



$$2^*r-1$$

$$2^*1-1 = 1$$

$$2^*2-1 = 3$$

$$2^*3-1 = 5$$

$$2^*4-1 = 7$$

The image shows two screenshots of the Turbo C++ (TC) IDE. The top screenshot displays the source code for a C program that prints a diamond pattern. The code includes `<stdio.h>` and `<conio.h>`, and uses `clrscr()` to clear the screen. It prompts the user to enter the number of rows, which is stored in `n`. The program uses nested loops to print spaces and asterisks to form a diamond shape. The bottom screenshot shows the program's execution. The user has entered '10' for the number of rows, and the output is a diamond pattern of asterisks on a black background. The pattern consists of 10 rows, with the number of spaces increasing from 0 to 9 and the number of asterisks decreasing from 10 to 1, forming a symmetrical diamond shape.

```
File Edit Run Compile Project Options Debug Break/watch
Line 10 Col 32 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows "); scanf("%d",&n);
for(r=1;r<=n;r++)
{
for(s=1;s<=n-r;s++)printf(" ");
for(c=1;c<=2*r-1;c++) printf("*");
printf("\n");
}
getch();
}
```

Enter how many rows 10

```

      *
     ***
    *****
   *********
  ***********
 *****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

```
TC
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows  "); scanf("%d",&n);
while(!kbhit())
{
for(r=1;r<=n;r++)
{
for(s=1;s<=n-r;s++)cprintf(" ");
for(c=1;c<=2*r-1;c++) {textcolor(random(16));cprintf("*");}
printf("\n");
}
}
}
```

TC

```

* * * * *
 * * * * *
  * * * * *
   * * * * *
    * * * * *
     * * * * *
      * * * * *
       * * * * *
        * * * * *
         * * * * *
          * * * * *
           * * * * *
            * * * * *
             * * * * *
              * * * * *
               * * * * *
                * * * * *
                 * * * * *
                  * * * * *
                   * * * * *
                    * * * * *
                     * * * * *
                      * * * * *
                       * * * * *
                        * * * * *
                         * * * * *
                          * * * * *
                           * * * * *
                            * * * * *
                             * * * * *
                              * * * * *
                               * * * * *
                                * * * * *
                                 * * * * *
                                  * * * * *
                                   * * * * *
                                    * * * * *
                                     * * * * *
                                      * * * * *
                                       * * * * *
                                        * * * * *
                                         * * * * *
                                          * * * * *
                                           * * * * *
                                            * * * * *
                                             * * * * *
                                              * * * * *
                                               * * * * *
                                                * * * * *
                                                 * * * * *
                                                  * * * * *
                                                   * * * * *
                                                    * * * * *
                                                     * * * * *
                                                      * * * * *
                                                       * * * * *
                                                        * * * * *
                                                         * * * * *
                                                          * * * * *
                                                           * * * * *
                                                            * * * * *
                                                             * * * * *
                                                              * * * * *
                                                               * * * * *
                                                                * * * * *
                                                                 * * * * *
                                                                  * * * * *
                                                                   * * * * *
                                                                    * * * * *
                                                                     * * * * *
                                                                      * * * * *
                                                                       * * * * *
                                                                        * * * * *
                                                                         * * * * *
                                                                          * * * * *
                                                                           * * * * *
                                                                            * * * * *
                                                                           * * * * *
                                                                          * * * * *
                                                                         * * * * *
                                                                        * * * * *
                                                                       * * * * *
                                                                      * * * * *
                                                                     * * * * *
                                                                    * * * * *
                                                                   * * * * *
                                                                  * * * * *
                                                                 * * * * *
                                                                * * * * *
                                                               * * * * *
                                                              * * * * *
                                                             * * * * *
                                                            * * * * *
                                                           * * * * *
                                                          * * * * *
                                                         * * * * *
                                                        * * * * *
                                                       * * * * *
                                                      * * * * *
                                                     * * * * *
                                                    * * * * *
                                                   * * * * *
                                                  * * * * *
                                                 * * * * *
                                                * * * * *
                                               * * * * *
                                              * * * * *
                                             * * * * *
                                            * * * * *
                                           * * * * *
                                          * * * * *
                                         * * * * *
                                        * * * * *
                                       * * * * *
                                      * * * * *
                                     * * * * *
                                    * * * * *
                                   * * * * *
                                  * * * * *
                                 * * * * *
                                * * * * *
                               * * * * *
                              * * * * *
                             * * * * *
                            * * * * *
                           * * * * *
                          * * * * *
                         * * * * *
                        * * * * *
                       * * * * *
                      * * * * *
                     * * * * *
                    * * * * *
                   * * * * *
                  * * * * *
                 * * * * *
                * * * * *
               * * * * *
              * * * * *
             * * * * *
            * * * * *
           * * * * *
          * * * * *
         * * * * *
        * * * * *
       * * * * *
      * * * * *
     * * * * *
    * * * * *
   * * * * *
  * * * * *
 * * * * *
* * * * *
```

TC

```
TC
Line 17 Col 1 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows "); scanf("%d",&n);
while(!kbhit())
{
for(r=1;r<=n;r++)
{textcolor(random(16));
for(s=1;s<=n-r;s++)cprintf(" ");
for(c=1;c<=2*r-1;c++)cprintf("*");
printf("\n");
}
}
}
```

```
TC
*****
*****
*****
*****
*****
*****
*
***
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*
```



```
TC
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows  "); scanf("%d",&n);
while(!kbhit())
{textcolor(LIGHTRED);
for(r=1;r<=n;r++)
{
for(s=1;s<=n-r;s++)cprintf(" ");
for(c=1;c<=2*r-1;c++)cprintf("*");
printf("\n");
}
}
}
```

TC

```
*****
*****
 *
 ***
*****
*****
 *
 ***
*****
*****
*****
 *
 ***
*****
*****
*****
*
```

TC

```
TC
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows "); scanf("%d",&n);
while(!kbhit())
{textcolor(LIGHTRED);
for(r=n;r>=1;r--)
{
for(s=1;s<=n-r;s++)cprintf(" ");
for(c=1;c<=2*r-1;c++)cprintf("*");
printf("\n");
}
}
}
```

TC

```

***
 *
*****
*****
  *****
   ***
    *
  *****
 *****
   *****
    ***
     *
  *****
 *****
   *****
    ***
     *
```

TC

```
TC
#include<stdio.h> #include<conio.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows "); scanf("%d",&n);
for(r=1;r<=n;r++)
{
for(s=1;s<=n-r;s++)cprintf(" ");
for(c=1;c<=2*r-1;c++)cprintf("*");
printf("\n");
}
for(r=n-1;r>=1;r--)
{
for(s=1;s<=n-r;s++)cprintf(" ");
for(c=1;c<=2*r-1;c++)cprintf("*");
printf("\n");
}
getch();}

Enter how many rows 4
*
***
*****
*****
*****
***
*

```



```
TC
#include<stdio.h> #include<conio.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows "); scanf("%d",&n);
for(r=1;r<=n;r++)
{
for(s=1;s<=n-r;s++)printf(" ");
for(c=1;c<=2*r-1;c++)if(c==1||c==r)printf("* ");else printf(" ");
printf("\n");
}
for(r=n-1;r>=1;r--)
{
for(s=1;s<=n-r;s++)printf(" ");
for(c=1;c<=2*r-1;c++)if(c==1||c==r)printf("* ");else printf(" ");
printf("\n");
}
getch();}

TC
Enter how many rows 7

      *
     * *
    *   *
   *     *
  *       *
 *         *
*          *
*         *
 *       *
  *     *
   *   *
    * *
     *

TC
02:46 PM
29-Aug-23
```

```
TC
#include<stdio.h> #include<conio.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows "); scanf("%d",&n);
for(r=n;r>=1;r--)
{
for(s=1;s<=n-r;s++)printf(" ");
for(c=1;c<=2*r-1;c++)printf("*");
printf("\n");
}
for(r=2;r<=n;r++)
{
for(s=1;s<=n-r;s++)printf(" ");
for(c=1;c<=2*r-1;c++)printf("*");
printf("\n");
}
getch();}

TC
Enter how many rows 5
*****
*****
****
***
**
*
**
***
****
*****
*****
*****
```

```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 30 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n,r,c,s; clrscr();
printf("Enter how many rows "); scanf("%d",&n);
for(r=1;r<=n;r++)
{
for(s=1;s<=n-r;s++)printf(" ");
for(c=1;c<=r;c++)printf("%3d",c);
for(c=r-1;c>=1;c--)printf("%3d",c);
printf("\n");
}
getch();
}
```

Enter how many rows 12

```

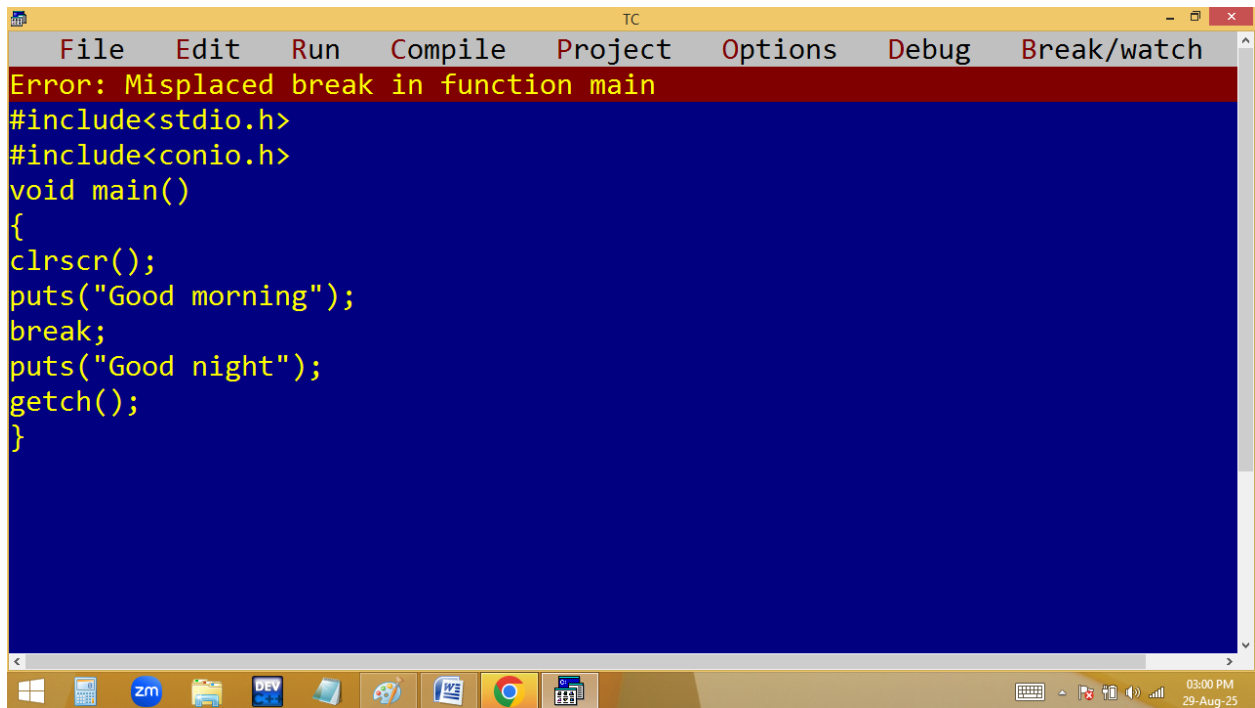
          1
        1 2 1
      1 2 3 2 1
    1 2 3 4 3 2 1
  1 2 3 4 5 4 3 2 1
1 2 3 4 5 6 5 4 3 2 1
  1 2 3 4 5 6 7 6 5 4 3 2 1
    1 2 3 4 5 6 7 8 7 6 5 4 3 2 1
      1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1
        1 2 3 4 5 6 7 8 9 10 9 8 7 6 5 4 3 2 1
          1 2 3 4 5 6 7 8 9 10 11 10 9 8 7 6 5 4 3 2 1
```

```
for(r=1;r<=4;r++)
{
for(s=1;s<=n-r;s++)p(" ");
for(c=1;c<=r;c++)p(c);
for(c=r-1;c>=1;c--)p(c);
```



## break:

It is a keyword used within switch/loop to stop the execution. When break is occurred, the program execution jumped to the first statement after the switch or loop.



The screenshot shows the Turbo C++ (TC) IDE interface. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. A red error message banner at the top reads "Error: Misplaced break in function main". The main editing area has a dark blue background with yellow text showing the following code:

```
#include<stdio.h>
#include<conio.h>
void main()
{
clrscr();
puts("Good morning");
break;
puts("Good night");
getch();
}
```

The Windows taskbar is visible at the bottom, showing icons for various applications and the system clock indicating 03:00 PM on 29-Aug-25.

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`, defines a `main` function, declares an integer `i`, clears the screen with `clrscr()`, and uses a `for` loop to print numbers 1 through 10. After the loop, it prints "Thank you" and waits for a key press with `getch()`. The bottom screenshot shows the same IDE with the program's output in a black window. The output consists of the numbers 1 through 10, each on a new line, followed by the text "Thank you". The Windows taskbar at the bottom of both screenshots shows the time as 03:01 PM and 03:02 PM on 29-Aug-23.

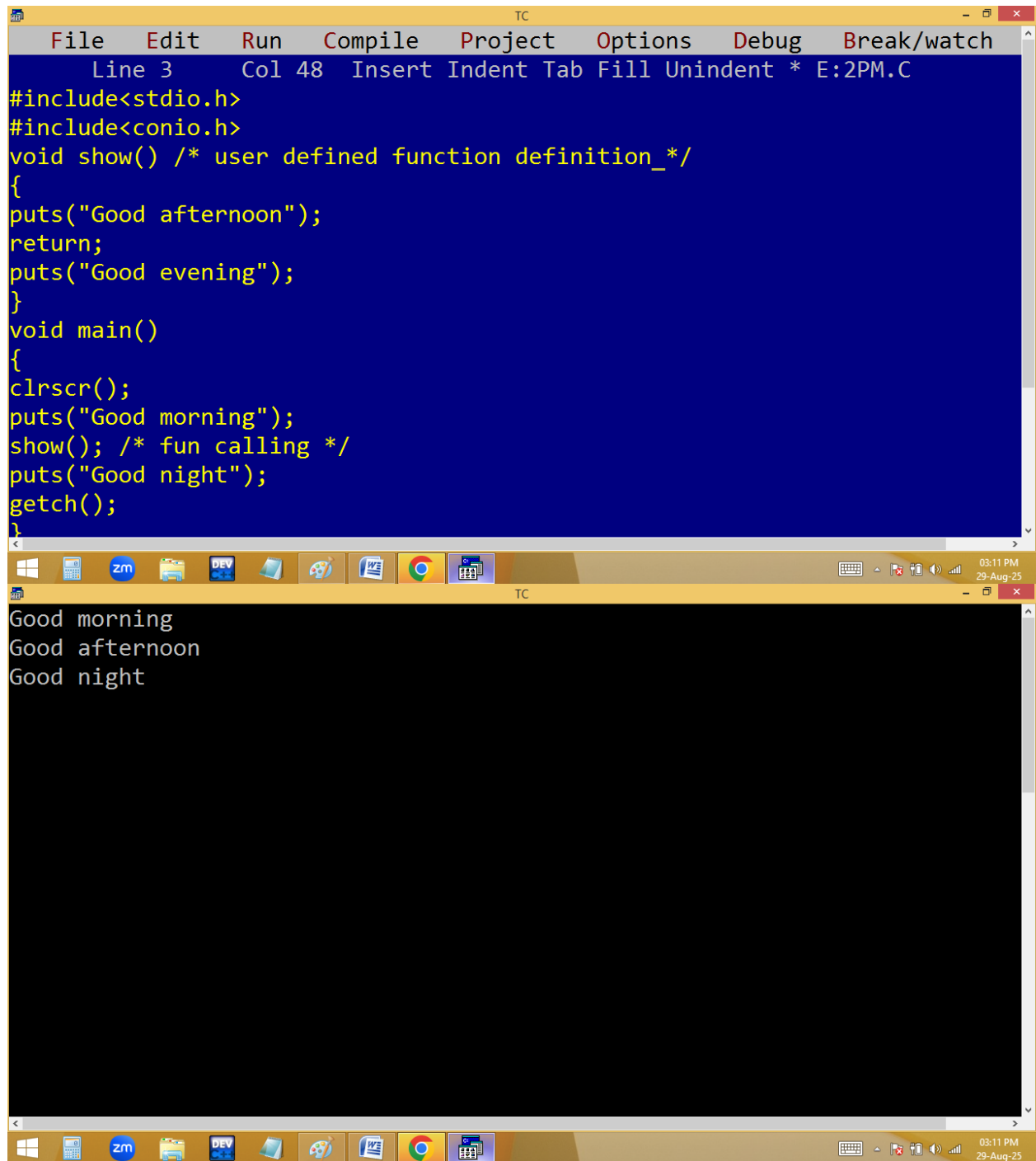
```
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 18 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i;
clrscr();
for(i=1;i<=10;i++)
{
printf("%d\n",i);
}
printf("Thank you");
getch();
}
```

```
1
2
3
4
5
6
7
8
9
10
Thank you
```

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 implements a `main` function that prints numbers 1 through 4, followed by "Thank you" and a newline character. The bottom screenshot shows the same IDE with the program's output in a black window, displaying the numbers 1, 2, 3, 4, and the text "Thank you" followed by a cursor.

```
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 15 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i;
clrscr();
for(i=1;i<=10;i++)
{
if(i==5)break;_
printf("%d\n",i);
}
printf("Thank you");
getch();
}
```

1  
2  
3  
4  
Thank you\_



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 3 Col 48 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void show() /* user defined function definition_*/
{
puts("Good afternoon");
return;
puts("Good evening");
}
void main()
{
clrscr();
puts("Good morning");
show(); /* fun calling */
puts("Good night");
getch();
}
```

The bottom window shows the output of the program:

```
Good morning
Good afternoon
Good night
```

The Windows taskbar at the bottom shows the time as 03:11 PM on 29-Aug-25. The taskbar includes icons for Windows, a calculator, Zoom (zm), File Explorer, DEV, a folder icon, a paint icon, a Word icon, Google Chrome, and the Turbo C++ application.

The screenshot shows the Turbo C++ (TC) IDE. The top window displays the following C code:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
void show() /* user defined function definition */
{
puts("Good afternoon");
exit(0);
puts("Good evening");
}
void main()
{
clrscr();
puts("Good morning");
show(); /* fun calling */
puts("Good night");
getch();
}
```

The bottom window shows the output of the program:

```
Good morning
Good afternoon
```

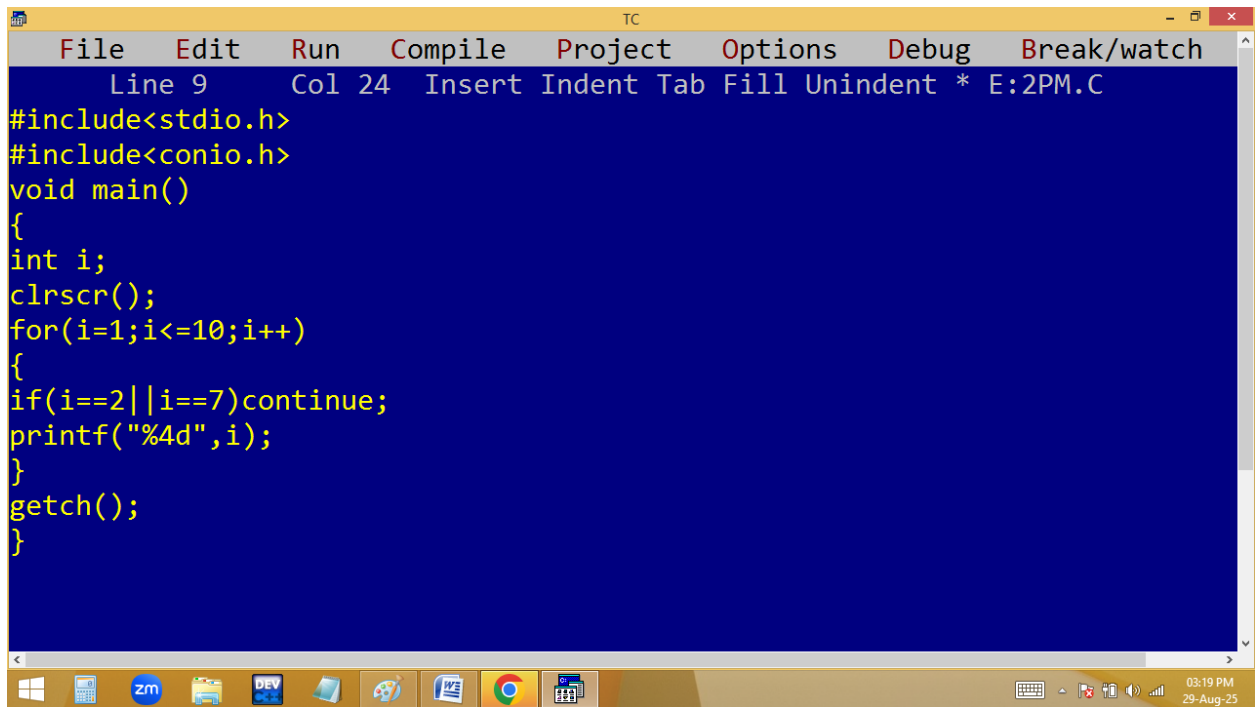
The taskbar at the bottom shows various icons including Windows, Zm, DEV, and the system clock indicating 03:14 PM on 29-Aug-23.

<b>break</b>	<b>exit()</b>	<b>return</b>
<b>Keyword</b>	<b>Function</b>	<b>Keyword</b>
<b>Header file not required</b>	<b>stdlib.h / process.h required</b>	<b>Header file not required</b>
<b>Terminates switch / loop</b>	<b>Terminates total program</b>	<b>Closes that function only</b>
<b>Should be used within</b>	<b>Used any where</b>	<b>Used any where</b>

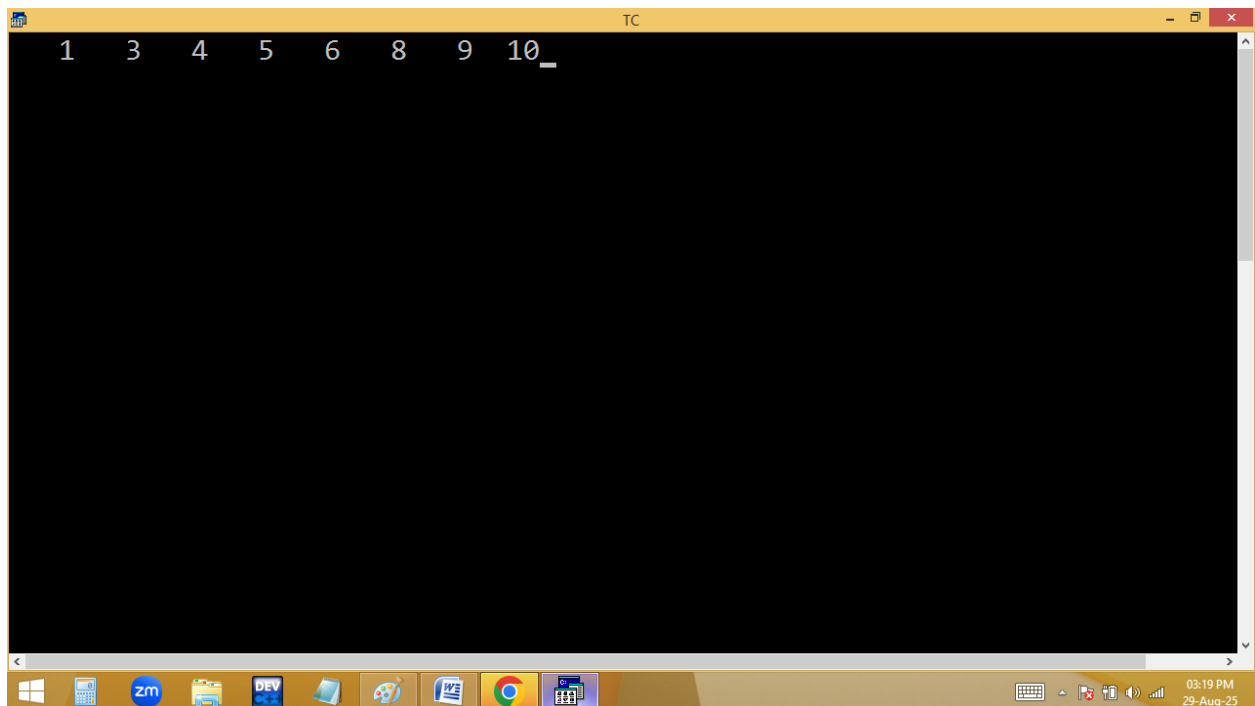


switch/loop		
-------------	--	--

**continue**: It is a keyword used within loops only. When continue occurred, the program execution jumped to loop ending and later to the beginning. Due to this the remaining statements in that loop are skipped.

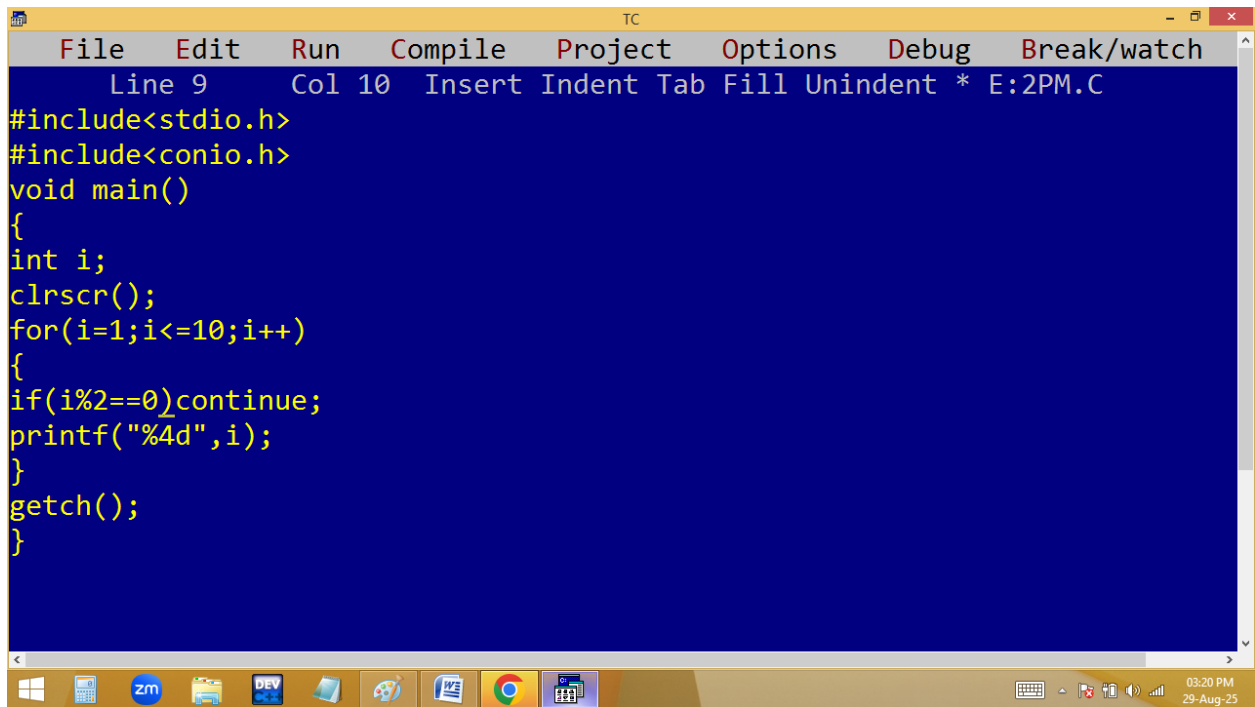


