Loops / Iterations / Repetitive statements

Loops are used to repeat a block/group of statements continuously until the given condition becomes false.

Loops reduce program size and improves performance.

In loops beginning and ending points are same.

C-Language supports basically 2 types of loops.

1. **Entry/pre controlled loops.**
2. **Exit/post controlled loops.**

In entry control loops, condition is tested first and it is true then only statements block is executed.

Under entry control loops we are having

i. **While loop**
ii. **For loop**

In exit control loop, the statements are executed first and later condition is tested.

Under exit control loop we are having

i. **do while.**

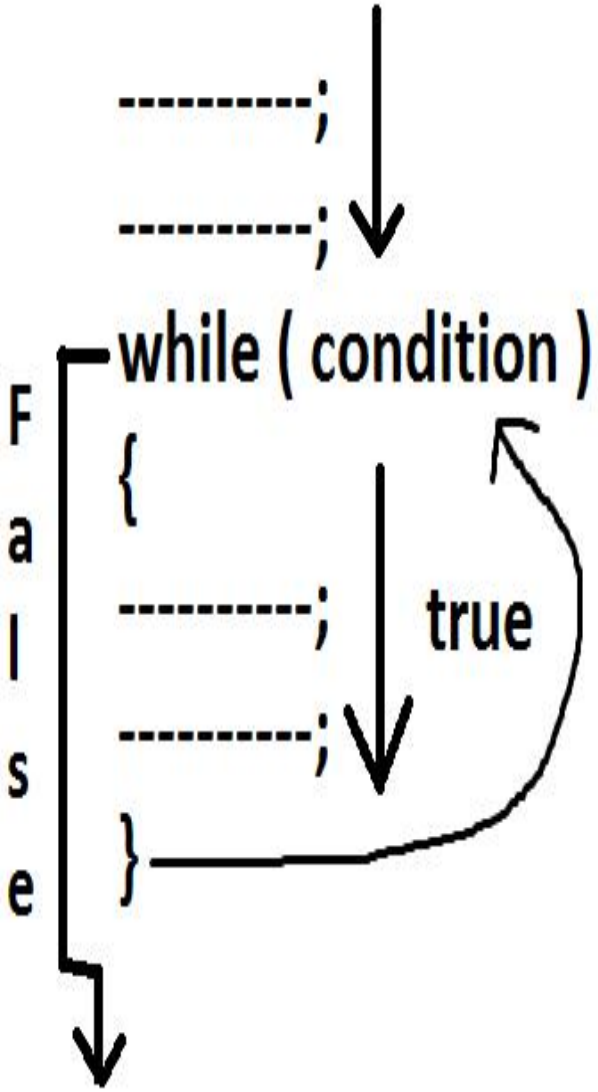
While loop:

- **while is a keyword.**
- **In while loop condition is tested first and it is true then only while block statements are executed. After executing while block statements, the program execution automatically shifted/jumped to while condition at the beginning. If it is true then once again the while block**

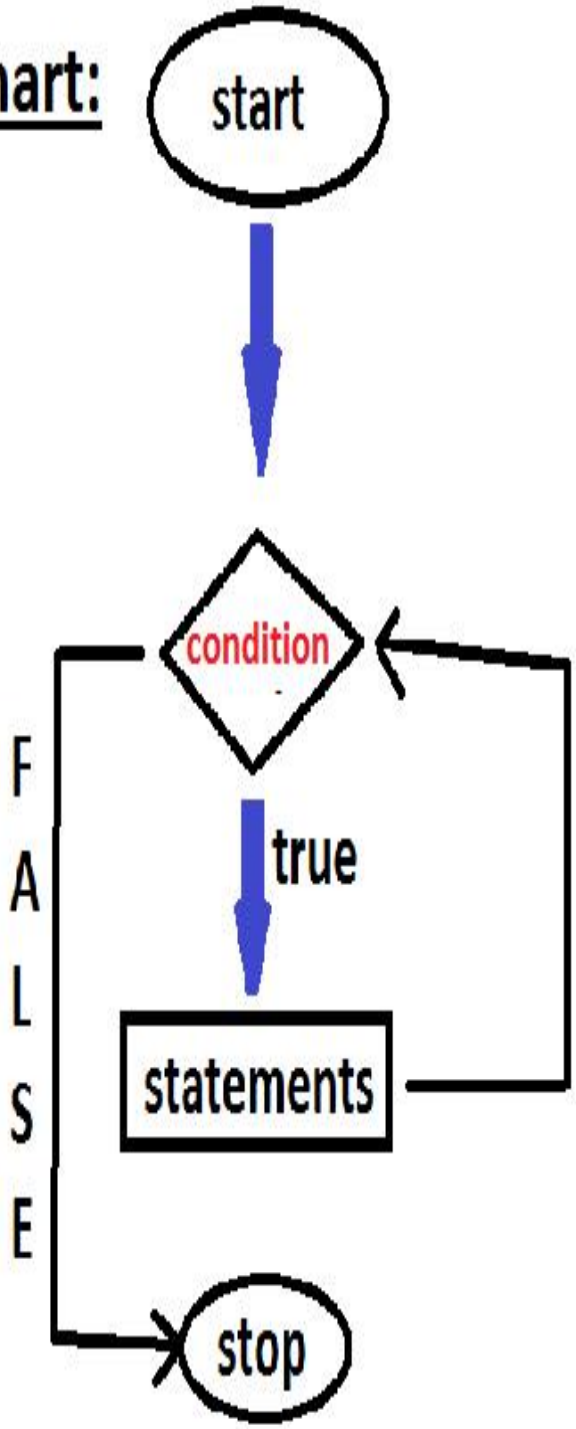
statements are repeated. Like this the process is continued until while condition becomes false.

- **While is entry control loop.**

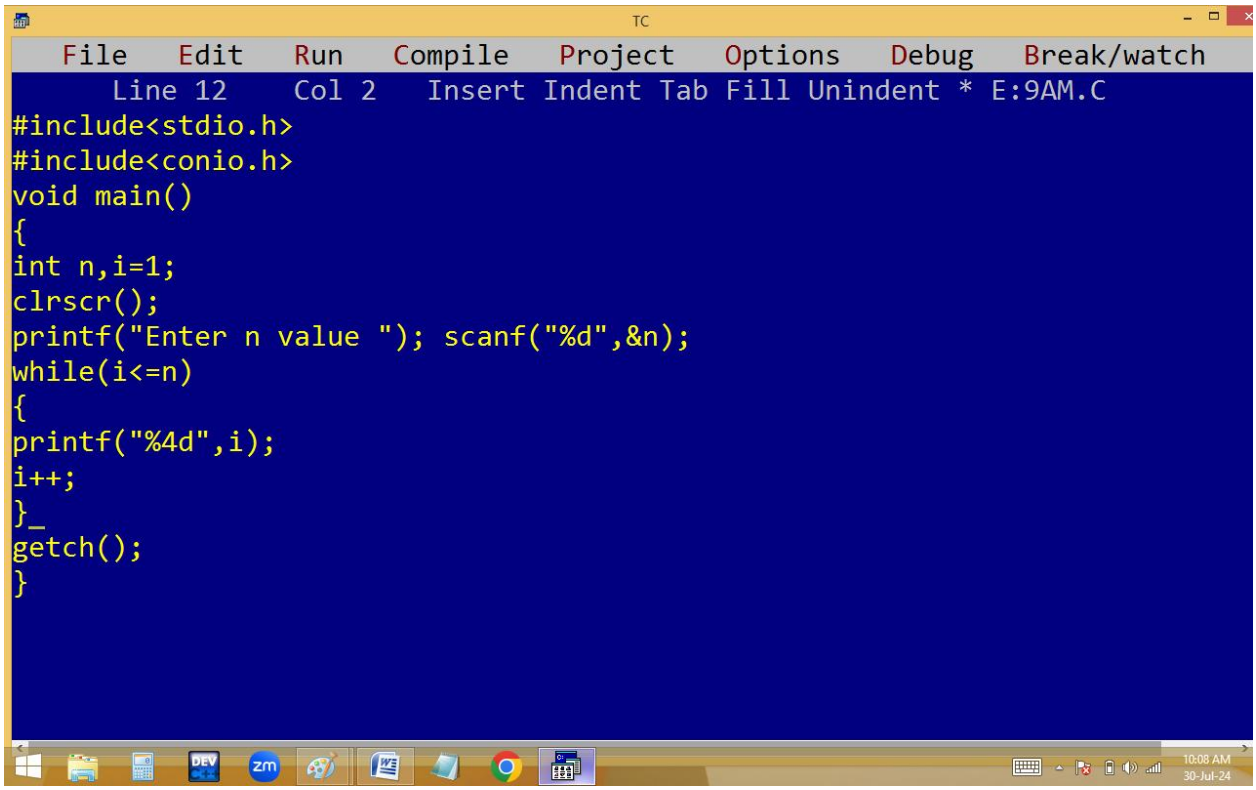
Syntax:



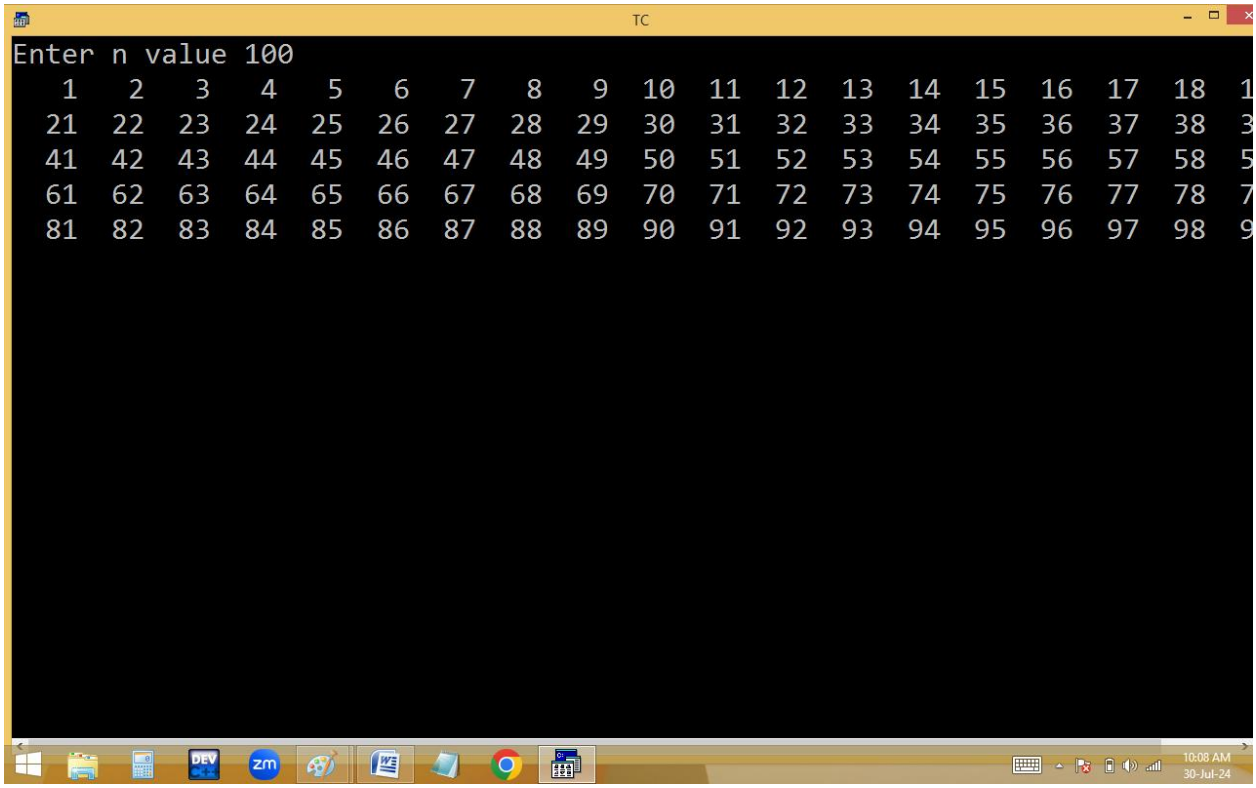
Flow chart:

