

goto label / jumping statement

It is used to transfer program execution from one place to another place [label].

In this process it is jumping from one area to another without any condition. Hence it is also called **unconditional** jumping statement.

Syntax:

```
.....;  
.....;  
goto label;  
.....;  
.....;  
label:  
.....;  
.....;
```

Here **goto** is a keyword.

Label is an identifier is used to identify the area[line].

Every label should be end with **:** (**colon**)

Keywords not allowed in labels i.e. label should be user defined.

Duplicate labels not allowed.

There is no space between go and to.

Label naming rules are similar to the identifier rules.

Note: goto label working style is similar to loops some times.

The image shows a screenshot of the Turbo C++ (TC) IDE. The top window is the source code editor, titled 'E:2PM.C', displaying a C program that uses goto statements to print a sequence of text. The code is as follows:

```
Line 15 Col 6 Insert Indent Tab Fill Unindent E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
clrscr();
goto a;
c:
puts("Hyd"); goto last;
b:
puts("Ameerpet");
goto c;
a:
puts("Naresh IT");
goto b;
last:_
getch();
}
```

The bottom window is the output console, titled 'TC', which shows the execution results of the program:

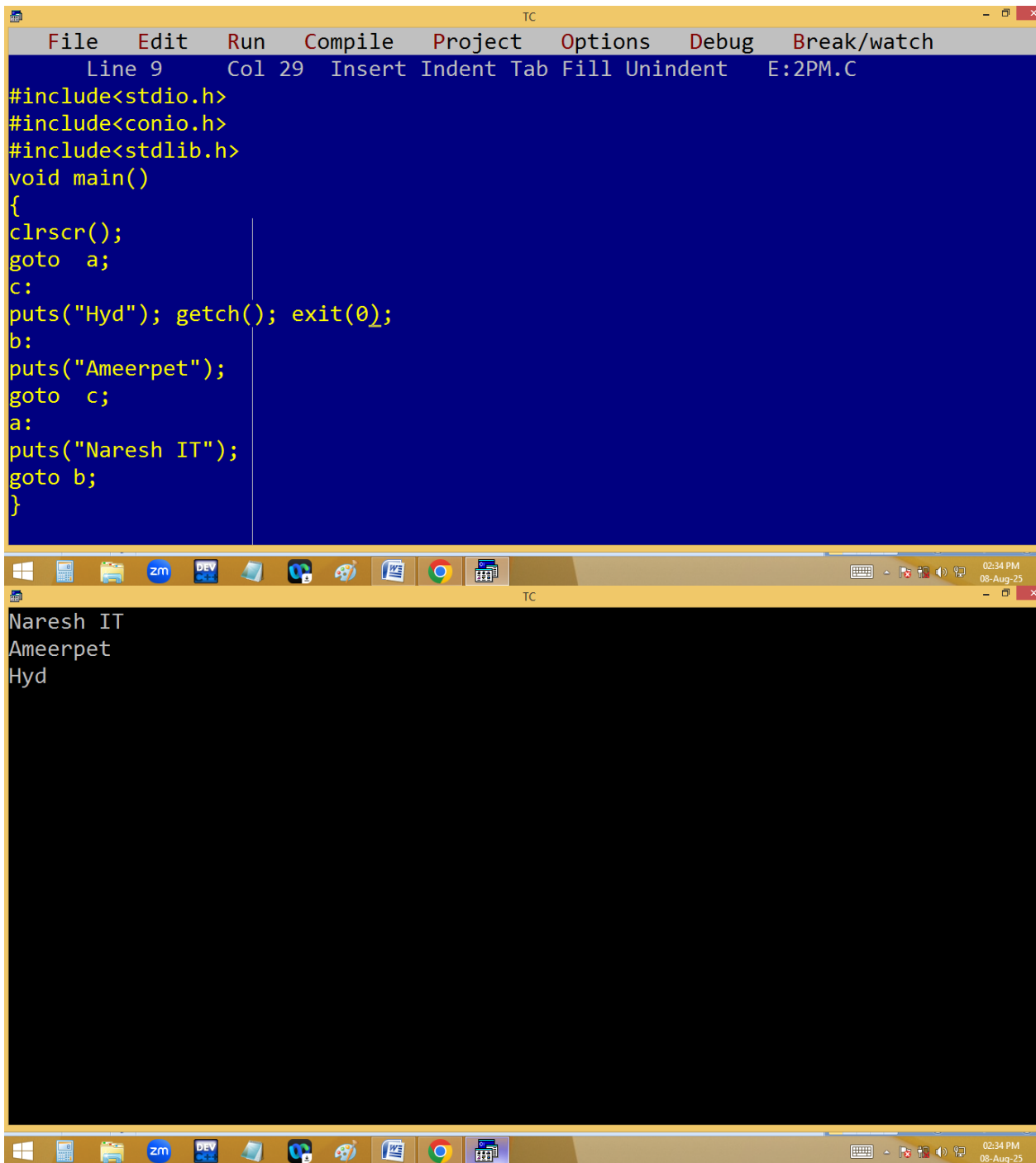
```
Naresh IT
Ameerpet
Hyd
_
```

The Windows taskbar at the bottom of the screen shows the time as 02:27 PM on 08-Aug-25. Various application icons are visible on the taskbar, including Zoho Mail (zm), DEV, and Google Chrome.

The image shows a screenshot of the Turbo C++ (TC) IDE. The top window displays the source code of a C program named E:2PM.C. The code uses goto statements to print three lines of text: "Naresh IT", "Ameerpet", and "Hyd". The bottom window shows the output of the program, which matches the text in the source code. The Windows taskbar at the bottom indicates the system time is 02:29 PM on 08-Aug-25.

```
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 30 Insert Indent Tab Fill Unindent E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
clrscr();
goto a;
c:
puts("Hyd"); getch(); return;
b:
puts("Ameerpet");
goto c;
a:
puts("Naresh IT");
goto b;
}
```

Naresh IT
Ameerpet
Hyd



The image shows a screenshot of the Turbo C++ (TC) IDE. The top window displays a C program with the following code:

```
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 29 Insert Indent Tab Fill Unindent E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
void main()
{
clrscr();
goto a;
c:
puts("Hyd"); getch(); exit(0);
b:
puts("Ameerpet");
goto c;
a:
puts("Naresh IT");
goto b;
}
```

The bottom window shows the output of the program, which is the text printed by the `puts` statements in reverse order of execution:

```
Naresh IT
Ameerpet
Hyd
```

The Windows taskbar at the bottom indicates the system time is 02:34 PM on 08-Aug-25. The taskbar includes icons for various applications like Zoho Mail (zm), DEV, and Google Chrome.

Conditional control statements:

They are working based on a condition.

C provides two types of conditional control statements.

1. If
2. Switch

if:

if is a keyword.

It is a decision making statement.

It is used to check the given condition / expression is true or false.

Note: In C & C++ other than 0 anything is 1 i.e. true.

We are having 4 types of if conditions.

- 1.Simple if
- 2.If..else
- 3.If..else if ladder
- 4.Nested if

Simple if: When the program is having only one condition then prefer simple if.

If condition true statements in if block { } are executed and later outside statements executed.

If condition false only the outside statements are executed.

Syntax:

