

LAB FILE

INTRODUCTION TO C PROGRAMMING



BATCH 2023-2027

BCA(Hons.) AI&DS

SUBMITTED BY:

SHARDUL RATURI

2316012137

SUBMITTED TO:

Mr. Rishi Kumar

Assistant. Prof (CS/IT)

GEU

INDEX

S.NO	NAME	PG.NO	REMARKS	T.SIGN
1	THIS IS MY FIRST PROGRAM	01-01		
2	ADDITION OF TWO NUMBERS	02-02		
3	AREA OF CIRCLE	03-03		
4	DIVIDE TWO NUMBERS	04-04		
5	FIND ASCII VALUE	05-05		
6	MULTIPLY FLOATING POINT NUMBERS	06-06		
7	SWAP TWO NUMBERS USING 3RD VARIABLE	07-08		
8	SWAP TWO NUMBERS WITHOUT USING 3RD VARIABLE	09-10		
9	SWAP THREE NUMBER WITHOUT USING 3RD VARIABLE	11-12		
10	AREA OF CIRCLE	13-13		
11	AREA OF SQUARE	14-14		
12	AREA OF RIGHT ANGLED TRIANGLE	15-15		
13	AREA OF ISOSCELES TRIANGLE	16-16		
14	AREA OF TRIANGLE WITH ANY THREE SIDES	17-18		

15	AREA AND VOLUME OF A CUBE	19-19		
16	AREA AND VOLUME OF A CUBOID	20-20		
17	FIND THE LARGEST NUMBER USING THE LOGICAL AND OPERATOR	21-21		
18	VALIDATE THE USERNAME AND PASSWORD ENTERED BY THE USER IS CORRECT OR NOT USING THE PREDEFINED USERNAME AND PASSWORD	22-23		
19	TO INPUT THE POSITIVE NUMBER FROM THE USER TO PERFORM THE LEFT SHIFT OPERATOR	24-24		
20	TO INPUT THE POSITIVE NUMBER FROM THE USER TO PERFORM THE RIGHT SHIFT OPERATOR	25-25		
21	TO PERFORM THE PRE-INCREMENT AND PRE-DECREMENT OPERATOR ON TWO INTEGER AND PRINT BOTH ORIGINAL VALUE AND UPDATE VALUE	26-26		

22	TO PERFORM THE POST-INCREMENT AND POST-DECREMENT OPERATOR ON TWO INTEGER AND PRINT BOTH ORIGINAL VALUE AND UPDATE VALUE	27-27		
23	TO CHECK WHETHER FOR AN INTEGER NUMBER THAT IT IS DIVISIBLE BY 9 OR 7 USING OR OPERATOR	28-28		
24	TO IDENTIFY GENDER IN SINGLE CHARACTER AND PRINT FULL GENDER(EX: IF INPUT IS 'M' OR 'M'-IT SHOULD PRINT "MALE")	29-29		
25	TO PRINT ALL NATURAL NUMBERS IN REVERSE (FROM N TO 1)	30-31		
26	TO PRINT ALL APLPHABET FROM A TO Z	31-32		
27	TO PRINT ALL NATURAL NUMBER FROM 1TON	32-33		
28	TO PRINT ALL EVEN NUMBER FROM 1 TO 100	33-34		
29	TO PRINT ALL ODD NUMBER BETWEEN 1 TO 100	34-36		

30	TO FIND SUM OF ALL NATURAL NUMBERS BETWEEN 1 TO N	36-37		
31	TO FIND THE SUM OF ALL EVEN NUMBER FROM 1 TO N	38-39		
32	TO FIND THE SUM OF ALL ODD NUMBER FROM 1 TO N	39-40		
33	TO PRINT MULTIPLICATION TABLE OF ANY NUMBER	41-42		
34	TO COUNT NUMBERS OF DIGIT IN A NUMBER	42-43		
35	TO FIND FIRST AND LAST DIGIT OF A NUMBER	43-44		
36	TO FIND SUM OF FIRST AND LAST DIGIT OF A NUMBER	45-46		
37	TO SWAP FIRST AND LAST DIGIT OF A NUMBER	46-47		
38	TO CALCULATE SUM OF DIGIT IN A NUMBER	47-48		
39	TO CALCULATE PRODUCT OF DIGIT OF A NUMBER	49-50		
40	TO ENTER A NUMBER AND PRINT ITS REVERSE	50-51		
41	TO CHECK WHETHER NUMBER IS PALINDROME OR NOT	53-54		