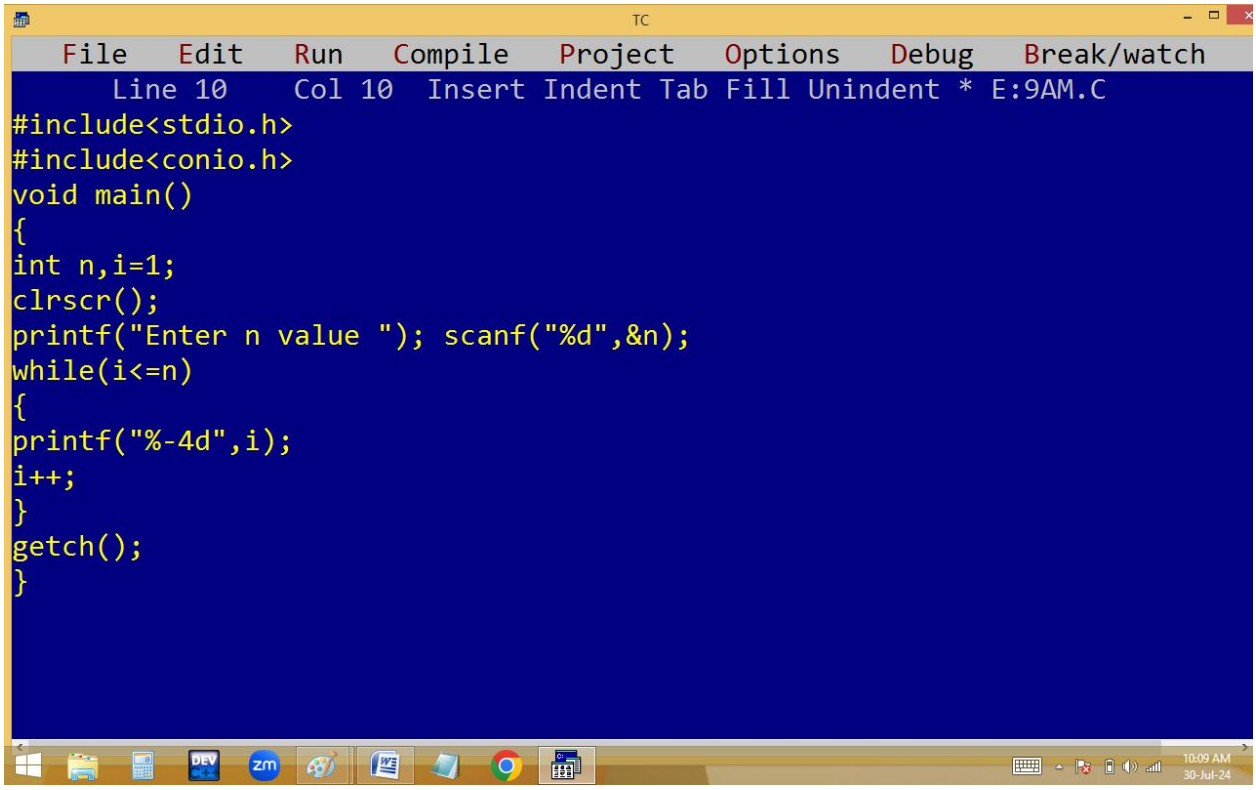


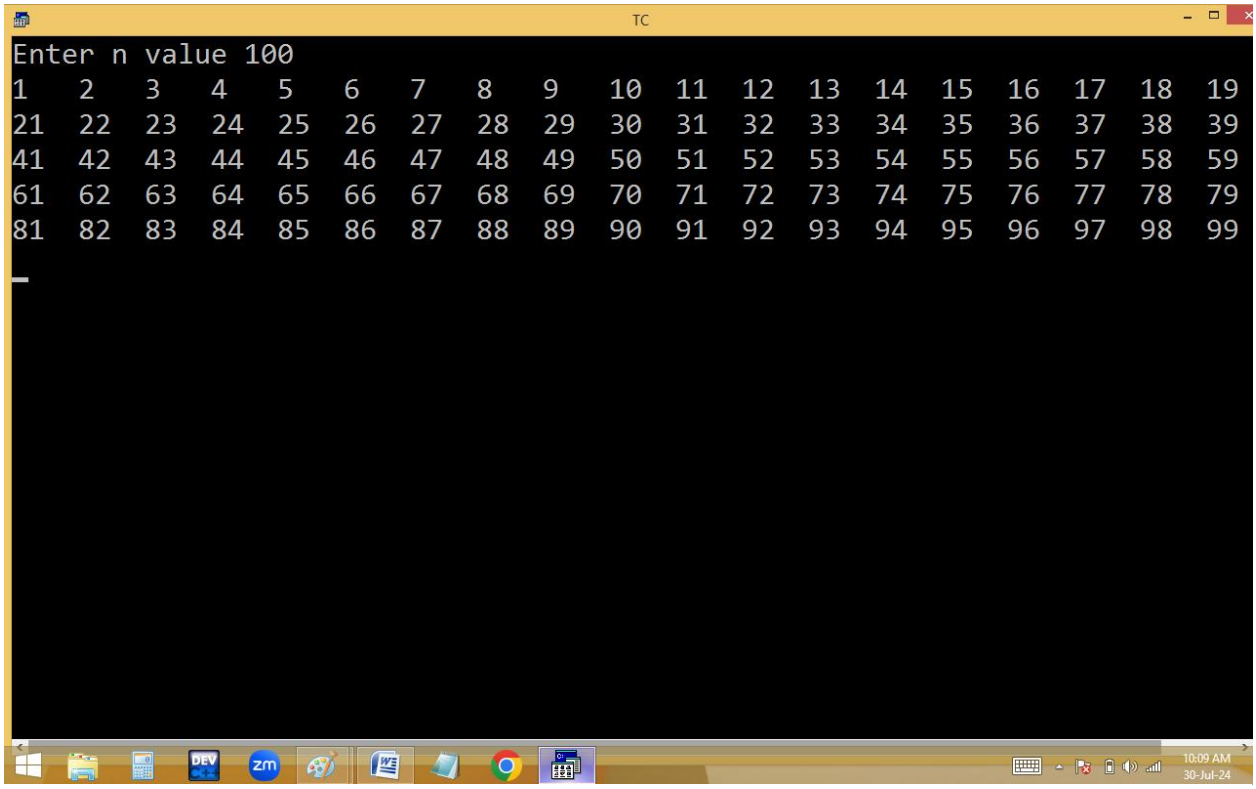
Eg. Printing 1..n numbers.

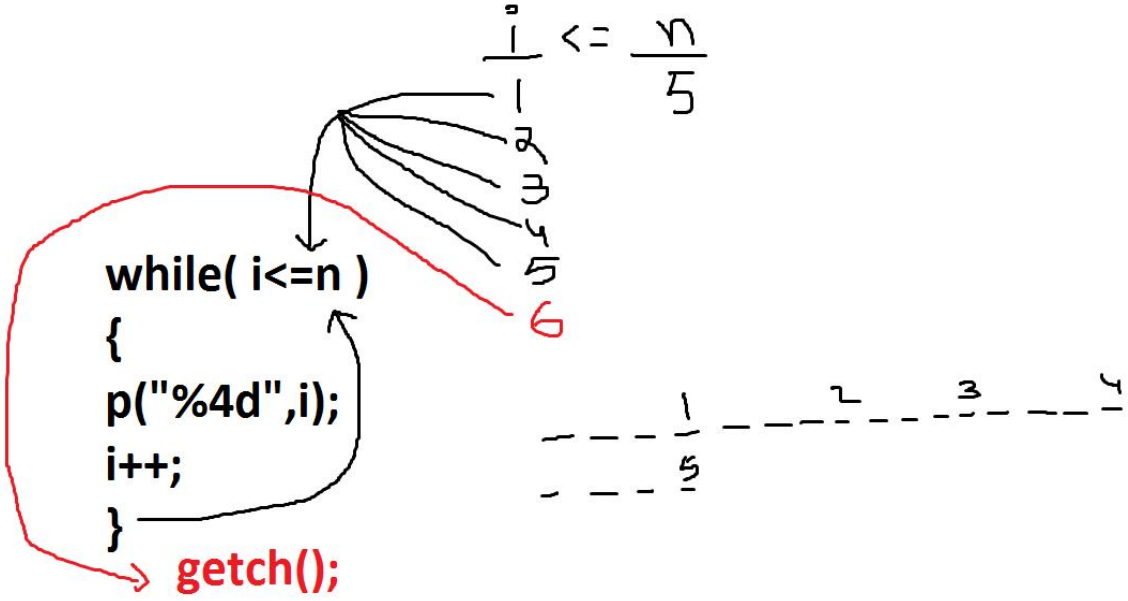


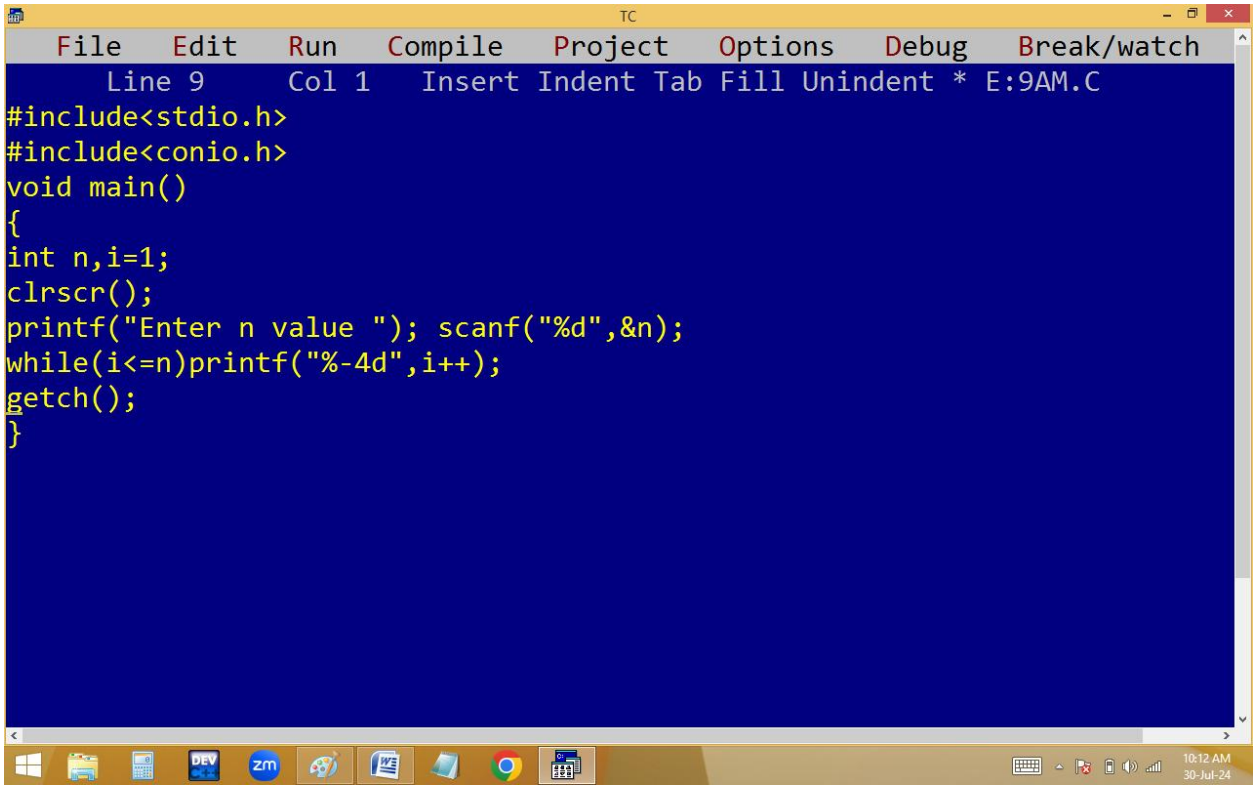
TC
Enter n value 5
1 2 3 4 5
Speakers/Headphones: 100%

