

Implementation of a rudimentary code snippets

a.

main.c	Output
<pre>1 #include <stdio.h> 2 int main() 3 { 4 int x = 3, y = 5, z = 15, w = -1, k; 5 6 k = x * w + y * (z=7); 7 printf("Value of given expression: %d", k); 8 return 0; 9 }</pre>	<pre>/tmp/7SxihJa0oS.o Value of given expression: 32</pre>

b.

main.c	Output
<pre>1 #include <stdio.h> 2 int main() 3 { 4 int w = 0, x = 2.5, y = 5, z = 3, r, s = 4, t = 5, u = -3, m, n; 5 double a = 2.36, b = 3.19, c = 3.0, d = 2.91726; 6 float o; 7 8 m = (int) (c * y / z + y / z * c); 9 n = (x - s * t * - c - u); 10 o = (float) (x + y < z + w && a > b - 17 * x ! x < 5); 11 12 printf("Value of given expression m= %d\n", m); 13 printf("Value of given expression n= %d\n", n); 14 printf("Value of given expression o= %f\n", o); 15 16 return 0; 17 } 18</pre>	<pre>/tmp/7SxihJa0oS.o Value of given expression m= 8 Value of given expression n= 65 Value of given expression o= 1.000000</pre>

Implementation of a rudimentary code snippets

c.

```
main.c
1  #include <stdio.h>
2
3  int main() {
4      int rows,col,i,j;
5      printf("Enter rows=");
6      scanf ("%d", &rows);
7      col=2*rows;
8      printf ("rows=%d col=%d\n", rows, col);
9
10     for (i=1; i<=rows; i++)
11     {
12         //printf("i=%d",i);
13         for (j=1; j<=col; j++)
14         {
15             if((j<=i) || (j>=(col-i+1)))
16                 printf ("**");
17             else
18                 printf (" ");
19         }
20         printf("\n");
21     }
22     return 0;
23 }
```

Output

```
/tmp/QBsxt05f1.o
Enter rows=3
rows=3 col=6
*  *
** **
*****
```

d.

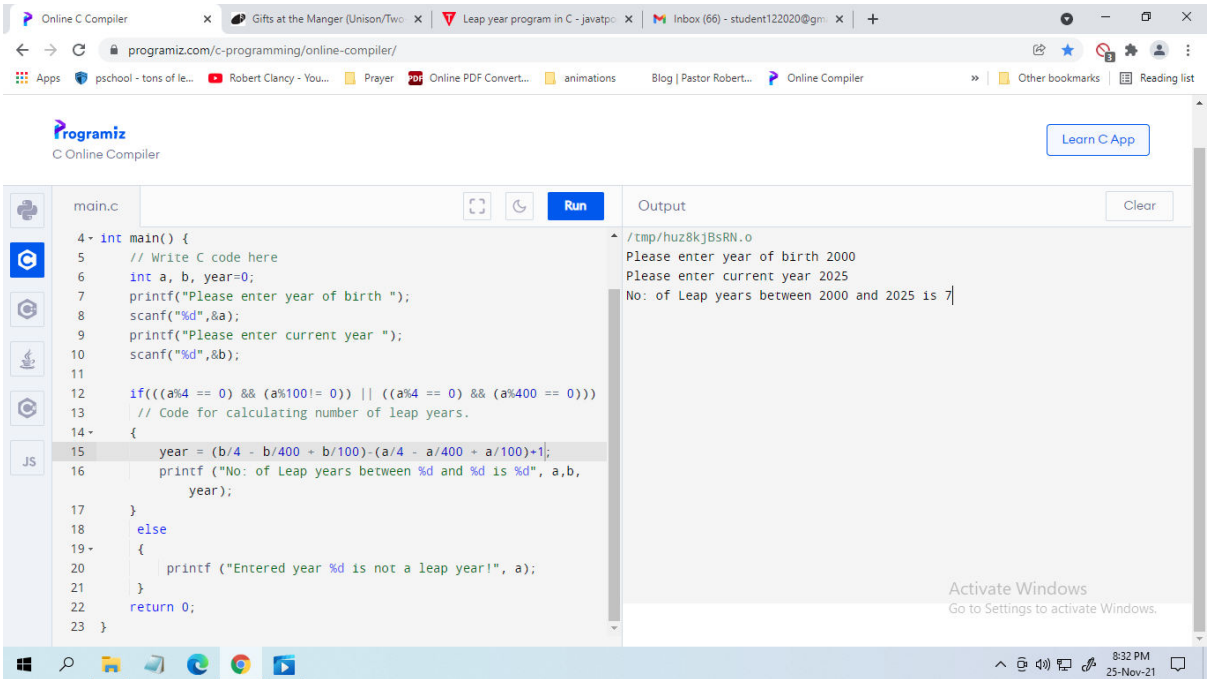
```
main.c
1  #include <stdio.h>
2
3  int main() {
4      int rows,col,i,j;
5      printf("Enter rows=");
6      scanf ("%d", &rows);
7      col=2*rows;
8      printf ("rows=%d col=%d\n", rows, col);
9
10     for (i=1; i<=rows; i++)
11     {
12         //printf("i=%d",i);
13         for (j=1; j<=col; j++)
14         {
15             if((j<=i) || (j>=(col-i+1)))
16                 printf ("**");
17             else
18                 printf (" ");
19         }
20         printf("\n");
21     }
22     return 0;
23 }
```

Output

```
/tmp/QBsxt05f1.o
Enter rows=/tmp/QBsxt05f1.o
Enter rows=5
rows=5 col=10
*  *
** **
*** ***
**** ****
*****
```

Implementation of a rudimentary code snippets

e.



The screenshot displays the Programiz Online C Compiler interface. The code editor on the left contains a C program named `main.c` that calculates the number of leap years between two input years, `a` and `b`. The program uses the following logic: it checks if a year is a leap year based on the rules $(a \% 4 == 0 \ \&\& \ (a \% 100 \neq 0)) \ || \ ((a \% 400 == 0))$. It then iterates from `a` to `b`, counting the number of leap years. The output on the right shows the program's execution with inputs 2000 and 2025, resulting in 7 leap years.

```
main.c
4 int main() {
5     // Write C code here
6     int a, b, year=0;
7     printf("Please enter year of birth ");
8     scanf("%d",&a);
9     printf("Please enter current year ");
10    scanf("%d",&b);
11
12    if(((a%4 == 0) && (a%100 != 0)) || ((a%400 == 0)))
13        // Code for calculating number of leap years.
14    {
15        year = (b/4 - b/400 + b/100) - (a/4 - a/400 + a/100) + 1;
16        printf ("No: of Leap years between %d and %d is %d", a,b,
17            year);
18    }
19    else
20    {
21        printf ("Entered year %d is not a leap year!", a);
22    }
23    return 0;
24 }
```

Output

```
/tmp/huz8kjBsRN.o
Please enter year of birth 2000
Please enter current year 2025
No: of Leap years between 2000 and 2025 is 7
```

Activate Windows
Go to Settings to activate Windows.