42	TO FIND ALL FACTORS OF A NUMBER	54-55		
43	TO FIND FACTORIAL OF A NUMBER	55-56		
44	TO FIND HCF OF TWO NUMBER	56-57		
45	TO FIND LCM OF TWO NUMBER	58-59		
46	TO CHECK WHETHER NUMBER IS PRIME OR NOT	59-60		
47	TO PRINT ALL PRIME NUMBER BETWEEN 1 TO N	61-62		
48	TO FIND SUM OF ALL PRIME NUMBER BETWEEN 1 TO N	62-63		
49	TO FIND ALL PRIME FACTORS OF A NUMBER	63-64		
50	TO CHECK WHETHER A NUMBER IS ARMSTRONG OR NOT	64-65		
51	TO PRINT ALL ARMSTRONG NUMBER BETWEEN 1 TO N	66-67		
52	TO CHECK WHETHER NUMBER IS PERFECT NUMBER OR NOT	67-68		
53	TO PRINT ALL PERFECT NUMBER BETWEEN 1 TO	69-70		

	N			
54	TO CHECK WHETHER NUMBER IS STRONG NUMBER OR NOT	70-72		
55	TO PRINT ALL STRONG NUMBER BETWEEN 1 TO N	72-74		
56	TO PRINT FIBONACCI SERIES UPTO N TERMS	74-75		
57	TO FIND ONE'S COMPLEMENT OF A BINARY NUMBER	75-77		
58	TO FIND TWO'S COMPLEMENT OF A BINARY NUMBER	77-80		
59	TO CONVERT BINARY NUMBER TO OCTAL NUMBER	80-81		
60	TO CONVERT BINARY NUMBER TO DECIMAL NUMBER	82-83		
61		84-87		
62	TO CONVERT OCTAL NUMBER TO BINARY NUMBER	88-89		
63	TO CONVERT OCTAL NUMBER TO DECIMAL NUMBER	89-90		
64	TO CONVERT OCTAL	91-93		

	NUMBER TO HEXADECIMAL NUMBER			
65	TO CONVERT DECIMAL NUMBER TO BINARY NUMBER	93-94		
66	TO FIND MAXIMUM BETWEEN TWO NUMBERS	95-96		
67	TO FIND MAXIMUM BETWEEN THREE NUMBERS	96-98		
68	TO CHECK WHETHER NUMBER IS POSITIVE NEGATIVE OR ZERO	98-99		
69	TO CHECK WHETHER NUMBER IS DIVISIBLE BY 5 AND 11 OR NOT	99-100		
70	TO CHECK WHETHER NUMBER IS EVEN OR ODD	101-102		
71	TO CHECK WHETHER A YEAR IS LEAP YEAR OR NOT	102-103		
72	TO CHECK WHETHER A CHARACTER IS ALPHABET OR NOT	103-104		
73	TO INPUT AND ALPHABET AND CHECK WHETHER IT IS VOWEL OR	104-106		

	CONSONANT			
74	TO INPUT ANY CHARACTER AND CHECK WHETHER IT IS ALPHABET DIGIT OR SPECIAL CHARACTER	106-107		
75	TO CHECK WHETHER A CHARACTER IS UPPERCASE OR LOWERCASE ALPHABET	107-109		
76	TO INPUT WEEK NUMBERS AND PRINT WEEK DAYS	109-111		
77	TO INPUT MONTH NUMBER AND PRINT NUMBER OF DAYS IN THAT MONTH	111-114		
78	TO COUNT TOTAL NUMBER OF NOTES IN GIVEN AMOUNT	114-117		
79	TO INPUT ANGLE OF A TRIANGLE AND CHECK WHETHER TRIANGLE IS VALID OR NOT	117-118		
80	TO INPUT ALL SIDES OF A TRIANGLE AND CHECK WHETHER TRIANGLE IS VALID OR NOT	118-120		
81	TO CHECK WHETHER THE TRIANGLE IS	120-121		

	EQUILATERAL OR SCALE IN TRIANGLE			
82	TO FIND ALL ROOTS OF A QUADRATIC EQUATION	121-123		
83	TO CALCULATE PROFIT AND LOSS	123-125		
84	TO INPUT MARKS OF FIVE SUBJECTS PHYSICS, CHEMISTRY, BIOLOGY, MATHEMATICS AND COMPUTER. CALCULATE PERCENTAGE AND GRADE ACCORDING TO FOLLOWING: PERCENTAGE $\geq 90\%$: GRADE A PERCENTAGE $\geq 80\%$: GRADE B PERCENTAGE $\geq 70\%$: GRADE C PERCENTAGE $\geq 60\%$: GRADE D PERCENTAGE $\geq 40\%$: GRADE E PERCENTAGE $< 40\%$: GRADE F	125-127		
85	TO INPUT BASIC SALARY OF AN EMPLOYEE AND CALCULATE ITS GROSS SALARY ACCORDING TO FOLLOWING: BASIC SALARY ≤ 10000 : HRA = 20%, DA = 80% BASIC SALARY ≤ 20000 : HRA = 25%, DA = 90% BASIC SALARY > 20000 : HRA = 30%, DA = 95%	127-129		
86	TO INPUT ELECTRICITY UNIT CHARGES AND CALCULATE TOTAL ELECTRICITY BILL ACCORDING TO THE GIVEN CONDITION: FOR FIRST 50 UNITS RS. 0.50/UNIT FOR NEXT 100 UNITS RS. 0.75/UNIT FOR NEXT 100 UNITS RS. 1.20/UNIT FOR UNIT ABOVE 250 RS. 1.50/UNIT	129-131		

