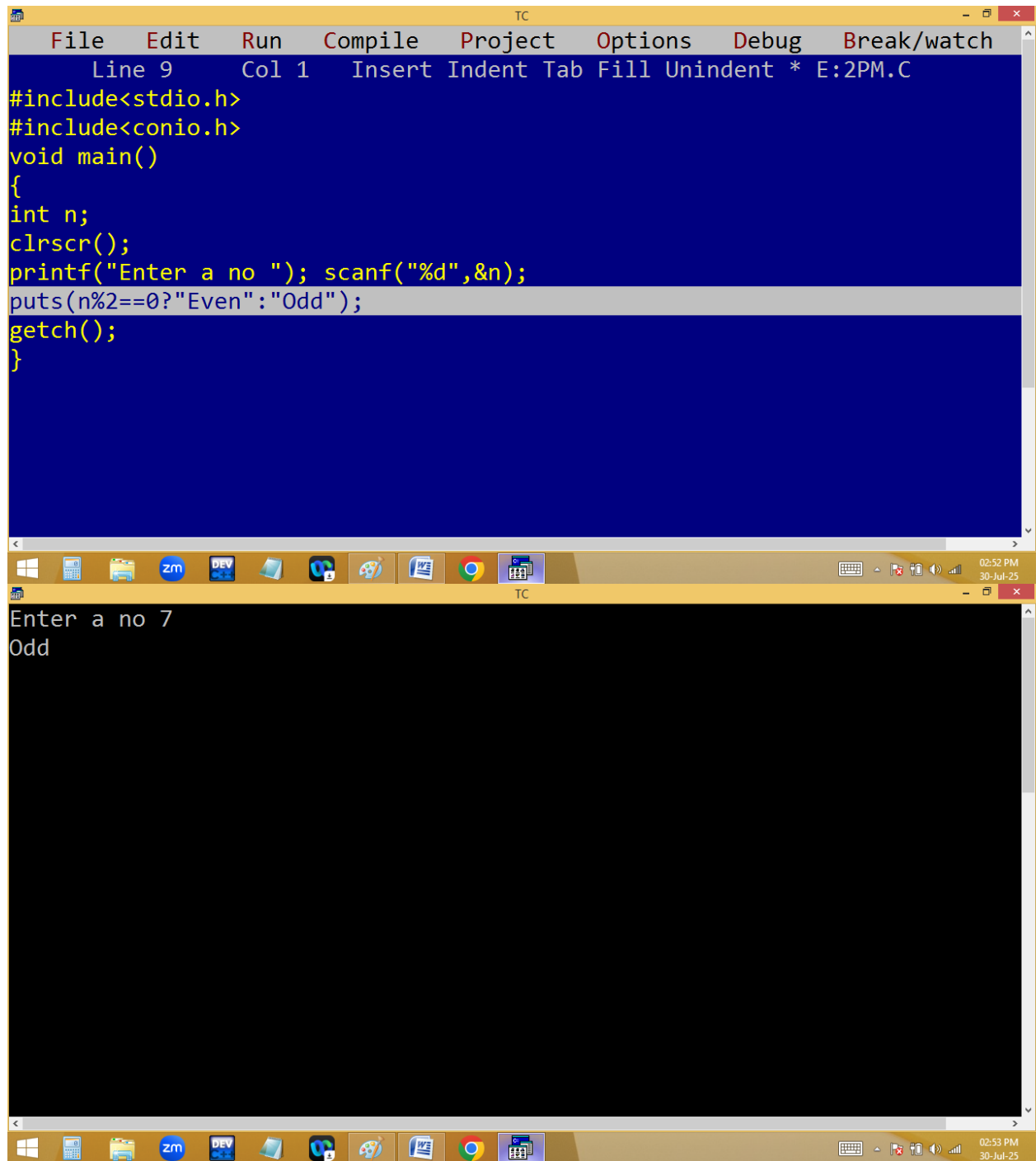
Ternary / conditional operator [?:]

It require three operands. It is going to start with condition. Hence it is also called conditional operator.