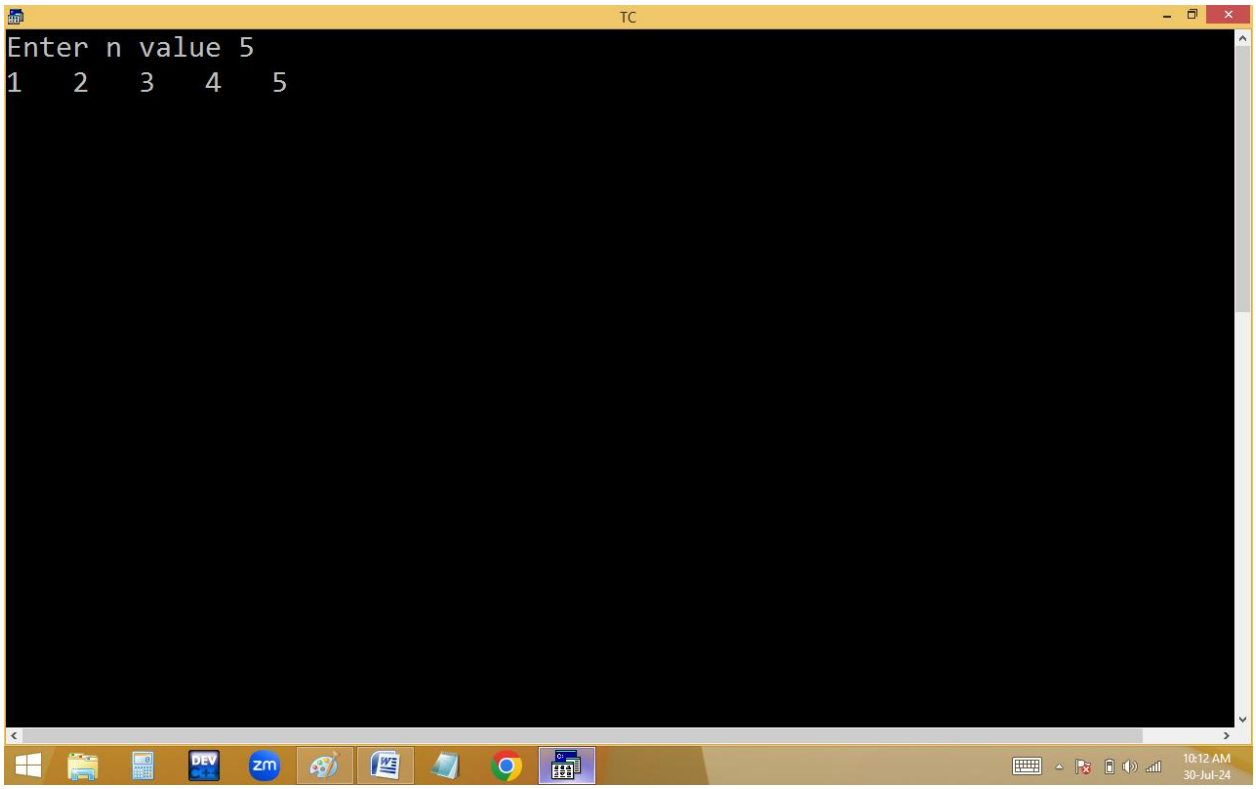

TC
Enter n value 100
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 1
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 3
41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 5
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 7
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 9
10:08 AM
30-Jul-24


TC
File Edit Run Compile Project Options Debug Break/watch
Line 10 Col 10 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n,i=1;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<=n)
{
printf("%-4d",i);
i++;
}
getch();
}

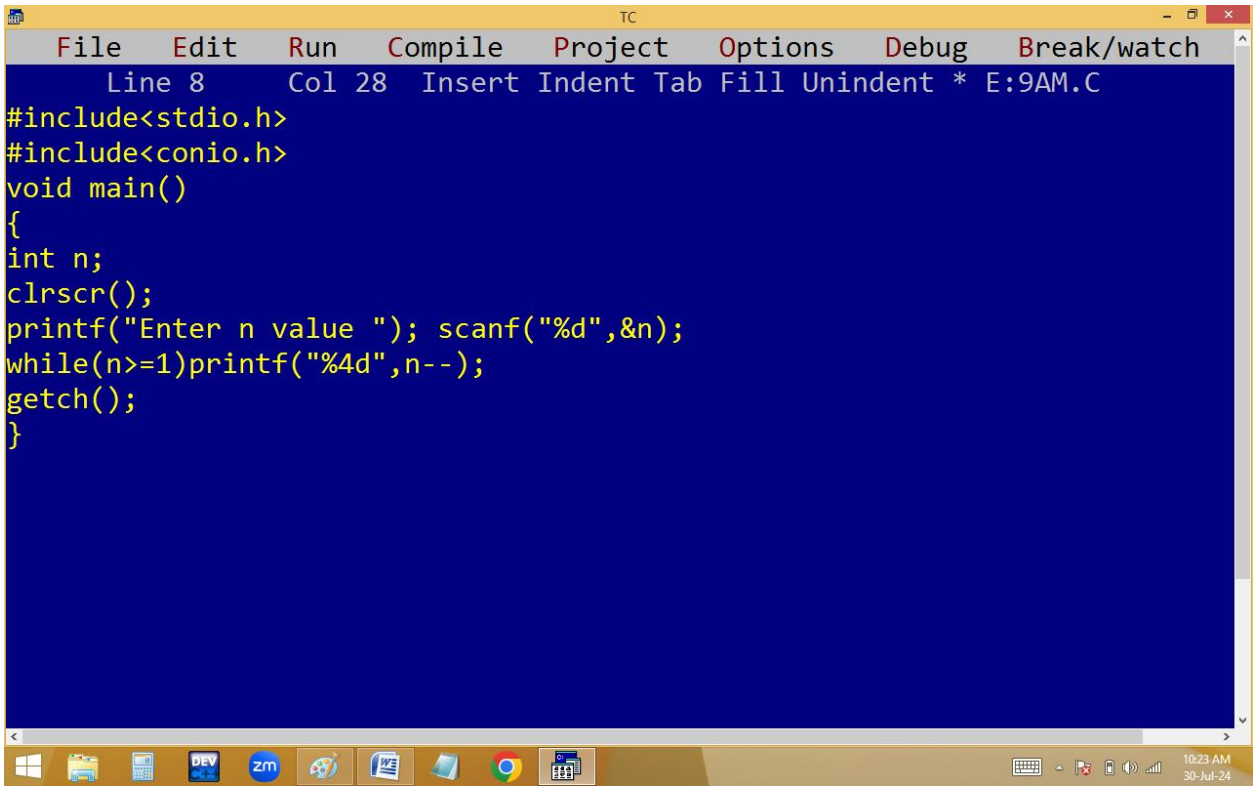

TC
Enter n value 100
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39
41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99

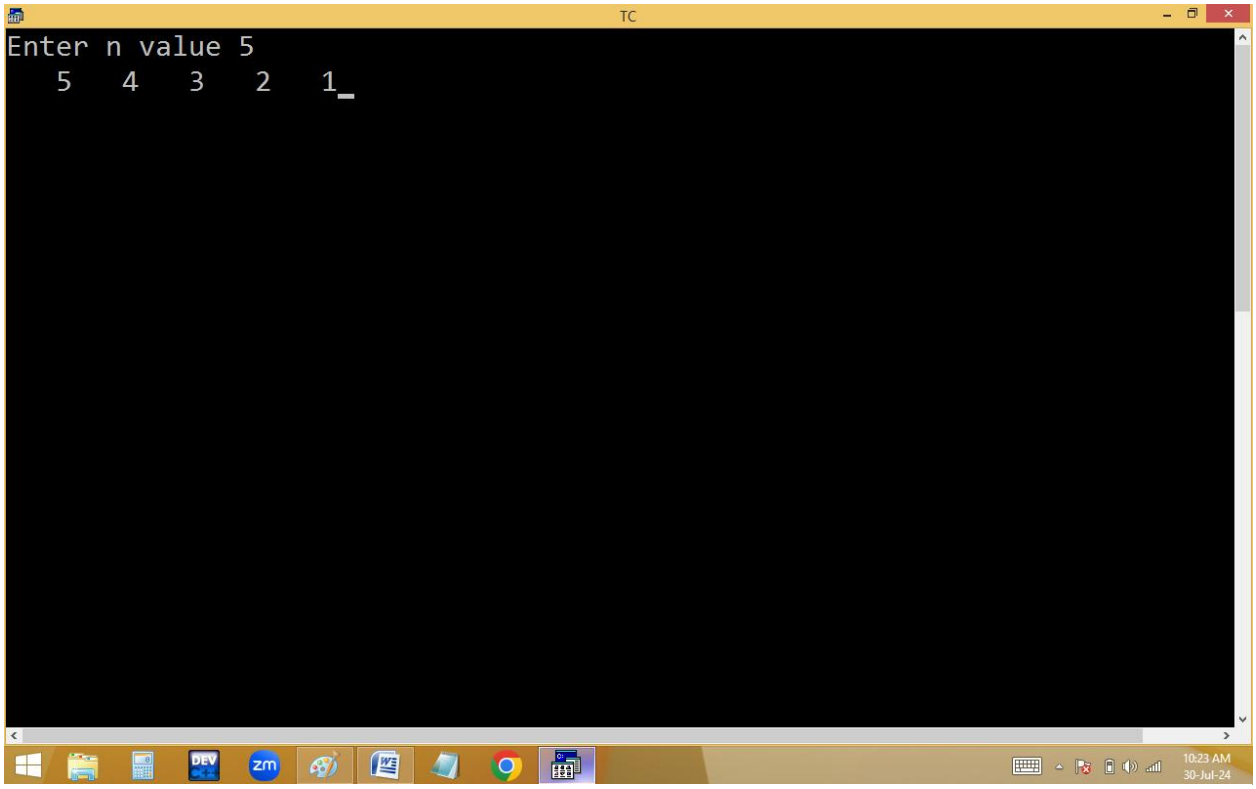

i <= n / 5
1
2
3
4
5
6
while(i<=n)
{
p("%4d",i);
i++;
}
getch();

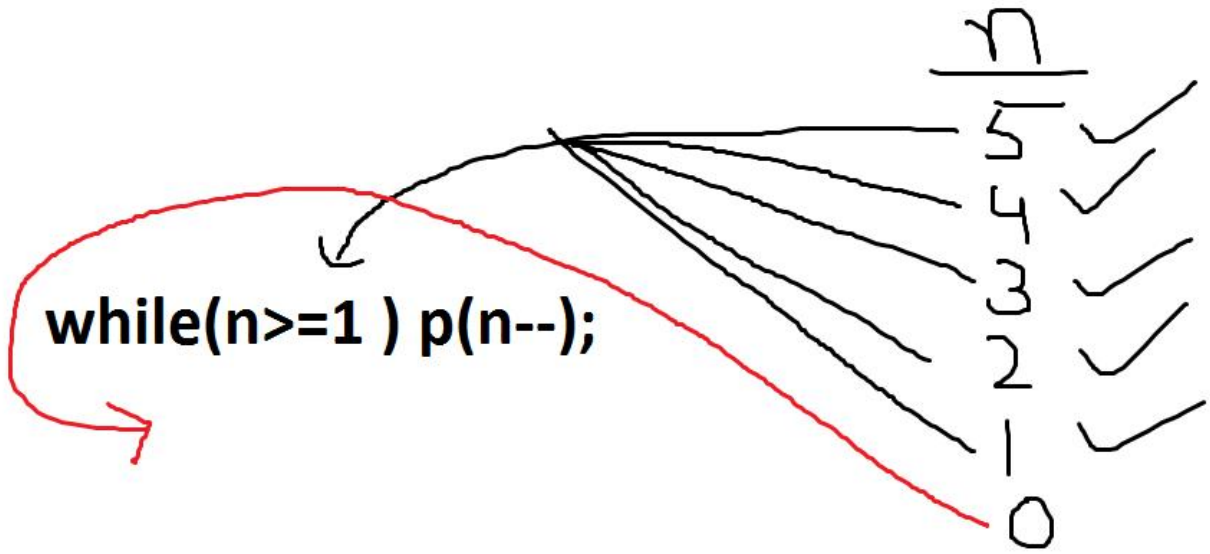

TC
File Edit Run Compile Project Options Debug Break/watch
Line 9 Col 1 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int n,i=1;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<=n)printf("%-4d",i++);
getch();
}


TC
Enter n value 5
1 2 3 4 5

Printing 1..n no’s in reverse order.