```
TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 24 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i;
clrscr();
for(i=1;i<=10;i++)
{
if(i==2||i==7)continue;
printf("%4d",i);
}
getch();
}
```



```
TC
1 3 4 5 6 8 9 10_
```

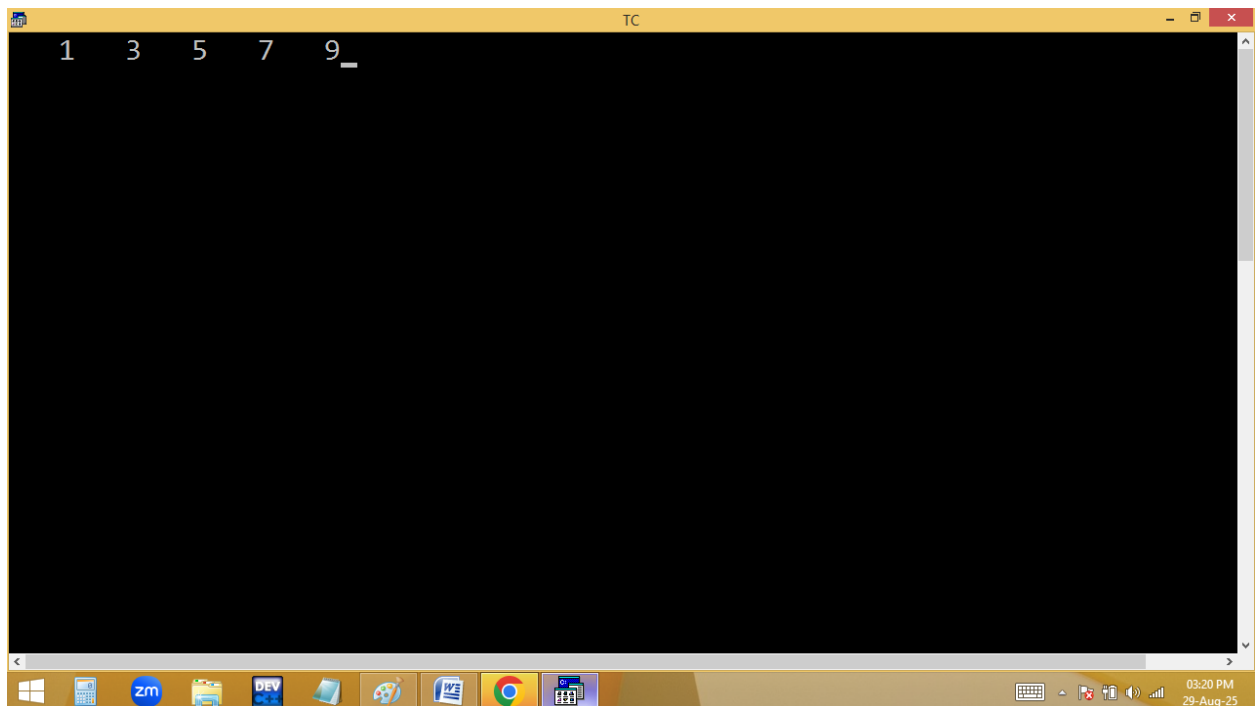
**Printing 1..10 odd no's using continue:**



The screenshot shows the Turbo C++ (TC) IDE with a yellow title bar and menu bar. The menu bar includes File, Edit, Run, Compile, Project, Options, Debug, and Break/watch. Below the menu bar, a status bar shows 'Line 9 Col 10 Insert Indent Tab Fill Unindent \* E:2PM.C'. The main editing area has a blue background and contains the following C code:

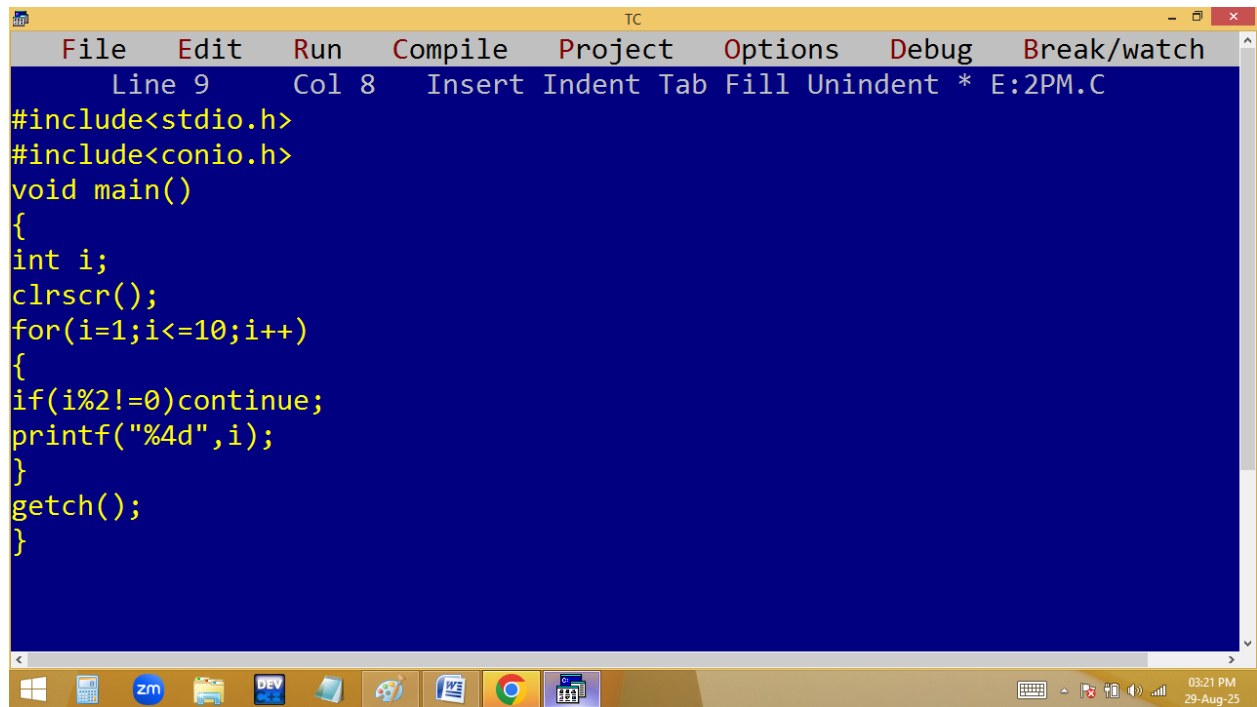
```
#include<stdio.h>
#include<conio.h>
void main()
{
    int i;
    clrscr();
    for(i=1;i<=10;i++)
    {
        if(i%2==0)continue;
        printf("%4d",i);
    }
    getch();
}
```

The Windows taskbar at the bottom shows various icons including Windows Start, Task View, ZOOM, File Explorer, DEV C++, Paint, Word, Chrome, and the TC application. The system tray on the right shows the time as 03:20 PM and the date as 29-Aug-25.



The screenshot shows the Turbo C++ (TC) IDE with a yellow title bar. The main editing area has a black background and displays the output of the program: '1 3 5 7 9\_'. The Windows taskbar at the bottom is identical to the first screenshot, showing the same icons and system tray information (03:20 PM, 29-Aug-25).

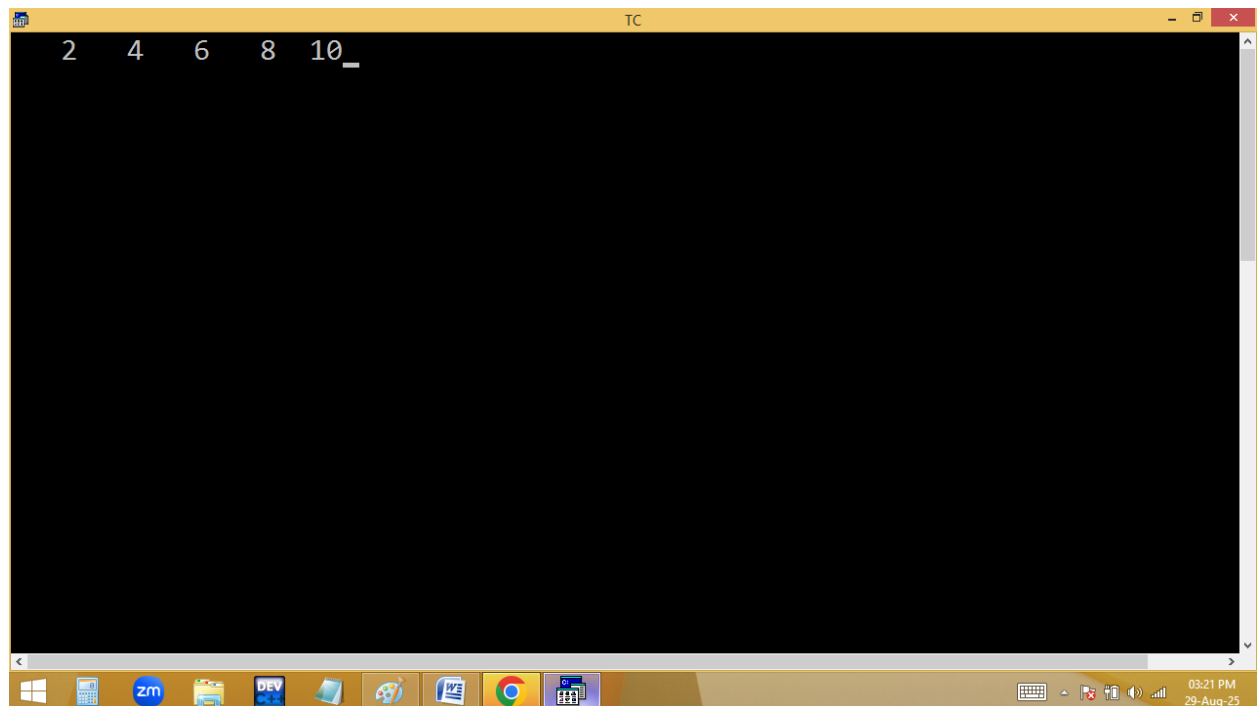
**Even no's:**



The screenshot shows the Turbo C++ (TC) IDE with a menu bar (File, Edit, Run, Compile, Project, Options, Debug, Break/watch) and a status bar (Line 9, Col 8, Insert, Indent, Tab, Fill, Unindent, \* E:2PM.C). The main editing area has a blue background and contains the following C code:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int i;
    clrscr();
    for(i=1;i<=10;i++)
    {
        if(i%2!=0)continue;
        printf("%4d",i);
    }
    getch();
}
```