Syntax:

Conditional part ? true part : false part ;



Finding even/odd using ternary op?



```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 1 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n;
clrscr();
printf("Enter a no "); scanf("%d",&n);
puts(n%2==0?"Even":"Odd");
getch();
}
```

Enter a no 7
Odd

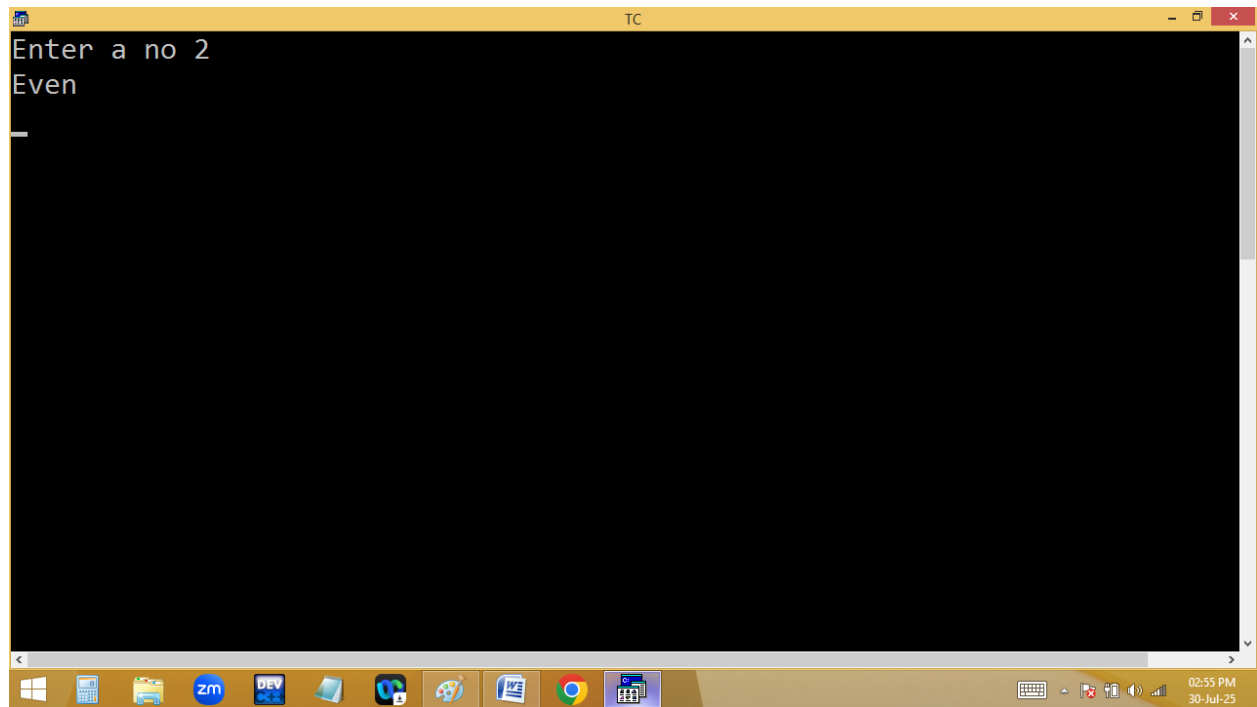
```
TC
Enter a no 8
Even
```

$8 \% 2 = 0 == 0$ 
`puts(n % 2 == 0 ? "Even" : "Odd");`
 $9 \% 2 = 1 == 0$ 

The image shows two screenshots of the Turbo C++ (TC) IDE. The top screenshot displays the source code of a C program designed to check if a number is odd or even. The code includes headers for `stdio.h` and `conio.h`, defines a `main` function, declares an integer `n`, clears the screen with `clrscr()`, prompts the user to enter a number using `printf`, reads the input with `scanf`, and prints the result using `puts` based on a modulo operation. The bottom screenshot shows the program's execution. It prompts the user to 'Enter a no' and shows the input '1'. The program then outputs 'Odd'.

```
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 22 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n;
clrscr();
printf("Enter a no "); scanf("%d",&n);
puts(n%2?"Odd":"Even");
getch();
}
```

Enter a no 1
Odd



```
Enter a no 2
Even
```

Finding +Ve / -Ve / Zero using ternary operator?