	AN ADDITIONAL SURCHARGE OF 20% IS ADDED TO THE BILL			
87	TO CONVERT SPECIFIED DAYS INTO YEARS WEEKS AND DAYS	131-132		
88	PYRAMID STAR PATTERN	133-134		
89	HOLO PYRAMID STAR PATTERN	134-135		
90	INVERTED PYRAMID STAR PATTERN	136-137		
91	HOLLOW INVERTED STAR PATTERN	137-138		
92	HALF DIAMOND STAR PATTERN	139-140		
93	MIRRORED HALF DIAMOND STAR PATTERN	140-142		
94	NUMBER PATTERN 1	142-143		
95	NUMBER PATTERN 2	143-145		
96	NUMBER PATTERN 3	145-147		
97	NUMBER PATTERN 4	147-149		
98	NUMBER PATTERN 5	149-151		
99	NUMBER PATTERN 6	151-153		
100	TO READ N NUMBERS OF VALUES IN AN ARRAY AND DISPLAY THEM IN REVERSE ORDER	153-154		
101	TO FIND THE SUM OF ALL ELEMENTS OF THE ARRAY	155-156		
102	TO COPY THE ELEMENTS OF ONE ARRAY INTO	156-157		

	ANOTHER ARRAY			
103	TO COUNT THE TOTAL NUMBERS OF DUPLICATE ELEMENTS IN AN ARRAY	157-159		
104	TO FIND THE MAXIMUM AND MINIMUM ELEMENTS IN AN ARRAY	159-161		
105	TO SORT THE ELEMENTS OF AN ARRAY IN DESCENDING ORDER	161-162		
106	TO SEPARATE ODD AND EVEN INTEGERS INTO SEPARATE ARRAY	162-164		
107	TO MERGE TWO ARRAYS OF THE SAME SIZE SORTED IN ASCENDING ORDER	164-166		
108	TO MERGE TWO ARRAYS OF THE SAME SIZE SORTED IN DESCENDING ORDER	167-169		
109	READ MATRIX ELEMENT AND DISPLAY	169-171		
110	MATRIX MULTIPLICATION AND DISPLAY	172-174		
111	ADDITION OF MATRIX AND DISPLAY	175-178		
112	SUBTRACTION OF MATRIX AND DISPLAY	179-182		
113	TRANSPOSE OF MATRIX	183-186		

	AND DISPLAY			
--	--------------------	--	--	--

❖ PROGRAM 1:- THIS IS MY FIRST PROGRAM

```
#include<stdio.h>
```

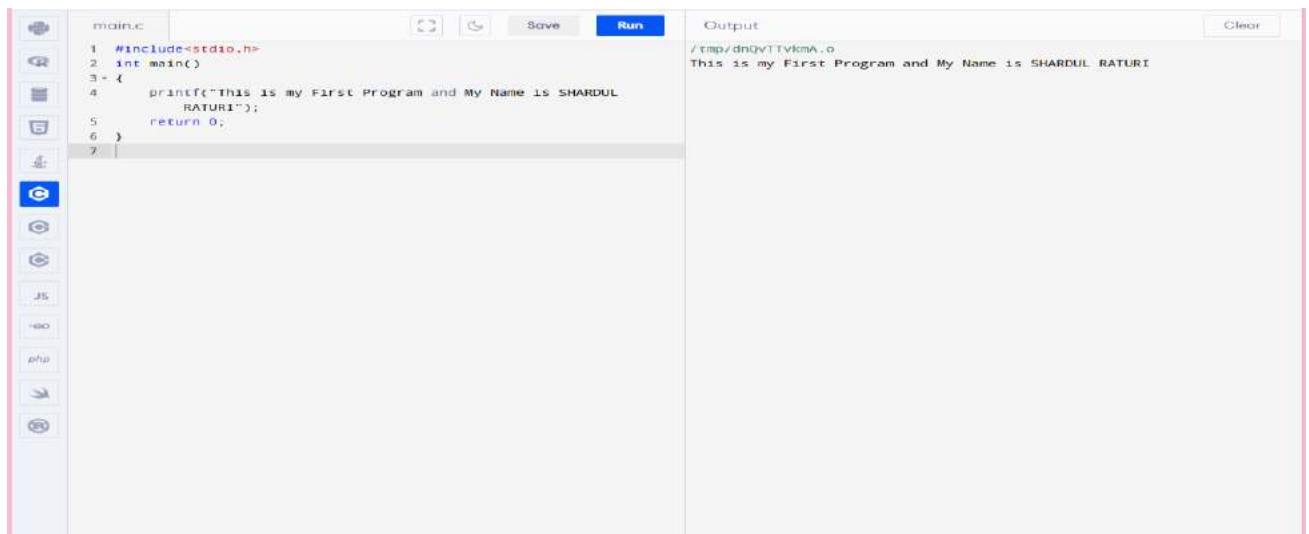
```
int main()
```

```
{
```

```
    printf("This is my First Program and My Name is SHARDUL  
RATURI");
```

```
    return 0;
```

```
}
```