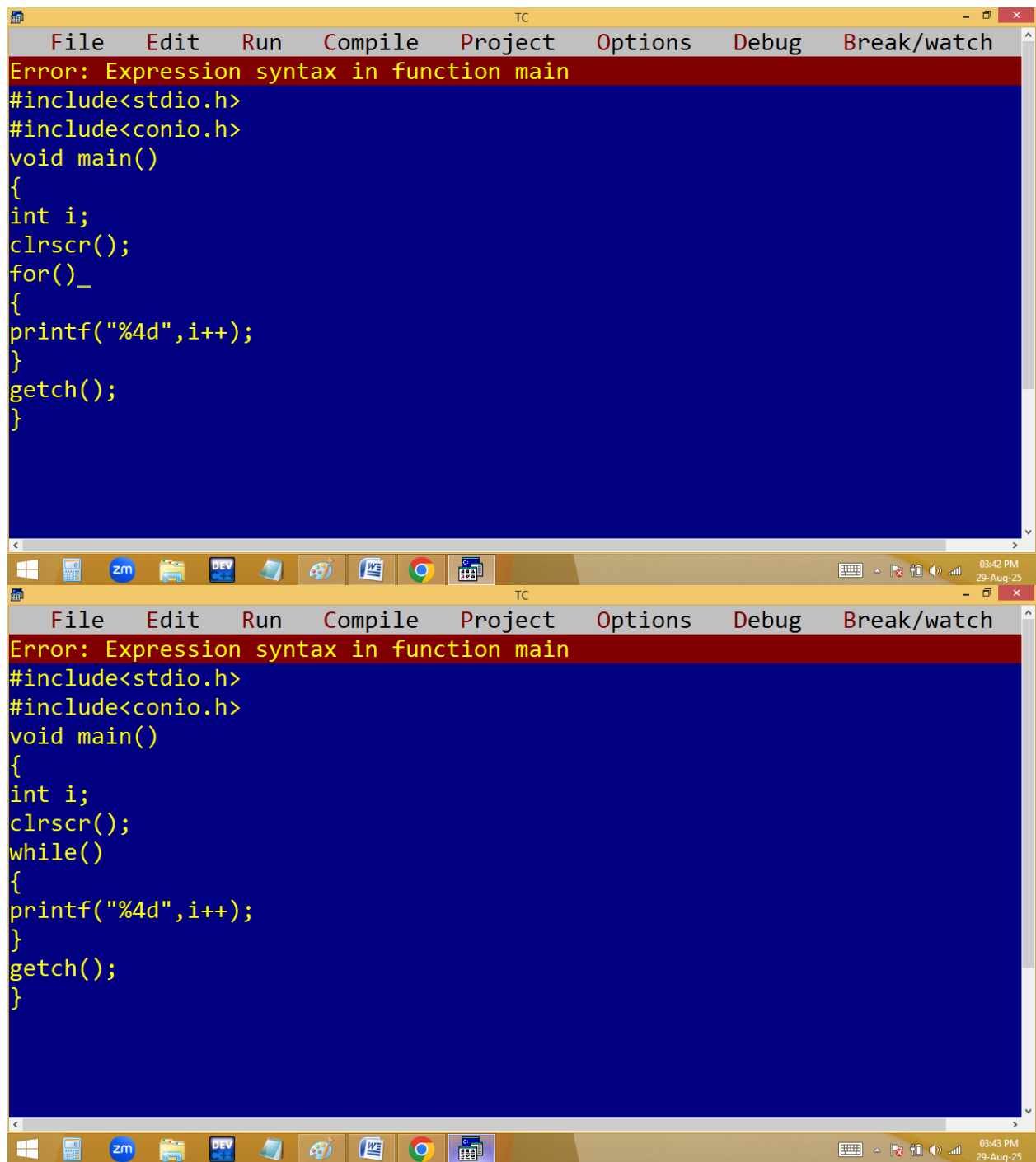
The Windows taskbar at the bottom shows icons for Windows, calculator, ZOOM, DEV, and other applications. The system clock indicates 03:21 PM on 29-Aug-25.