The image shows two screenshots of the Turbo C++ (TC) IDE. The top screenshot displays the source code of a C program in a blue-themed editor. The code is as follows:

```
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 32 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n;
clrscr();
printf("Enter a no "); scanf("%d",&n);
puts(n>0?" +Ve":n<0?" -Ve":"Zero");
getch();
}
```

The bottom screenshot shows the same IDE with the program executed. The output window displays the text "Enter a no 4" followed by "+Ve" on the next line, indicating that the user entered the number 4 and the program correctly identified it as positive.

```
TC
Enter a no -4
-Ve
```

```
TC
Enter a no 0
Zero
```


0 > 0 0 < 0
-3 > 0 -3 < 0
4 > 0

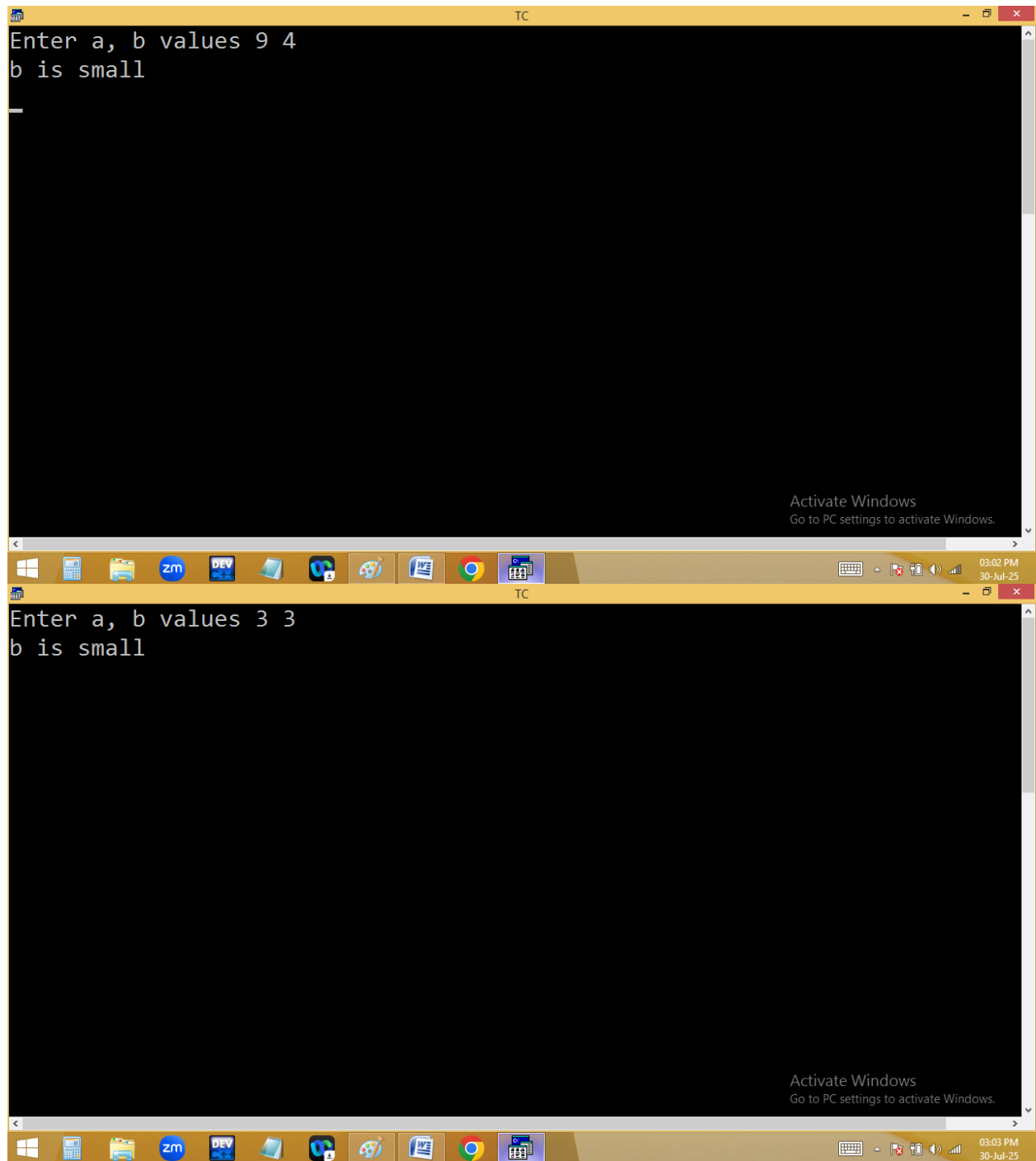
puts(n>0 ? "+Ve" : n<0 ? "-Ve" : "Zero");