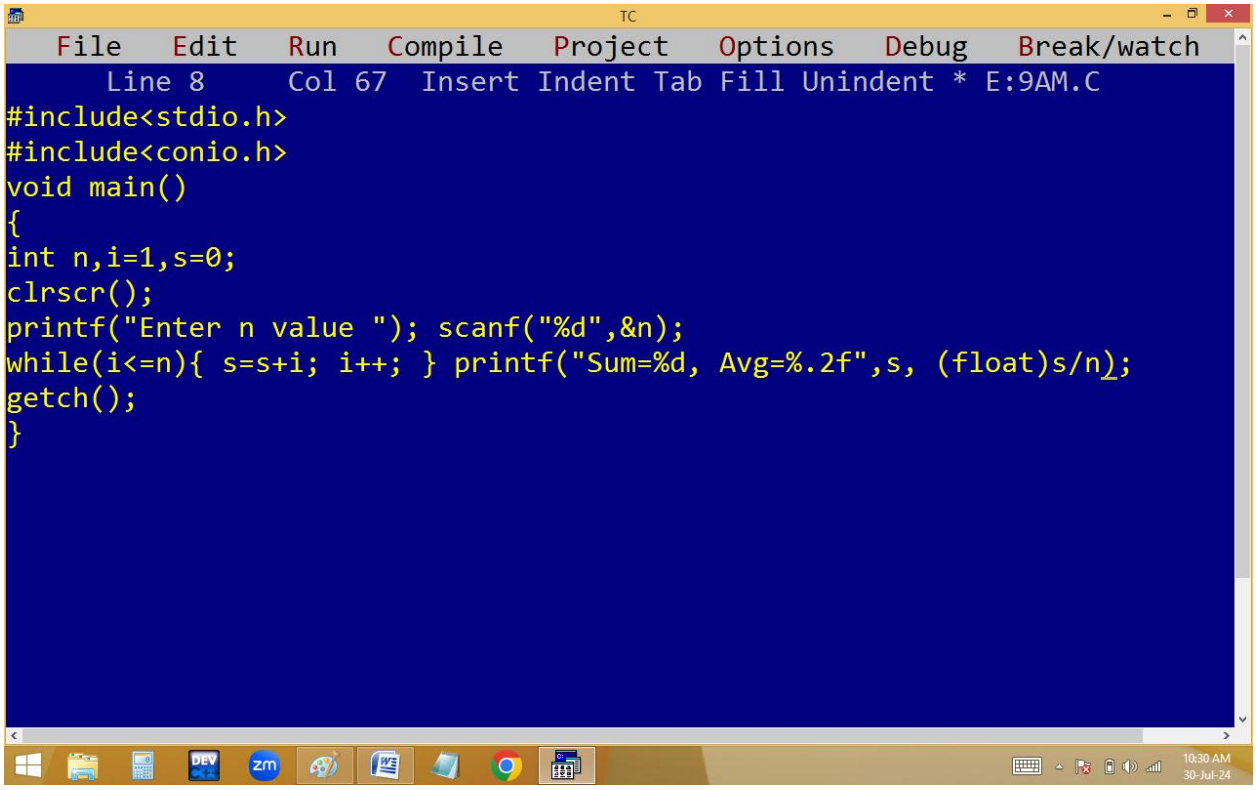


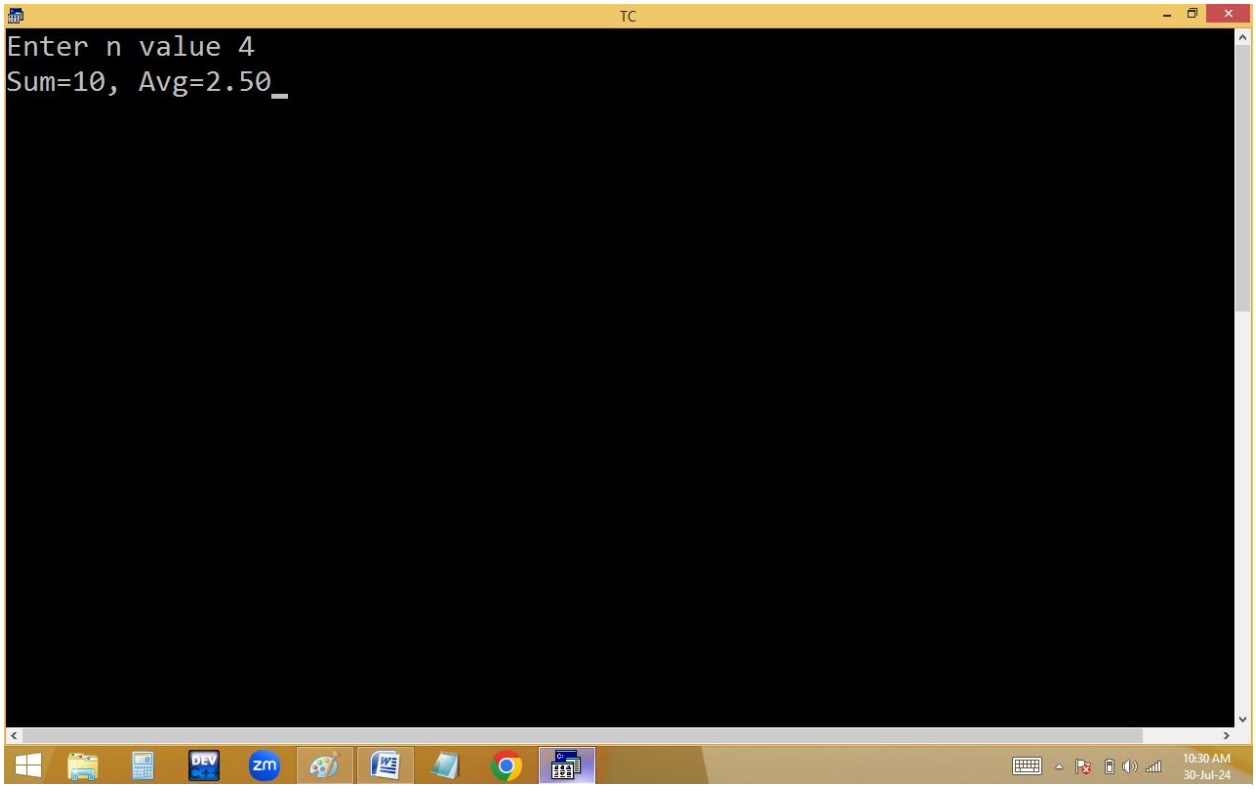

TC
Enter n value 5
5 4 3 2 1

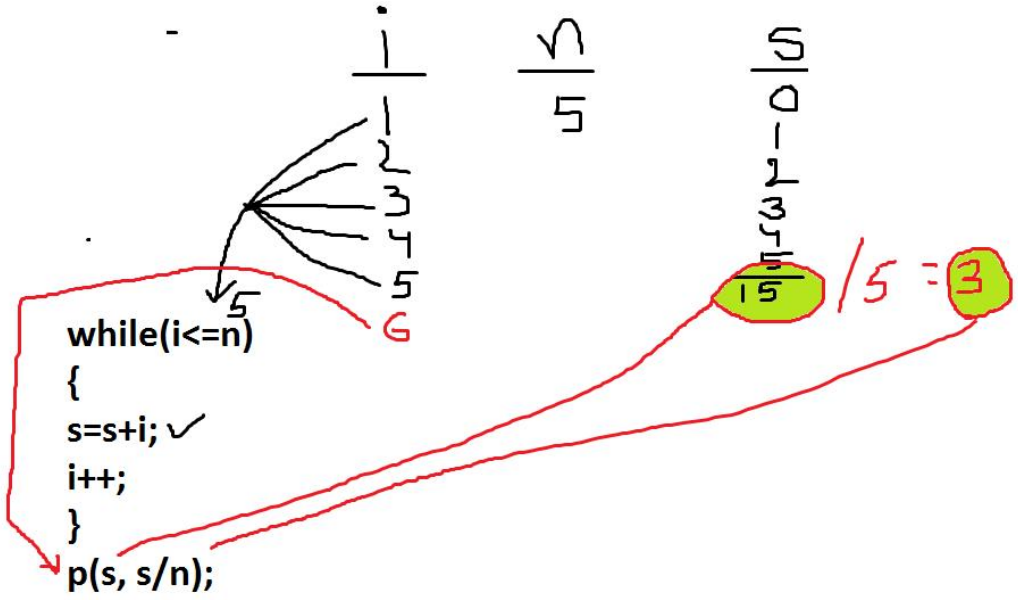

while(n>=1) p(n--);
n
5
4
3
2
1
0

Finding 1..n number sum and avg[mean]