❖ PROGRAM 2:- TO ADD TWO NUMBERS

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    float Num1,Num2,Sum;
```

```
    printf("\nENTER FIRST NUMBER:");
```

```
    scanf("%f",&Num1);
```

```
    printf("ENTER SECOND NUMBER:");
```

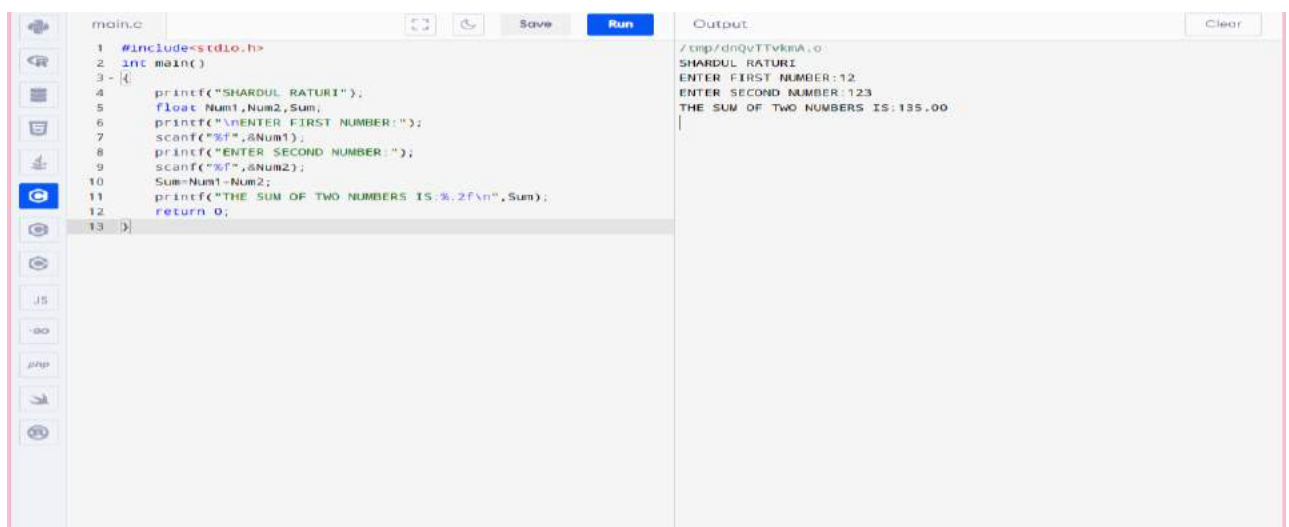
```
    scanf("%f",&Num2);
```

```
    Sum=Num1+Num2;
```

```
    printf("THE SUM OF TWO NUMBERS IS:%.2f\n",Sum);
```

```
    return 0;
```

```
}
```



The screenshot shows a C program in an IDE. The code is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float Num1,Num2,Sum;
6     printf("\nENTER FIRST NUMBER:");
7     scanf("%f",&Num1);
8     printf("ENTER SECOND NUMBER:");
9     scanf("%f",&Num2);
10    Sum=Num1+Num2;
11    printf("THE SUM OF TWO NUMBERS IS:%.2f\n",Sum);
12    return 0;
13 }
```

The output window shows the following text:

```
/tmp/dnQvTTvkmA.o
SHARDUL RATURI
ENTER FIRST NUMBER:12
ENTER SECOND NUMBER:123
THE SUM OF TWO NUMBERS IS:135.00
```