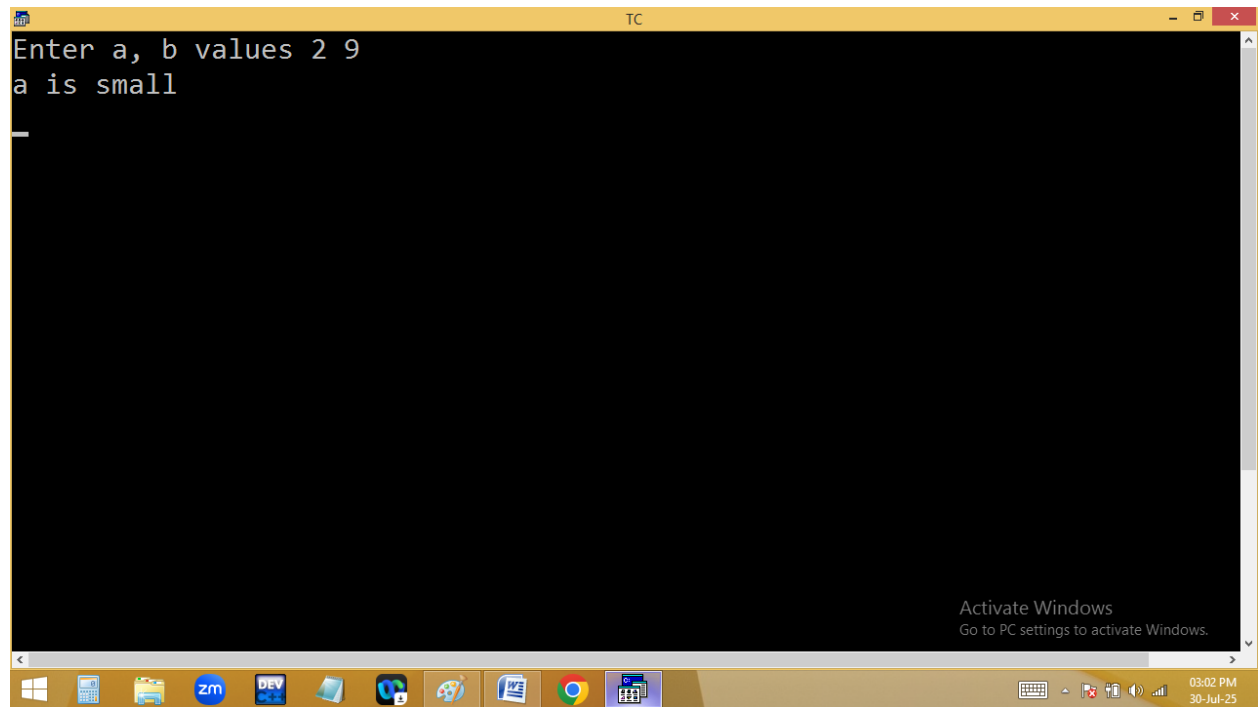
Finding min in 2 no's:

```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 34 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a, b;
clrscr();
printf("Enter a, b values "); scanf("%d %d",&a, &b);
puts(a<b?"a is small":"b is small");
getch();
}
```

Activate Windows
Go to PC settings to activate Windows.

03:02 PM
30-Jul-25





The image shows a screenshot of the Turbo C++ (TC) IDE. The top window displays the source code of a C program. The code includes headers for `stdio.h` and `conio.h`, defines a `main` function, declares two integer variables `a` and `b`, clears the screen with `clrscr()`, prompts the user to enter values for `a` and `b` using `scanf`, and then prints a conditional message using `puts` based on the comparison of `a` and `b`. The program ends with `getch()` to pause execution.

```
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 56 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a, b;
clrscr();
printf("Enter a, b values "); scanf("%d %d",&a, &b);
puts(a<b?"a is small":b<a?"b is small":"Both are equal");
getch();
}
```

The bottom window shows the output of the program. It displays the prompt "Enter a, b values" followed by the user input "3 3". The program then outputs "Both are equal".

```
Enter a, b values 3 3
Both are equal
```

The Windows taskbar at the bottom shows the time as 03:05 PM on 30-Jul-25. A watermark "Activate Windows" is visible in the bottom right corner of both windows.