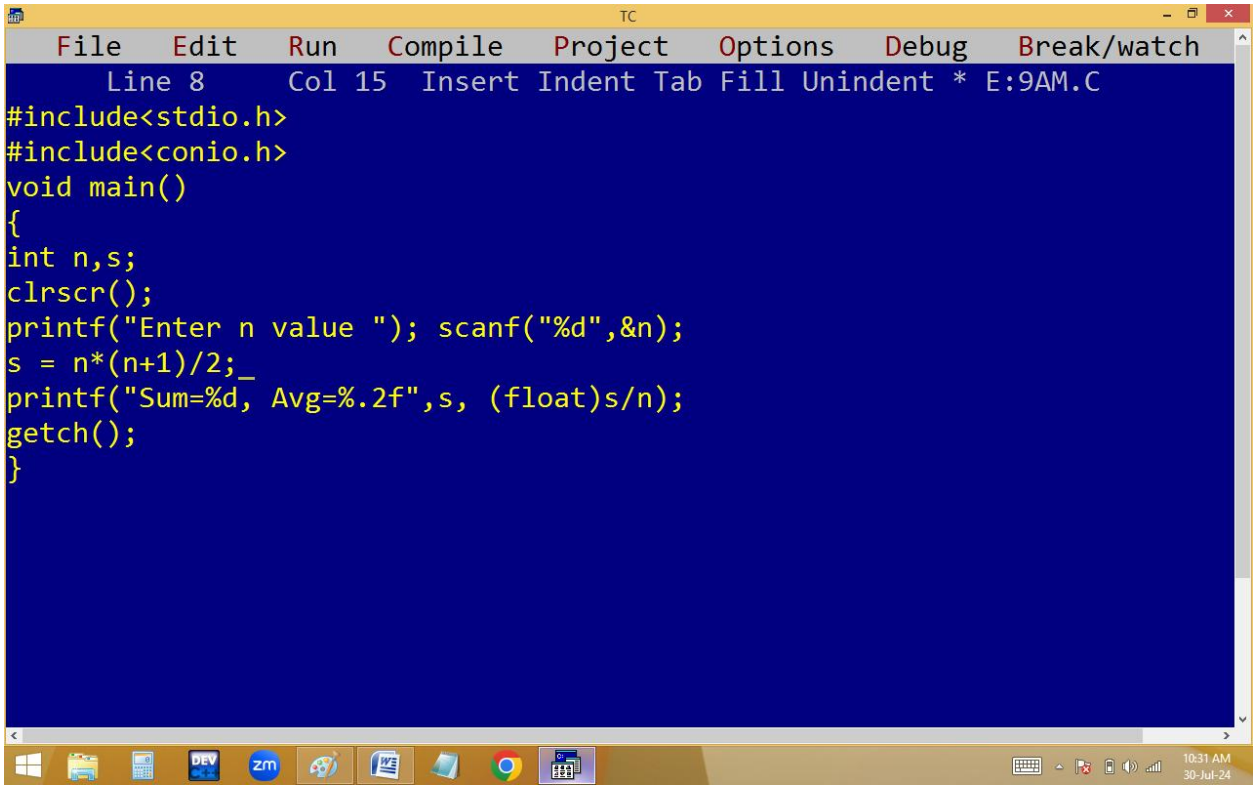
n=4 ➔ 1+2+3+4=10 ➔ 10/4=2.5➔avg

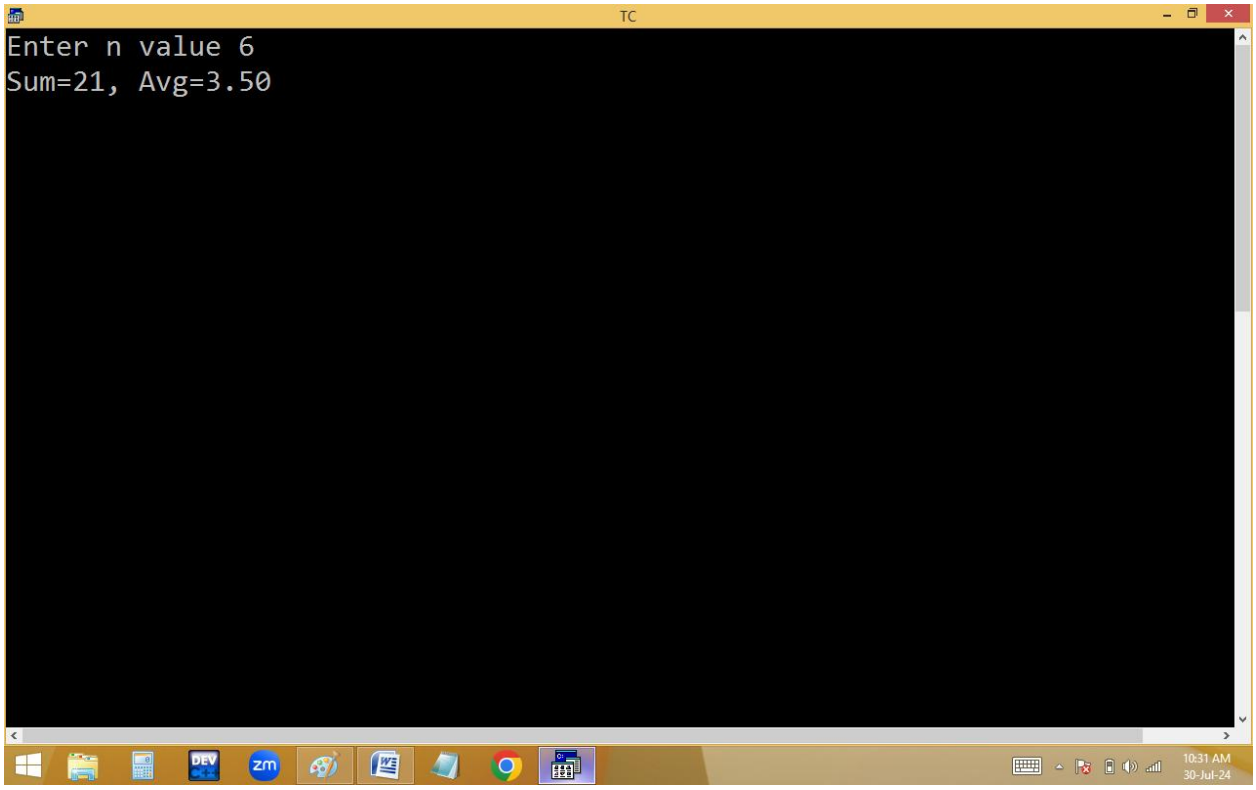


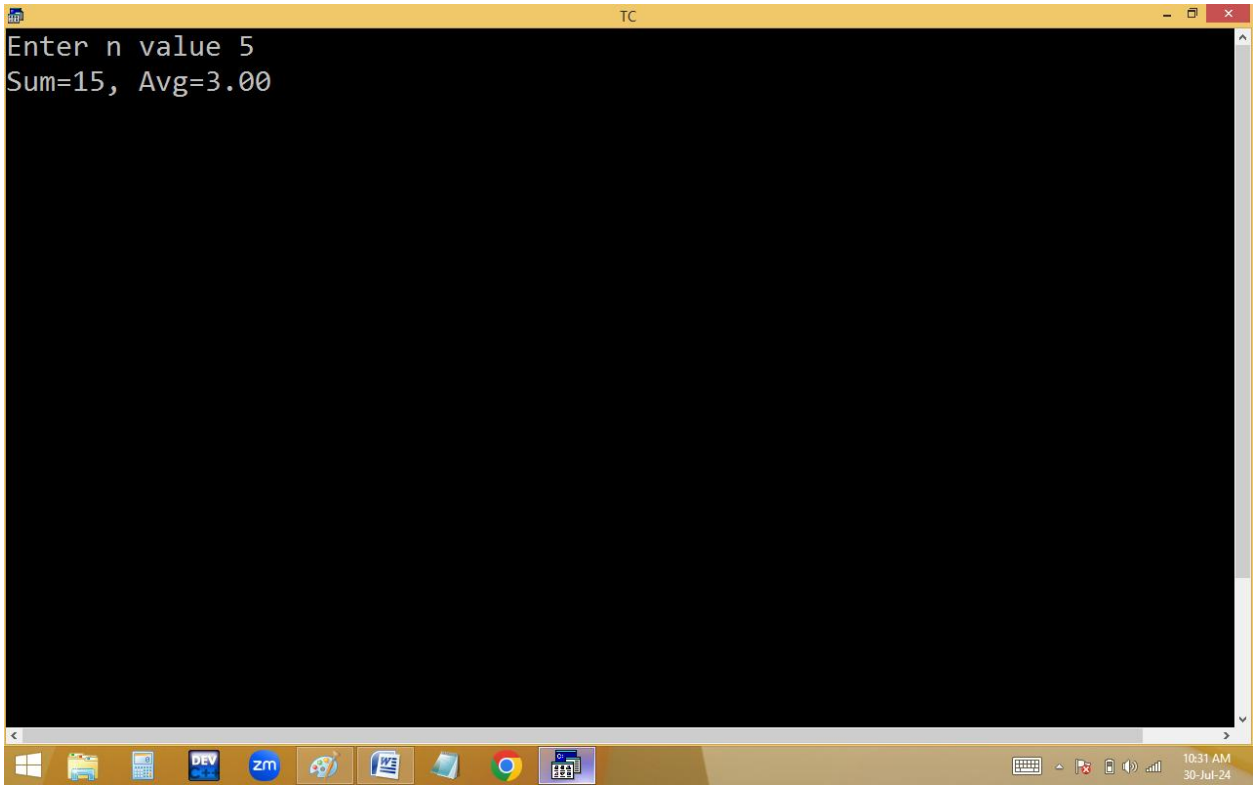

TC
Enter n value 4
Sum=10, Avg=2.50


i
n
5
1
2
3
4
5
6
0
1
3
6
10
15
/5 = 3
while(i<=n)
{
s=s+i;
i++;
}
p(s, s/n);

Without using loop/goto label:




TC
Enter n value 6
Sum=21, Avg=3.50

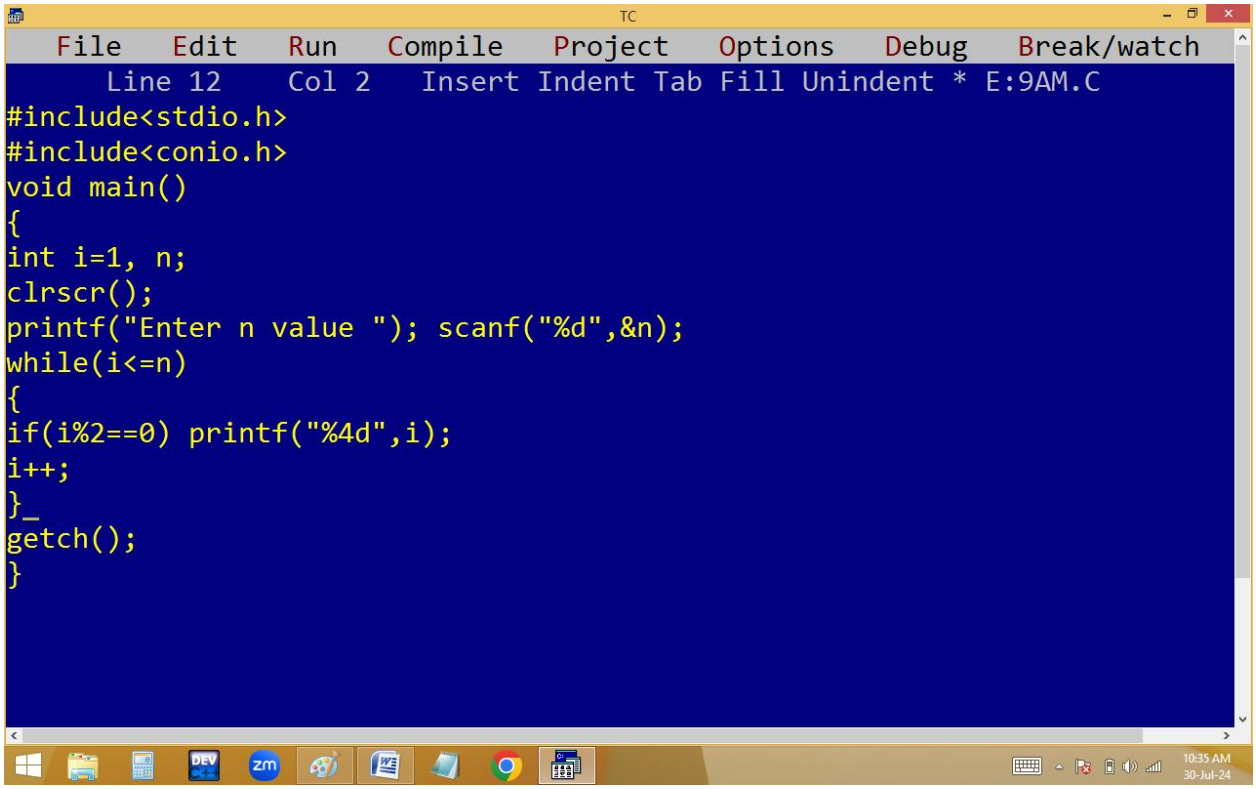

TC
Enter n value 5
Sum=15, Avg=3.00

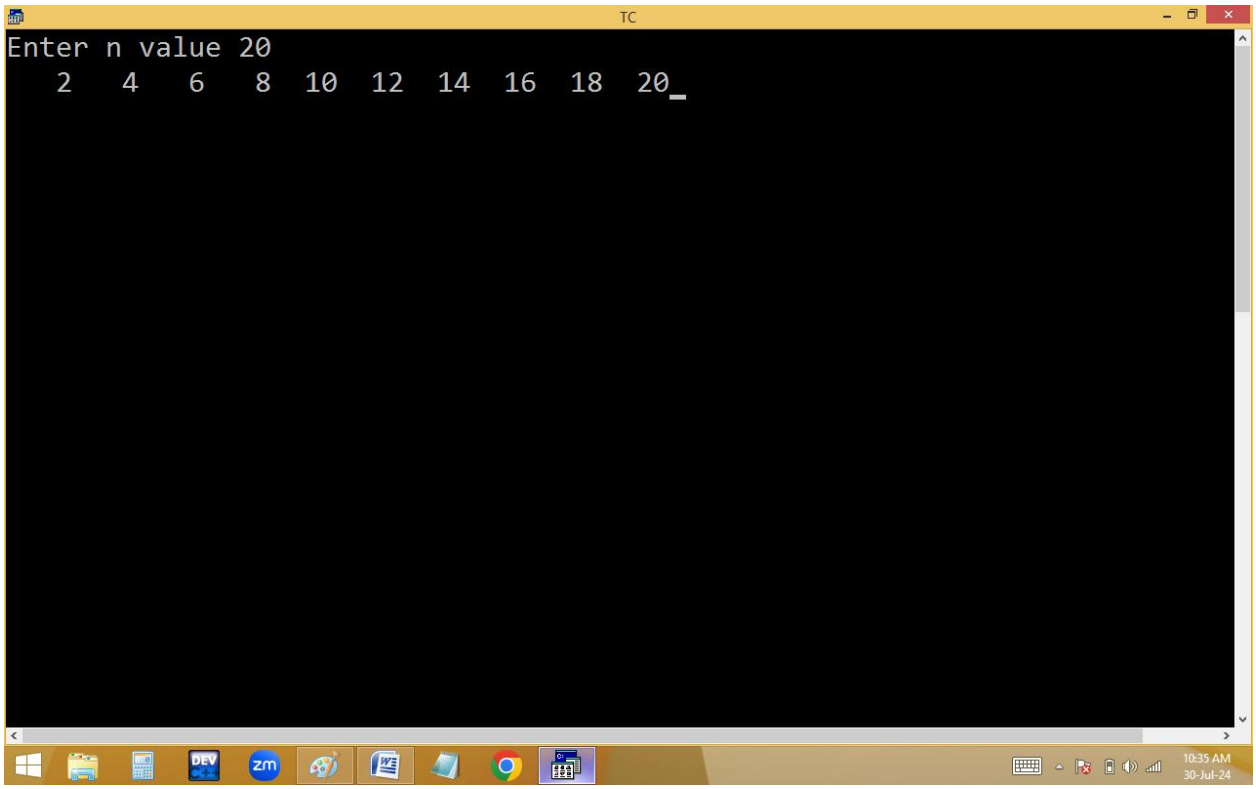
n=4

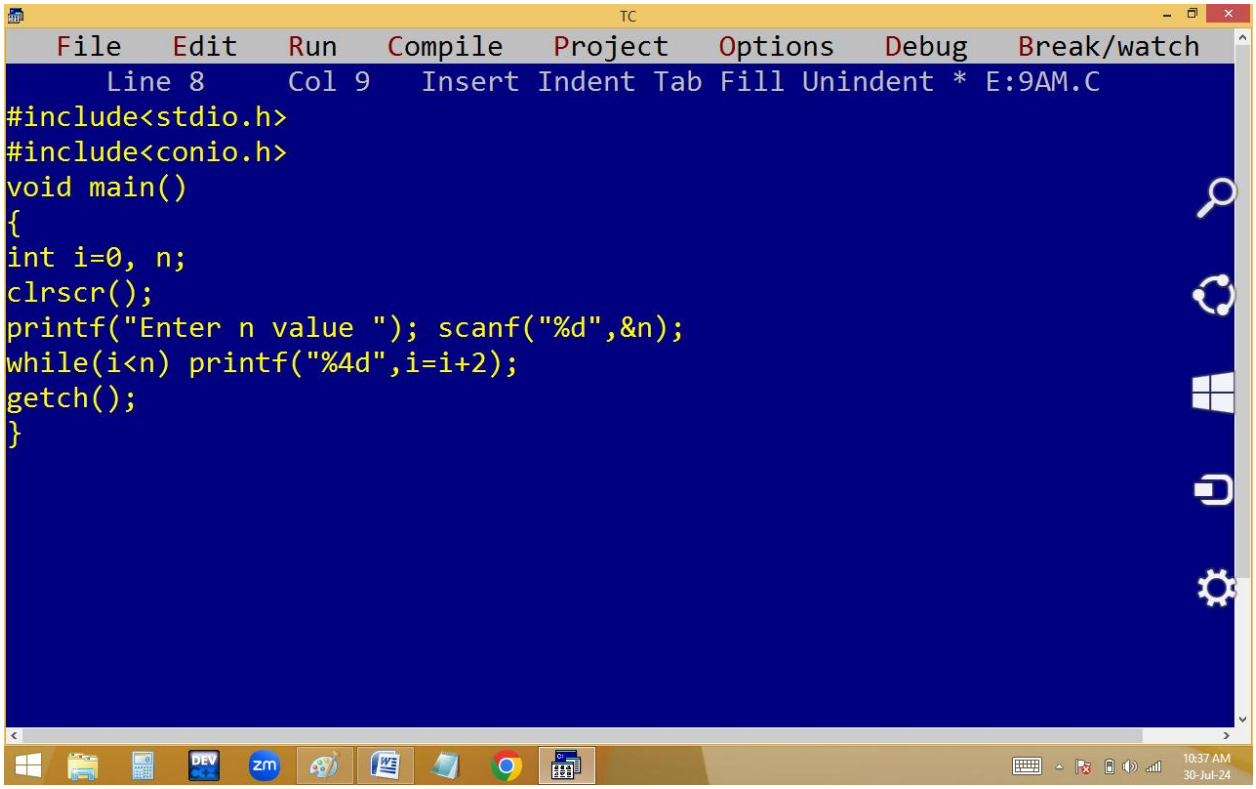
s = n*(n+1)/2;

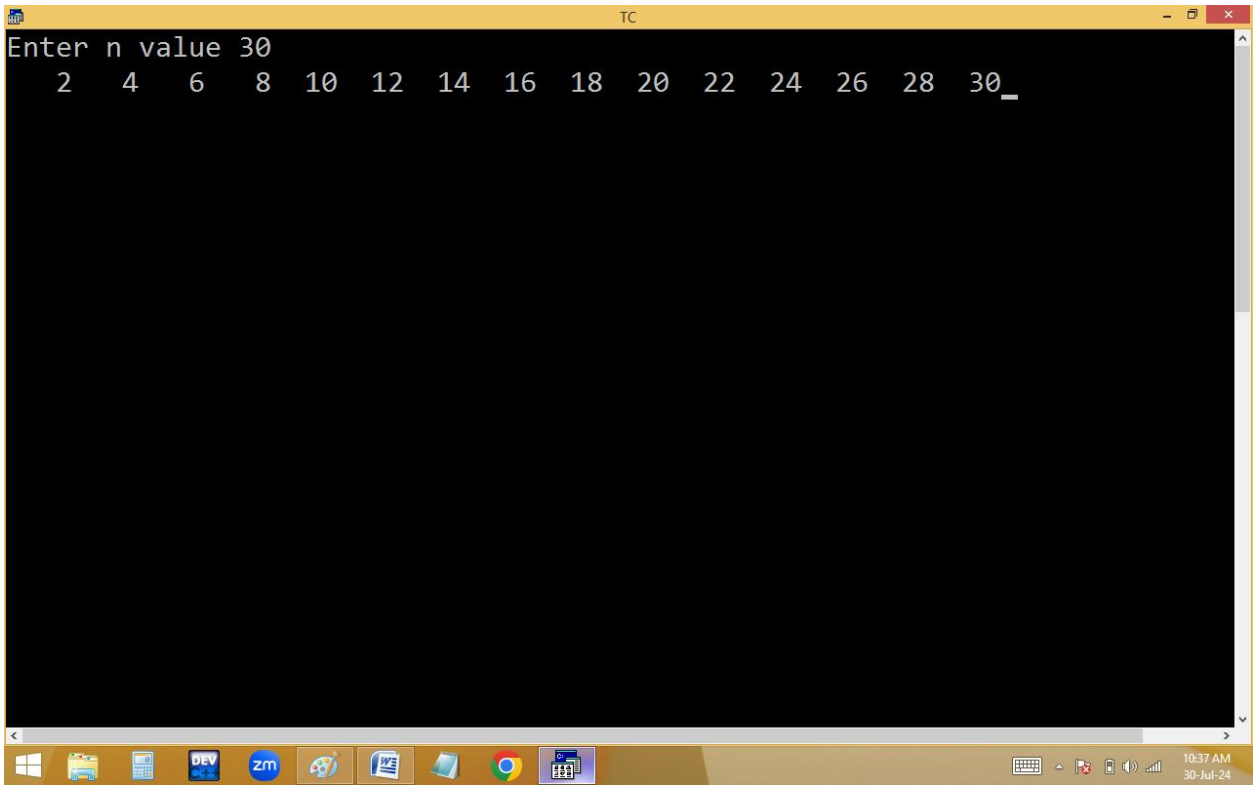
s = 4*(4+1)/2 ==> 4*5/2=10/4=2.5

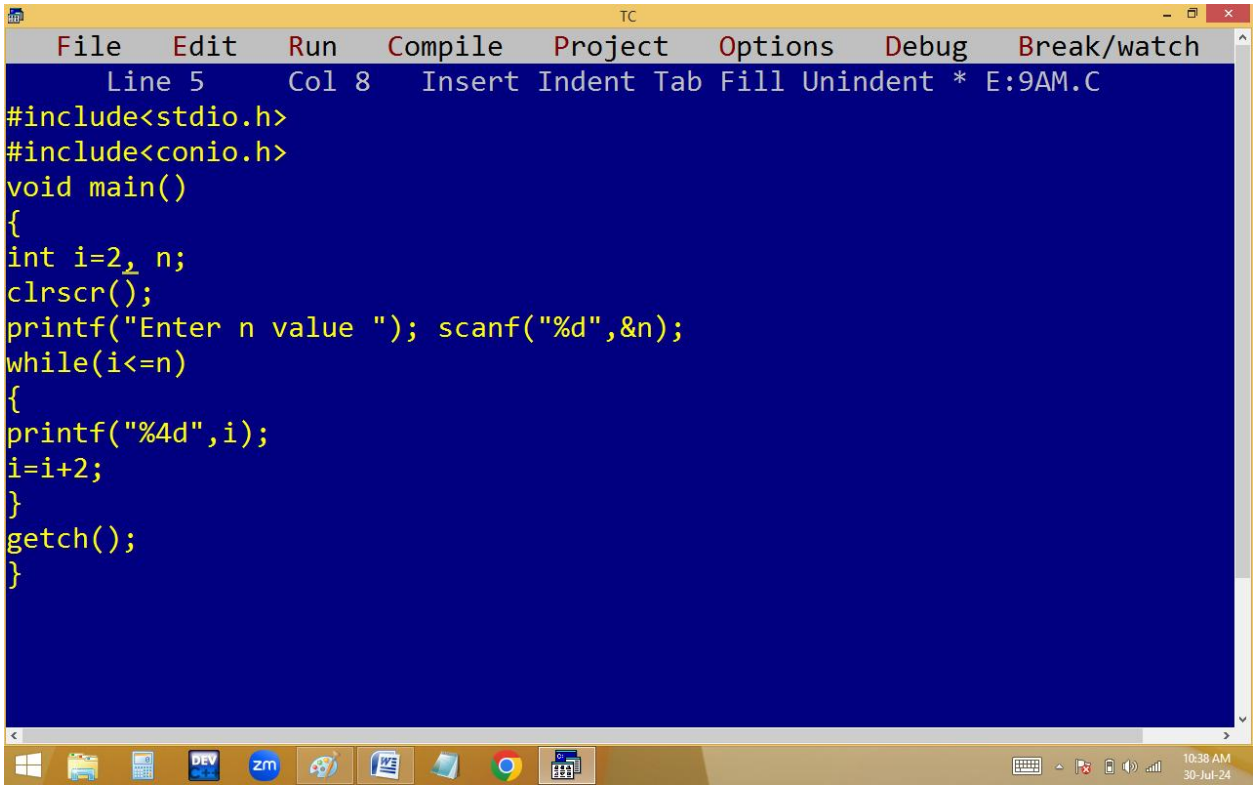
Printing 1..n even numbers.