❖ PROGRAM 3:- TO FIND AREA OF CIRCLE

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    float Radius,Area,;
```

```
    printf("\nEnter Radius of a circle:");
```

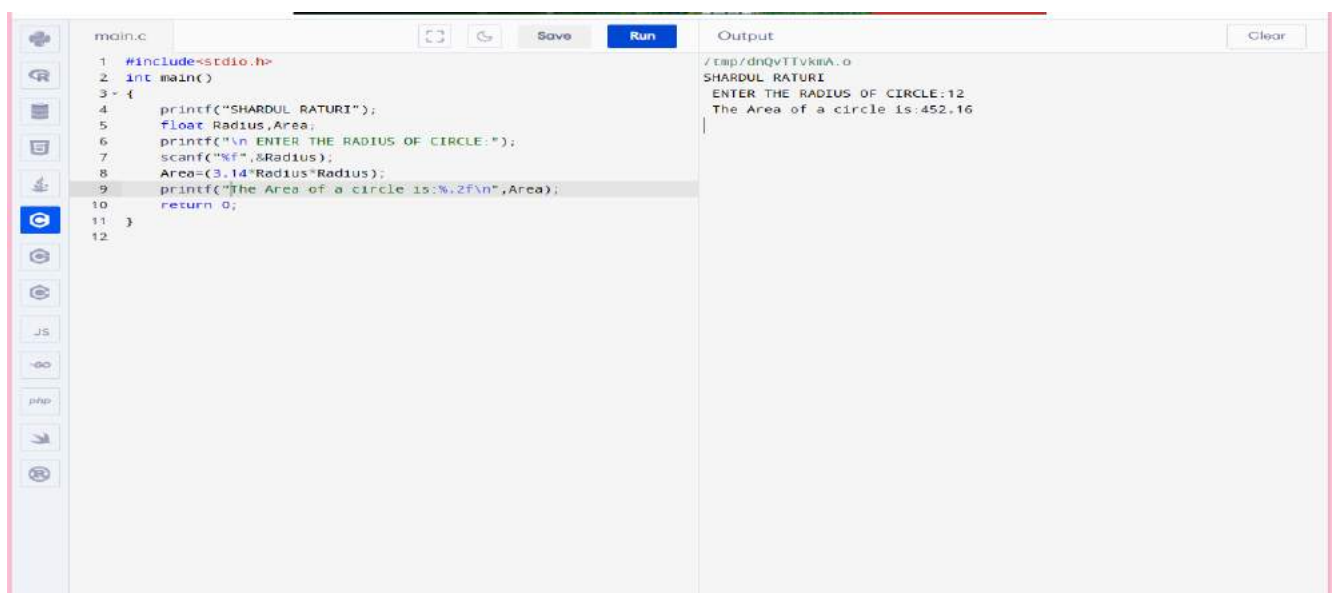
```
    scanf("%f",&Radius);
```

```
    Area=(3.14*Radius*Radius;
```

```
    printf("The Area of a circle is:%.2f\n",Area);
```

```
    return 0;
```

```
}
```



The screenshot shows a code editor window with a file named 'main.c'. The code is a C program to calculate the area of a circle. It includes the standard input/output header, defines the main function, declares variables for radius and area, prompts the user for the radius, reads the input, calculates the area using the formula $A = \pi r^2$ (approximated as 3.14), and prints the result. The program is compiled and run, showing the output in the 'Output' pane. The output shows the name 'SHARDUL RATURI', the prompt 'ENTER THE RADIUS OF CIRCLE:', the user input '12', and the calculated area '452.16'.

```
main.c 1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float Radius,Area;
6     printf("\n ENTER THE RADIUS OF CIRCLE:");
7     scanf("%f",&Radius);
8     Area=(3.14*Radius*Radius);
9     printf("The Area of a circle is:%.2f\n",Area);
10    return 0;
11 }
12
```

Output

```
/tmp/dnQvTTvkiA.o
SHARDUL RATURI
ENTER THE RADIUS OF CIRCLE:12
The Area of a circle is:452.16
```


❖ PROGRAM 4:- TO DIVIDE TWO NUMBERS

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    float Num1,Num2,Result;
```

```
    printf("\nEnter First Number:");
```

```
    scanf("%f",&Num1);
```

```
    printf("Enter Second Number:");
```