$3 < 8$ ✓
`puts(a < b ? "a is small": b < a ? "b is small": "equal");`
 $9 < 4$
 $3 < 3$ ✓

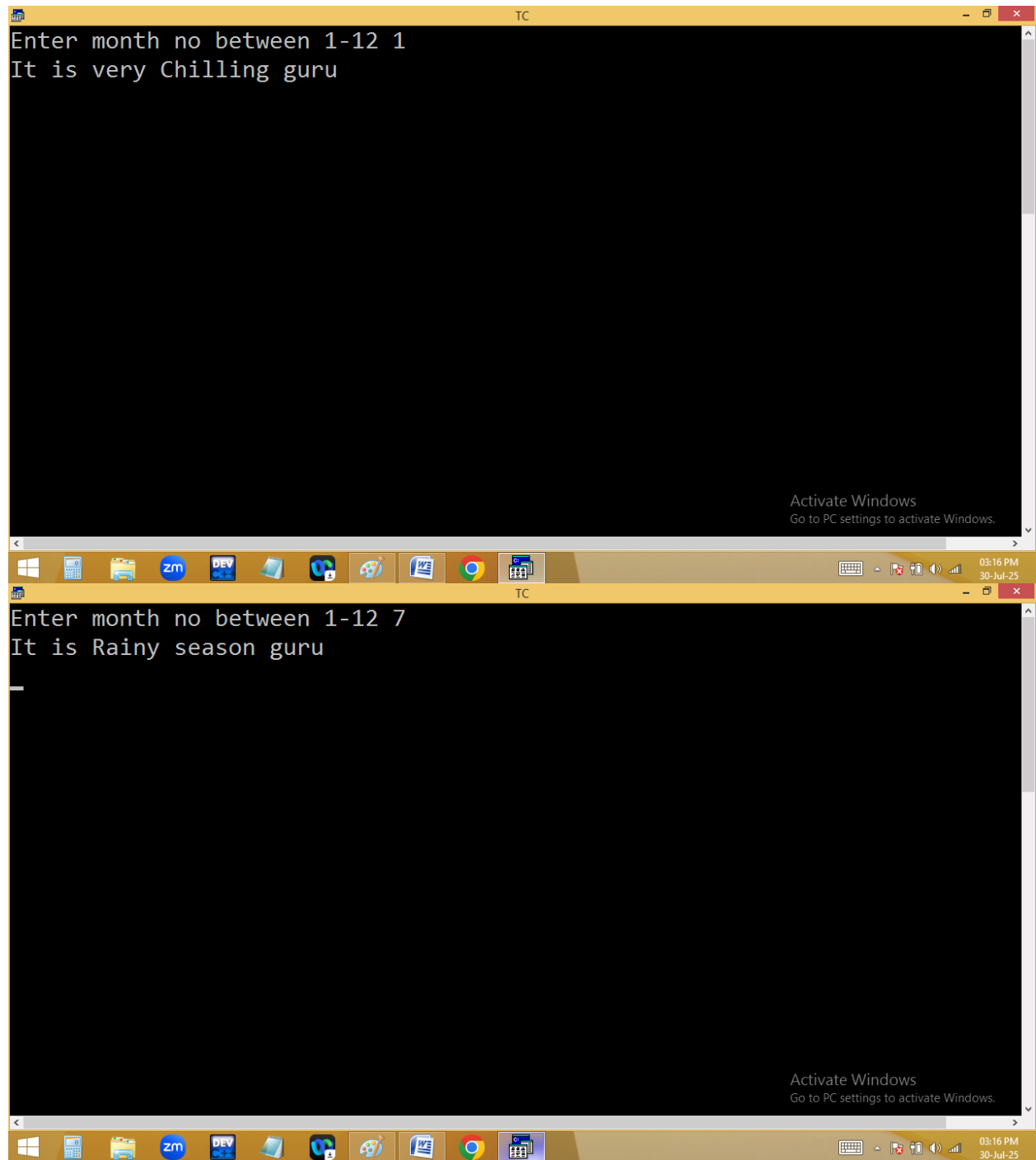
The image shows a screenshot of the Turbo C++ (TC) IDE. The top window displays the source code of a C program. The code includes headers for `stdio.h` and `conio.h`, and defines a `main` function. Inside `main`, it declares an integer `m`, clears the screen with `clrscr()`, and prompts the user to enter a month number between 1 and 12. It then uses `scanf` to read the input. A conditional statement `puts(m<1 || m>12 ? "Invalid month":m>=3 && m<=6 ? "It is very hot guru" : m>=7 && m<=10 ? "It is Rainy season guru" : "It is very Chilling guru");` checks the input and prints the corresponding season. Finally, it calls `getch()` to pause the program.