The screenshot shows the Turbo C++ (TC) IDE with the same menu bar and status bar as the first image. The main editing area has a black background and displays the output of the program:

```
2 4 6 8 10_
```

The Windows taskbar at the bottom is identical to the first image, showing the same icons and system clock (03:21 PM on 29-Aug-25).



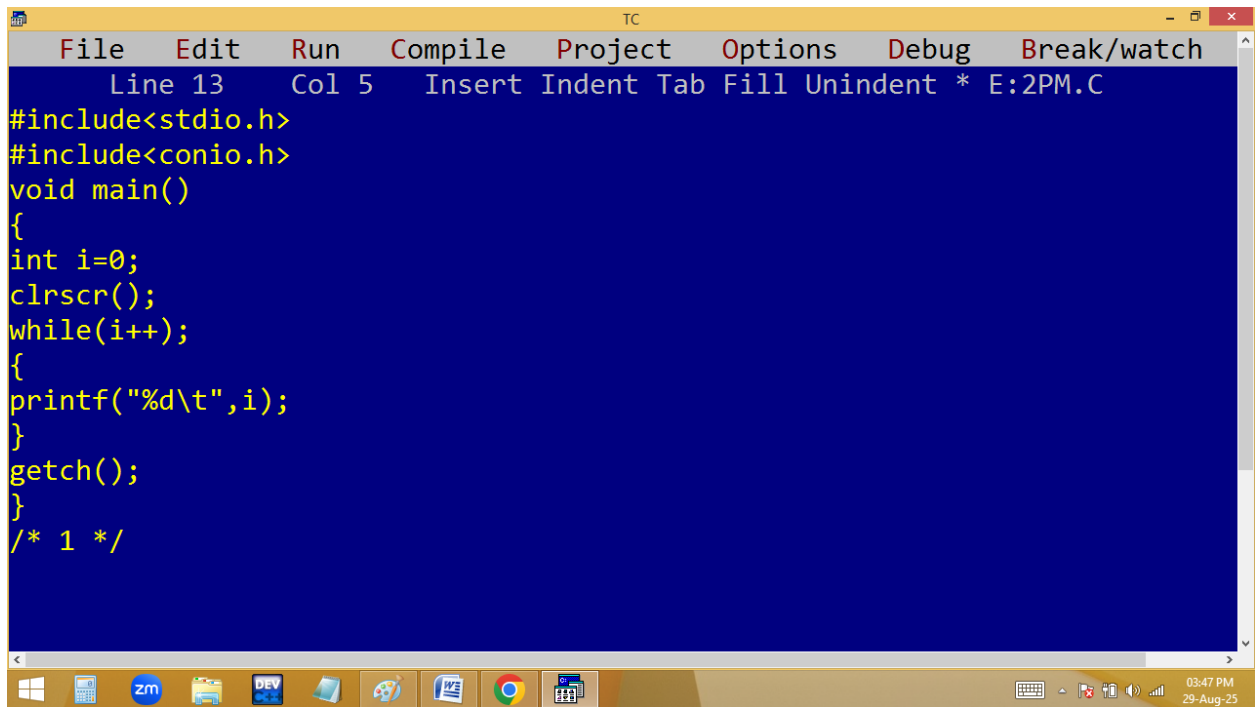
The image shows two instances of the Turbo C++ (TC) IDE. The top window displays a C program that prints a sequence of numbers from 0 to 3, followed by a range of negative numbers, and then repeats the sequence from 0 to 3 infinitely. The bottom window displays a C program that prints the number 0 and then clears the screen.

**Top Window Code:**

```
File Edit Run Compile Project Options Debug Break/watch
Line 7 Col 13 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=0;
clrscr();
while(1)
{
printf("%d\t",i++);
}
getch();
}
/* 0 1 2 3 ....32767 -32768 -32767...-1 0 1 2 3 infinite times */
```