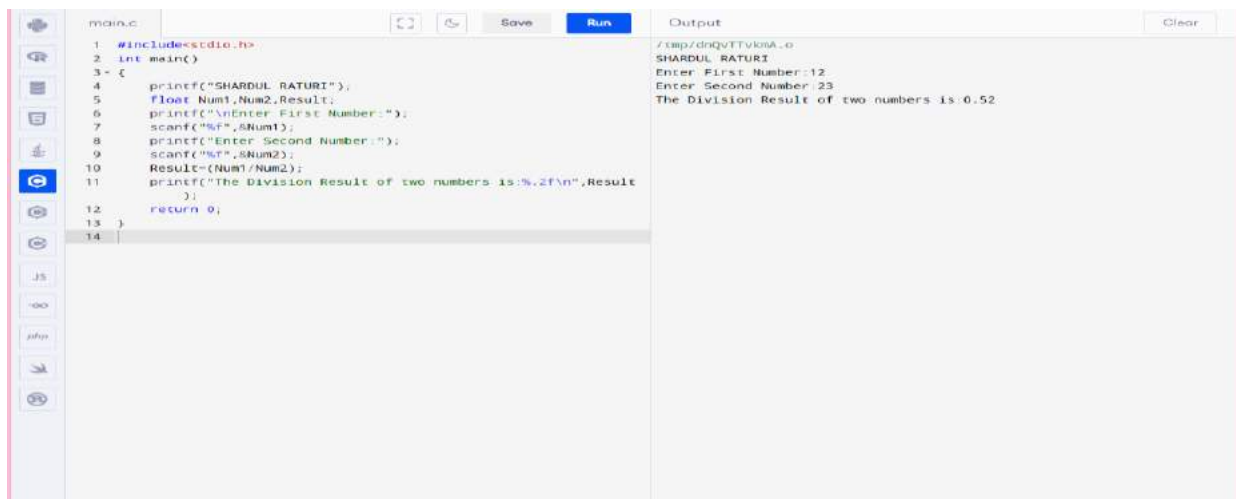
```
    scanf("%f",&Num2);
```

```
    Result=(Num1/Num2);
```

```
    printf("The Division Result of two numbers is:%.2f\n",Result);
```

```
    return 0;
```

```
}
```



The screenshot shows a C program in a code editor. The code is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float Num1,Num2,Result;
6     printf("\nEnter First Number:");
7     scanf("%f",&Num1);
8     printf("Enter Second Number:");
9     scanf("%f",&Num2);
10    Result=(Num1/Num2);
11    printf("The Division Result of two numbers is:%.2f\n",Result);
12
13    return 0;
14 }
```

The output window shows the following text:

```
/tmp/dnQvTTv10wA.o
SHARDUL RATURI
Enter First Number:12
Enter Second Number:23
The Division Result of two numbers is 0.52
```

❖ PROGRAM 5:- TO PRINT ASCII VALUE

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    char H;
```

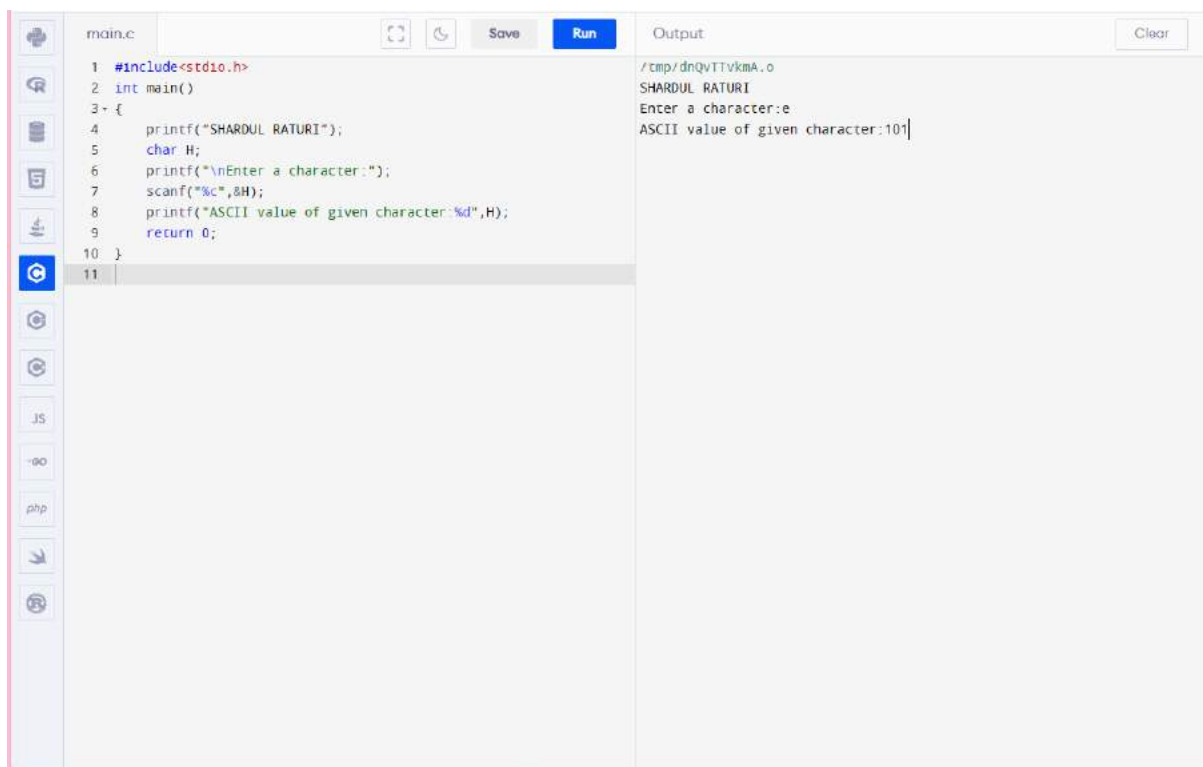
```
    printf("\nEnter a character:");
```

```
    scanf("%c",&H);
```

```
    printf("ASCII value of given character:%d",H);
```

```
    return 0;
```

```
}
```