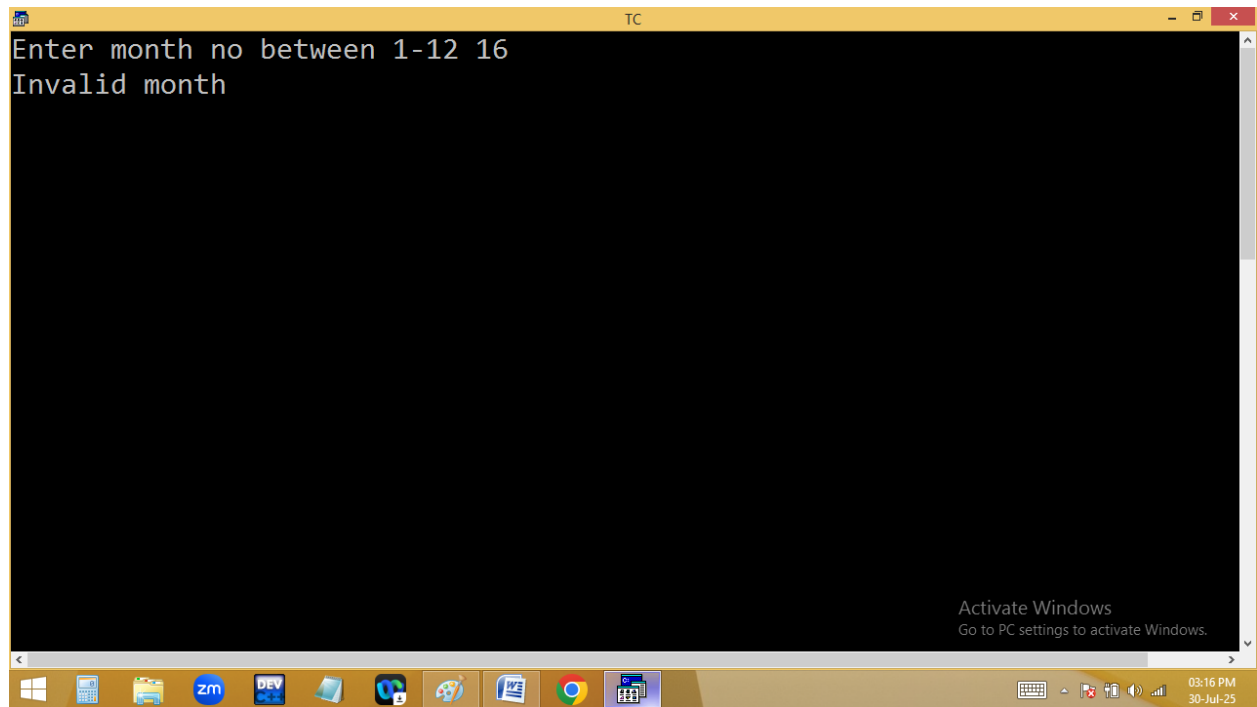
```
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 18 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int m;
clrscr();
printf("Enter month no between 1-12 "); scanf("%d",&m);
puts(m<1 || m>12 ? "Invalid month":m>=3 && m<=6 ? "It is very hot guru"
: m>=7 && m<=10 ? "It is Rainy season guru" : "It is very Chilling guru");
getch();
}
```