**Bottom Window Code:**

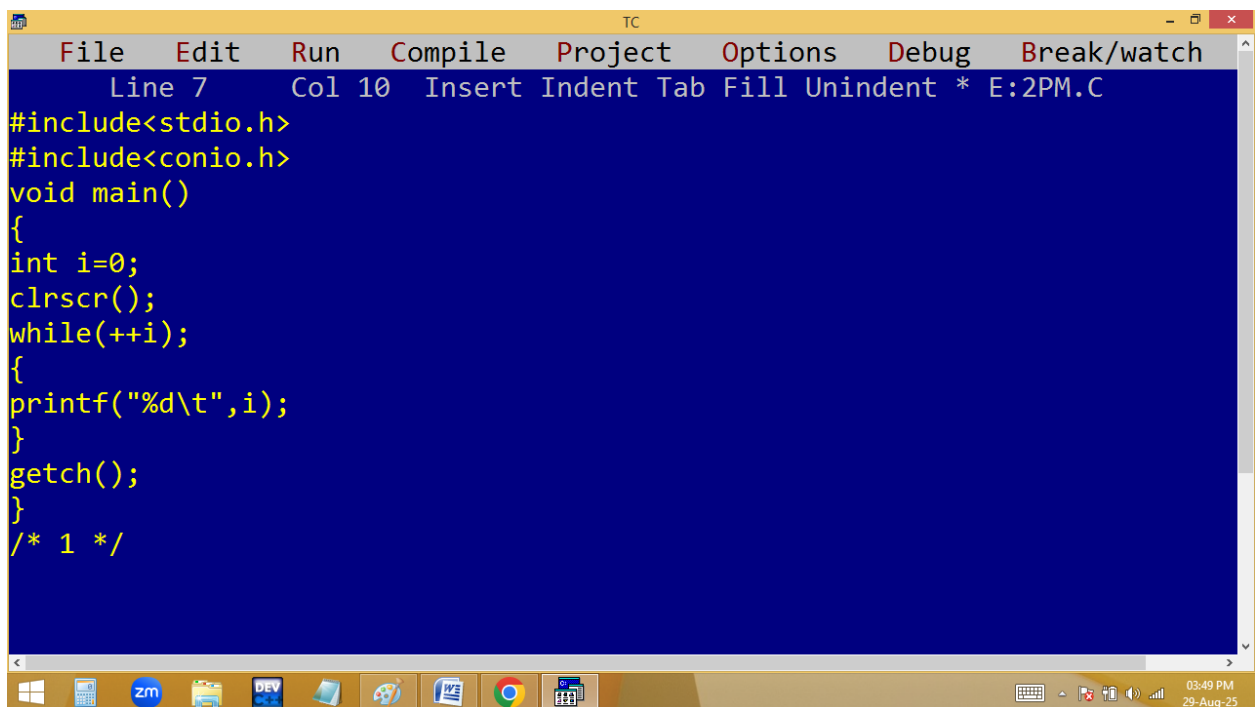
```
File Edit Run Compile Project Options Debug Break/watch
Line 13 Col 16 Insert Indent Tab Fill Unindent * E:2PM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=0;
clrscr();
while(i++)
{
printf("%d\t",i);
}
getch();
}
/* Blank screen_*/
```



The screenshot shows the Turbo C++ IDE with a menu bar (File, Edit, Run, Compile, Project, Options, Debug, Break/watch) and a status bar (Line 13, Col 5, Insert, Indent, Tab, Fill, Unindent, \* E:2PM.C). The code in the editor is as follows:

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i=0;
clrscr();
while(i++);
{
printf("%d\t",i);
}
getch();
}
/* 1 */
```

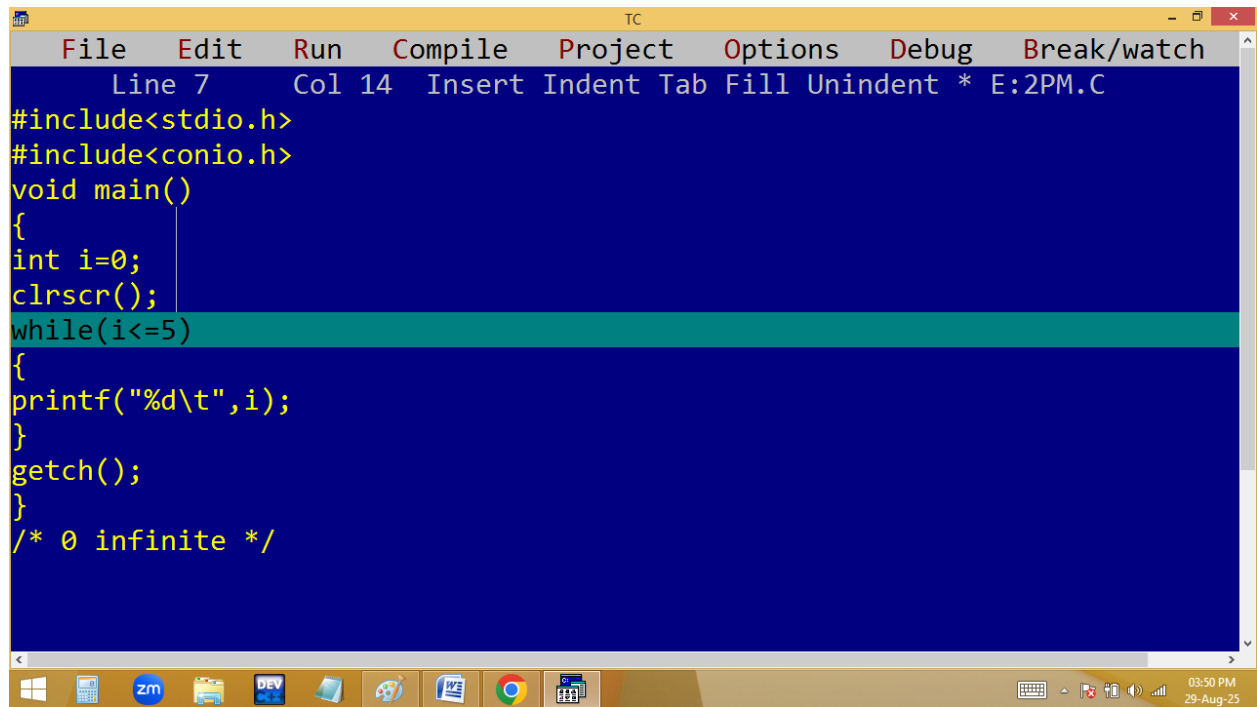
The Windows taskbar at the bottom shows the time as 03:47 PM on 29-Aug-25.



The screenshot shows the Turbo C++ IDE with a menu bar (File, Edit, Run, Compile, Project, Options, Debug, Break/watch) and a status bar (Line 7, Col 10, Insert, Indent, Tab, Fill, Unindent, \* E:2PM.C). The code in the editor is as follows:

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i=0;
clrscr();
while(++i);
{
printf("%d\t",i);
}
getch();
}
/* 1 */
```

The Windows taskbar at the bottom shows the time as 03:49 PM on 29-Aug-25.



The image shows 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. The status bar at the top indicates "Line 7", "Col 14", and "Insert Indent Tab Fill Unindent \* E:2PM.C". The code editor has a dark blue background with yellow text. The code is as follows:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int i=0;
    clrscr();
    while(i<=5)
    {
        printf("%d\t",i);
    }
    getch();
}
/* 0 infinite */
```

The line "while(i<=5)" is highlighted with a teal background. The Windows taskbar is visible at the bottom, showing icons for Windows, a calculator, a ZOOM application, a folder, a DEV C++ icon, a paint application, a Word document, a Google Chrome browser, and a calendar. The system clock in the bottom right corner shows "03:50 PM" and "29-Aug-25".