The screenshot shows a code editor with a file named 'main.c'. The code is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     char H;
6     printf("\nEnter a character:");
7     scanf("%c",&H);
8     printf("ASCII value of given character:%d",H);
9     return 0;
10 }
11
```

The editor has a 'Run' button. The output window on the right shows the following text:

```
/cmp/dnQvTtVkmA.o
SHARDUL RATURI
Enter a character:e
ASCII value of given character:101
```

❖ PROGRAM 6:- TO MULTIPLY FLOATING NUMBERS

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    float num1, num2, result;
```

```
    printf("\nEnter the first number: ");
```

```
    scanf("%f", &num1);
```

```
    printf("Enter the second number: ");
```

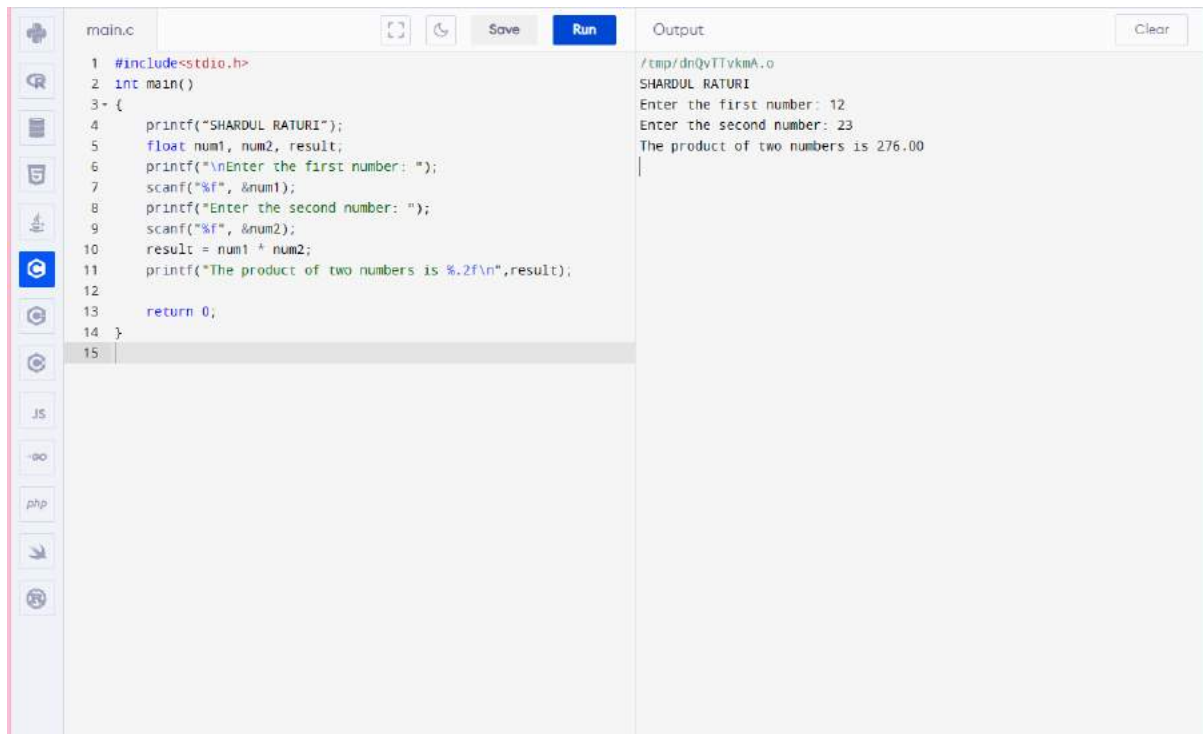
```
    scanf("%f", &num2);
```

```
    result = num1 * num2;
```

```
    printf("The product of two numbers is %.2f\n",result);
```

```
    return 0;
```

```
}
```



```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float num1, num2, result;
6     printf("\nEnter the first number: ");
7     scanf("%f", &num1);
8     printf("Enter the second number: ");
9     scanf("%f", &num2);
10    result = num1 * num2;
11    printf("The product of two numbers is %.2f\n", result);
12
13    return 0;
14 }
15
```