The bottom window shows the program's execution. It displays the prompt "Enter month no between 1-12" followed by the user input "5". The program then outputs "It is very hot guru".

```
Enter month no between 1-12 5
It is very hot guru
```

Both windows include a Windows taskbar at the bottom with various application icons and a system clock showing 03:15 PM and 03:16 PM on 30-Jul-25. An "Activate Windows" watermark is visible in the bottom right corner of each window.





```
TC
Enter month no between 1-12 16
Invalid month

Activate Windows
Go to PC settings to activate Windows.

03:16 PM
30-Jul-25
```

Finding given char is alphabet or not using ternary operator?

The image shows two screenshots of the Turbo C++ (TC) IDE. The top screenshot displays the source code of a C program in a blue editor window. The code includes headers for `stdio.h` and `conio.h`, and defines a `main` function that prompts the user for a character and checks if it is an alphabet. The bottom screenshot shows the same IDE with the program's output, where the user has entered 'H' and the program has responded that it is an alphabet. Both windows have a yellow title bar and a menu bar with options like File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. The Windows taskbar at the bottom shows various application icons and the system clock.

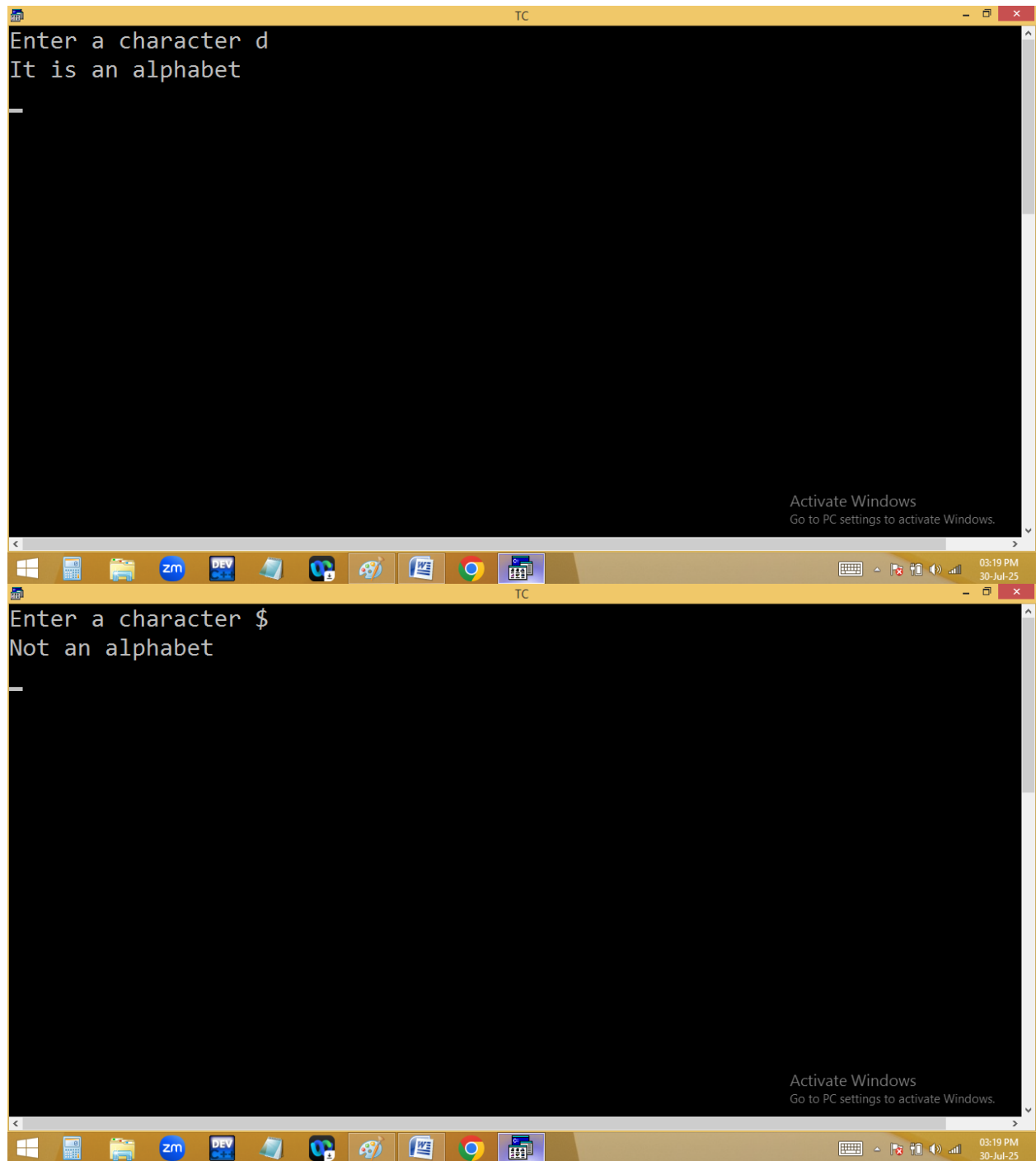
```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 1 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
char ch;
clrscr();
printf("Enter a character "); scanf("%c",&ch);
puts(ch>='a'&&ch<='z' || ch>='A'&&ch<='Z'? "It is an alphabet":
"Not an alphabet");
getch();
}

Activate Windows
Go to PC settings to activate Windows.
```