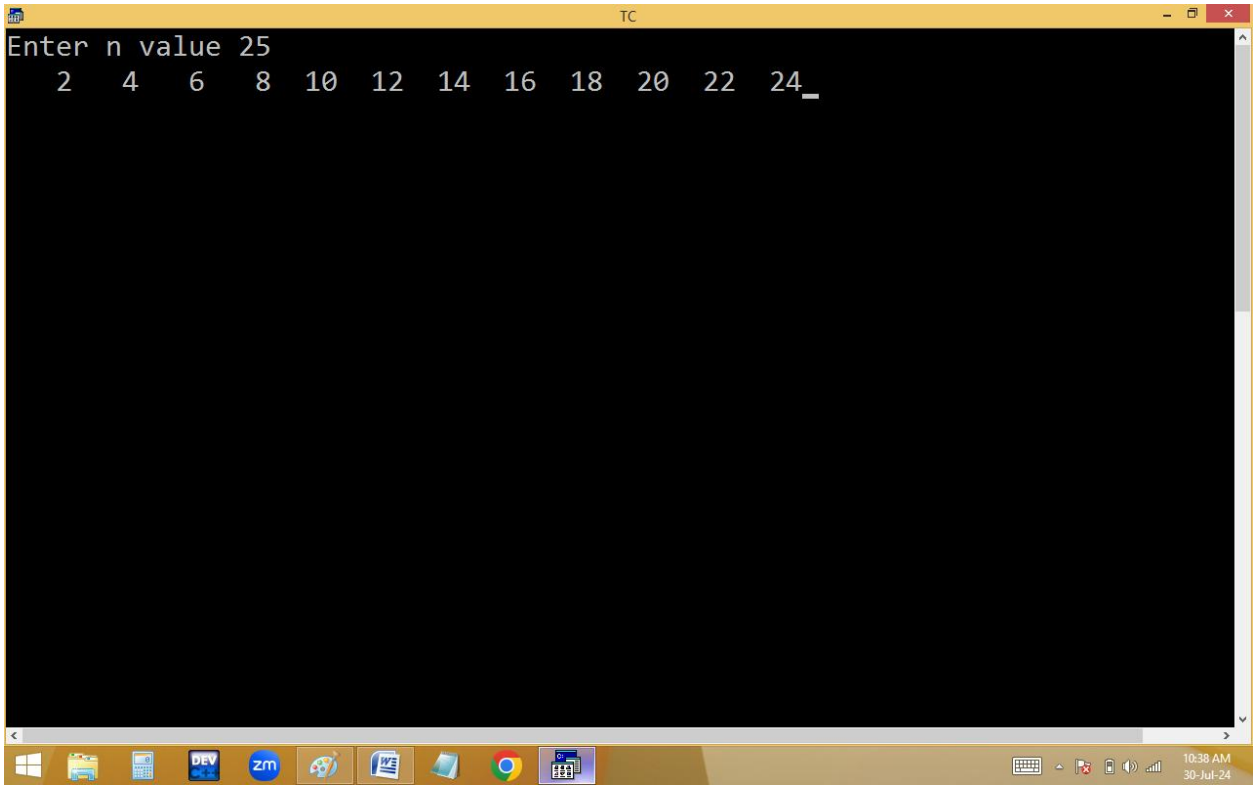

TC
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 2 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=1, n;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<=n)
{
if(i%2==0) printf("%4d",i);
i++;
}
getch();
}


TC
Enter n value 20
2 4 6 8 10 12 14 16 18 20

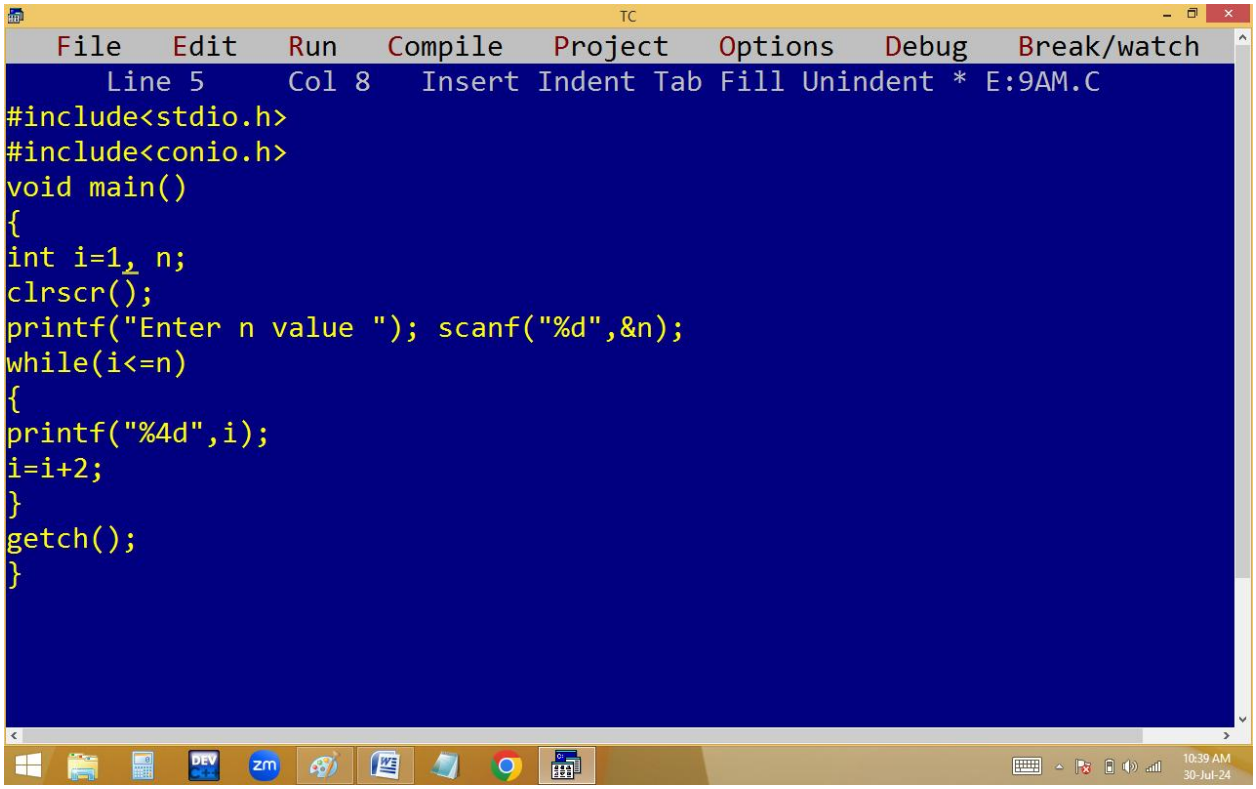

TC
File Edit Run Compile Project Options Debug Break/watch
Line 8 Col 9 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=0, n;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<n) printf("%4d",i=i+2);
getch();
}

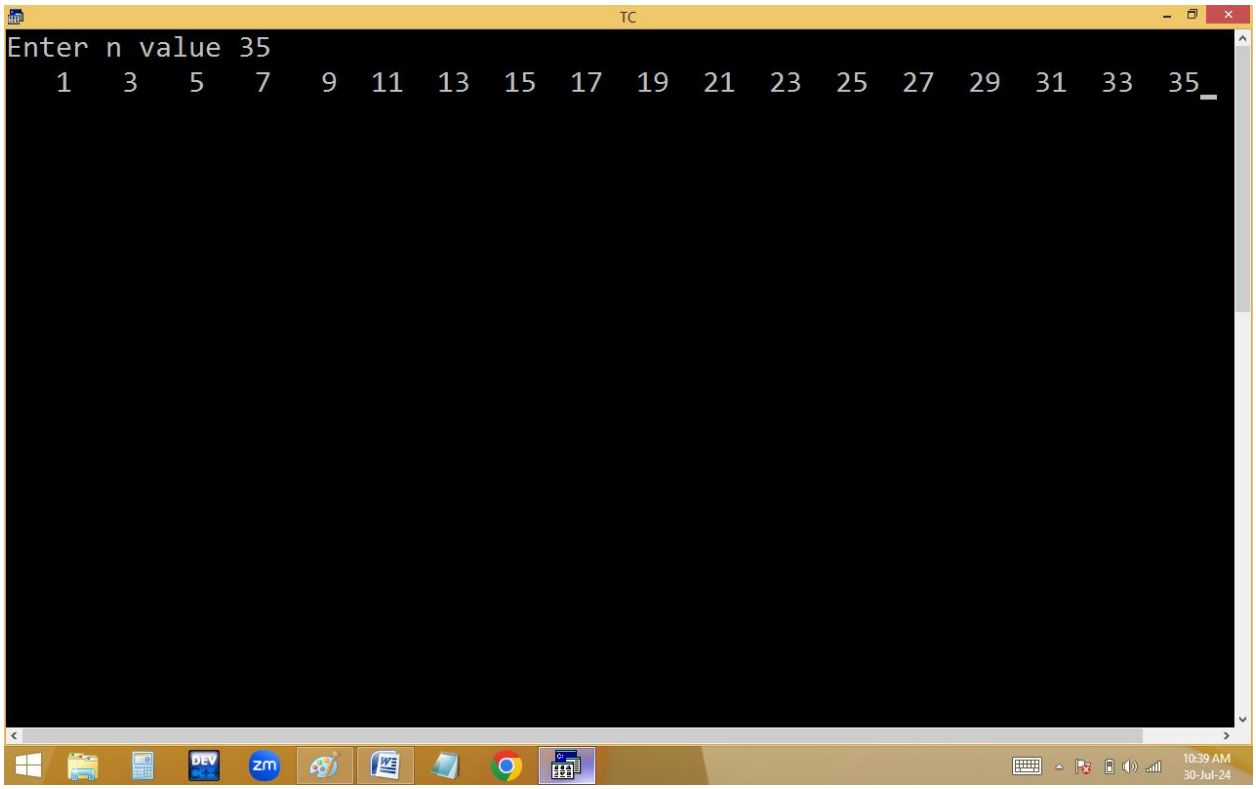

TC
Enter n value 30
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30

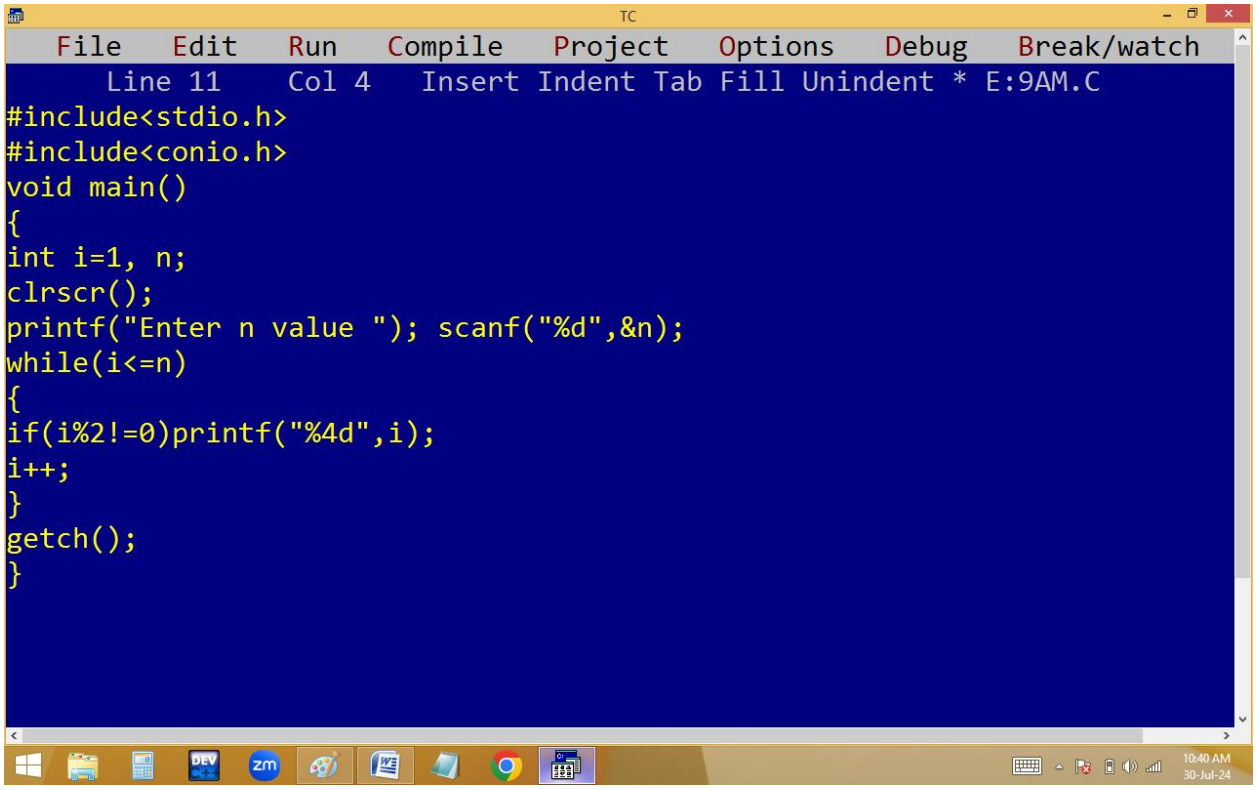

TC
File Edit Run Compile Project Options Debug Break/watch
Line 5 Col 8 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=2, n;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<=n)
{
printf("%4d",i);
i=i+2;
}
getch();
}