TC

Enter a character H
It is an alphabet

Activate Windows
Go to PC settings to activate Windows.



Bitwise operators?

They work on bits [0,1].

It is one of the low level feature of c.

They work on int data only.

C comes with **16 bit compiler**. Hence in c language the bitwise operators are limited from 2^0 to 2^{15}

We have to take only the 1's position, not the 0 positions. Because of any number * 0 gives again 0.

When starting bit is 0, the given no is positive no. if the starting bit 1 it is a negative no. hence the starting big is also called **sign bit**.

The bitwise operators very much used in os, device drivers, translators development.

They are faster than normal operators.

C comes with 6 bitwise operators?

1. & - bitwise and
2. | - bitwise or
3. ^ - Xor – Exclusive Or

4. ~ - compliment op

5. << - left shift

6. >> - right shift

& - bitwise and: When both bits are 1 then result bit also 1 otherwise result is 0.

Eg: **25 & 15 = 9**

$$25 \& 15 = 9$$

$$\begin{array}{r} 2 \overline{) 25} \\ 2 \overline{) 12-1} \\ 2 \overline{) 6-0} \\ 2 \overline{) 3-0} \\ 1-1 \\ \hline 11001 \end{array}$$

$$\begin{array}{r} 2 \overline{) 15} \\ 2 \overline{) 7-1} \\ 2 \overline{) 3-1} \\ 1-1 \\ \hline 1111 \end{array}$$

The image shows two windows of the Turbo C++ (TC) IDE. The top window is the source code editor for a file named E:2PM.C. It contains a C program that uses printf to display the bitwise AND, OR, and XOR of 25 and 15. The bottom window shows the output of the program, which displays the results 9, 31, and 22 on separate lines. Both windows have a taskbar at the bottom with various application icons and a system clock showing 03:51 PM and 03:52 PM on 30-Jul-25.

```
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 20 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
clrscr();
printf("%d\n", 25 & 15);
printf("%d\n", 25 | 15);
printf("%d\n", 25 ^ 15);

getch();
}
```

9
31
22

$$25 \wedge 15 =$$

$$25 = 0000\ 0000\ 0001\ 1001$$

$$15 = 0000\ 0000\ 0000\ 1111$$

$$\underline{0000\ 0000\ 0001\ 1111}$$

$$16 + 4 + 2 = 22$$

$$\begin{array}{r} 2 \overline{) 25} \end{array}$$

$$\begin{array}{r} 2 \overline{) 12-1} \end{array}$$

$$\begin{array}{r} 2 \overline{) 6-0} \end{array}$$

$$\begin{array}{r} 2 \overline{) 3-0} \end{array}$$

$$1-1$$

$$11001$$

$$\begin{array}{r} 2 \overline{) 15} \end{array}$$

$$\begin{array}{r} 2 \overline{) 7-1} \end{array}$$

$$\begin{array}{r} 2 \overline{) 3-1} \end{array}$$

$$1-1$$

$$1111$$