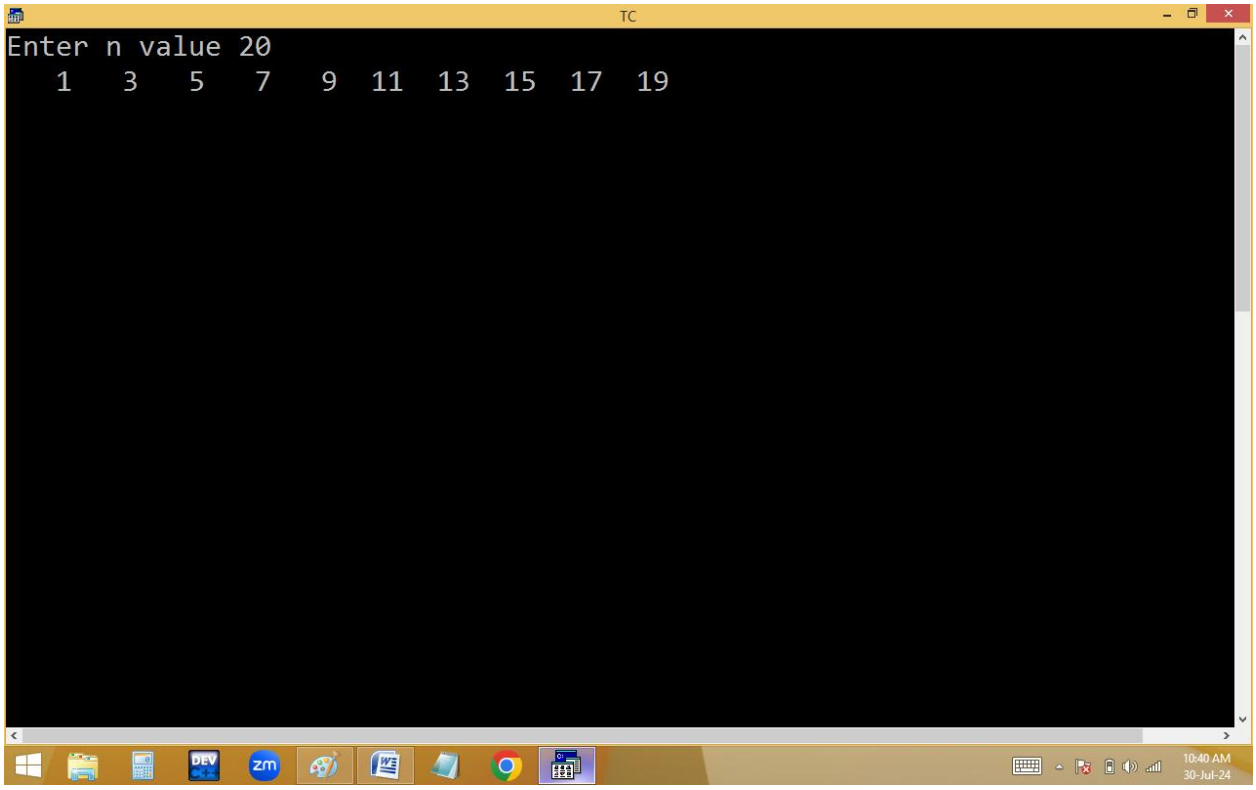


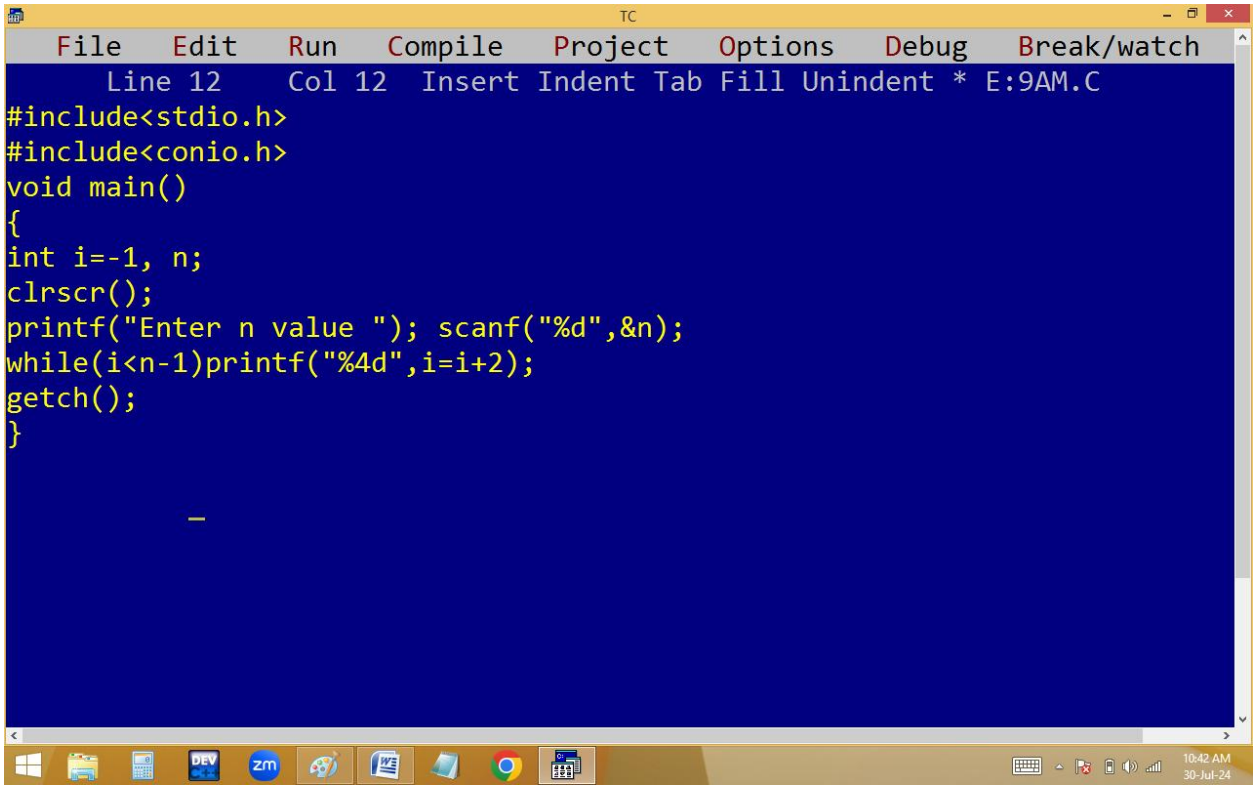
Odd numbers:

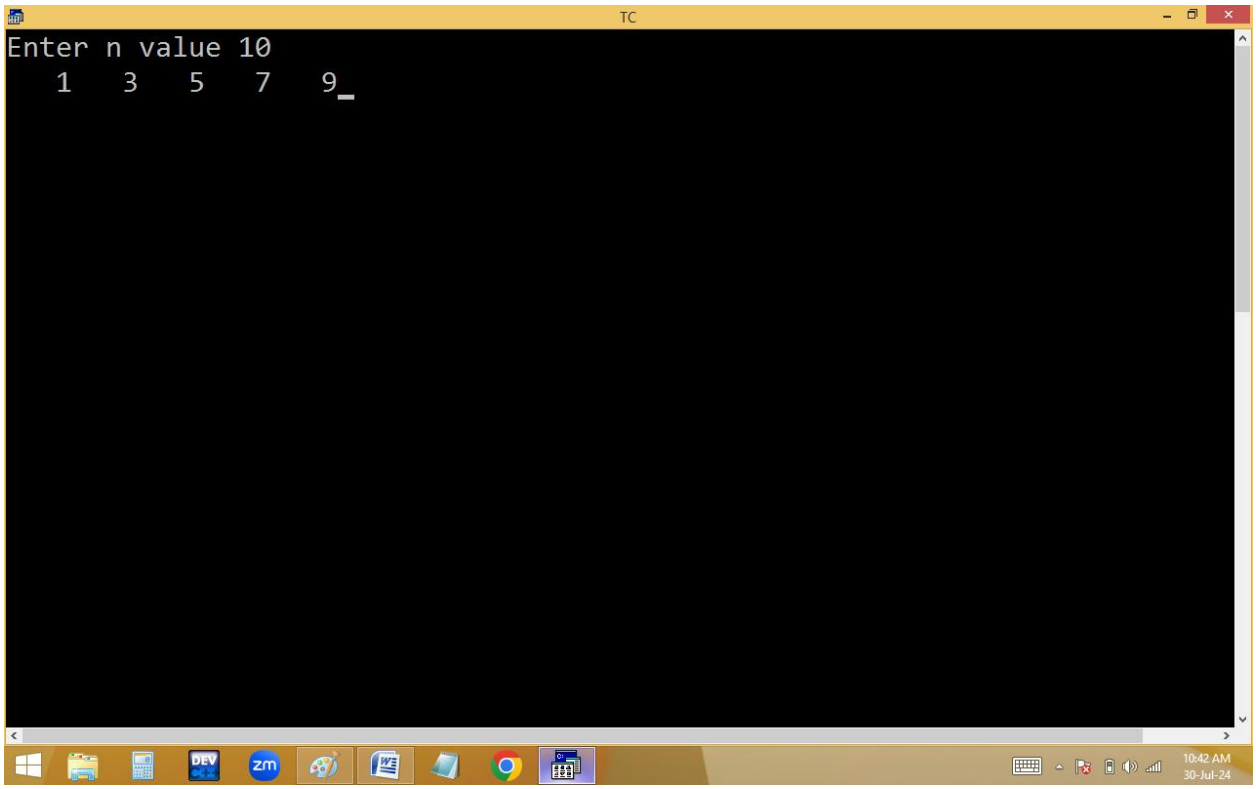

TC
File Edit Run Compile Project Options Debug Break/watch
Line 5 Col 8 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=1, n;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<=n)
{
printf("%4d",i);
i=i+2;
}
getch();
}

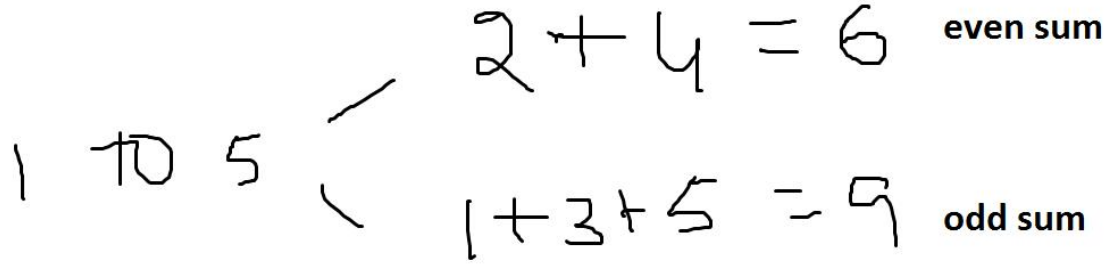

TC
Enter n value 35
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35


TC
File Edit Run Compile Project Options Debug Break/watch
Line 11 Col 4 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=1, n;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<=n)
{
if(i%2!=0)printf("%4d",i);
i++;
}
getch();
}


TC
Enter n value 20
1 3 5 7 9 11 13 15 17 19
10:40 AM
30-Jul-24


TC
File Edit Run Compile Project Options Debug Break/watch
Line 12 Col 12 Insert Indent Tab Fill Unindent * E:9AM.C
#include<stdio.h>
#include<conio.h>
void main()
{
int i=-1, n;
clrscr();
printf("Enter n value "); scanf("%d",&n);
while(i<n-1)printf("%4d",i=i+2);
getch();
}


TC
Enter n value 10
1 3 5 7 9

Home work:

Finding 1..n even , odd numbers sum