Output

```
/tmp/dnQyTtykmA.o
SHARDUL RATURI
Enter the first number: 12
Enter the second number: 23
The product of two numbers is 276.00
```

❖ PROGRAM 7:- TO SWAP TWO NUMBERS BY USING 3rd VARIABLE

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int N1,N2;
```

```
    int T;
```

```
    printf("\nEnter N1:");
```

```
    scanf("%d",&N1);
```

```
    printf("Enter N2:");
```

```
    scanf("%d",&N2);
```

```
    T=N1;
```

N1=N2;

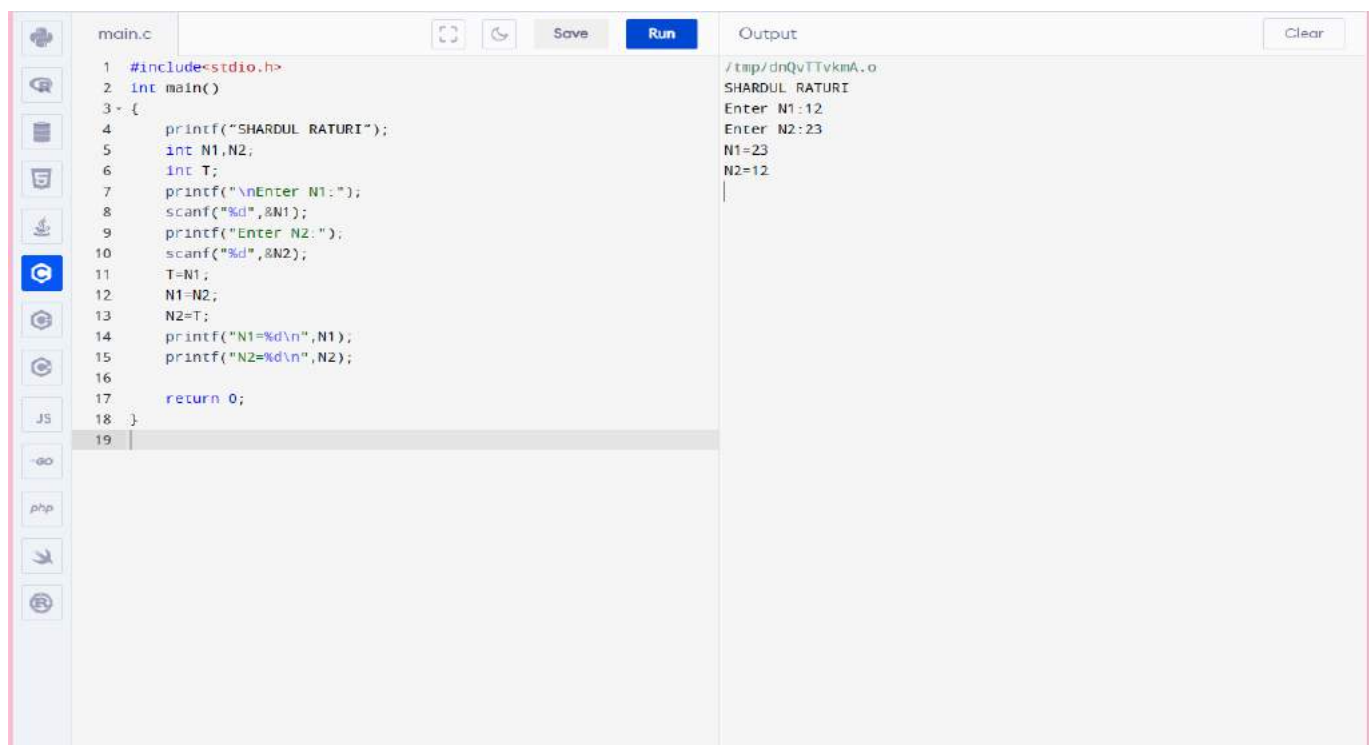
N2=T;

printf("N1=%d\n",N1);

printf("N2=%d\n",N2);

return 0;

}



```
main.c
1  #include<stdio.h>
2  int main()
3  {
4      printf("SHARDUL RATURI");
5      int N1,N2;
6      int T;
7      printf("\nEnter N1:");
8      scanf("%d",&N1);
9      printf("Enter N2:");
10     scanf("%d",&N2);
11     T=N1;
12     N1=N2;
13     N2=T;
14     printf("N1=%d\n",N1);
15     printf("N2=%d\n",N2);
16
17     return 0;
18 }
19
```

Output

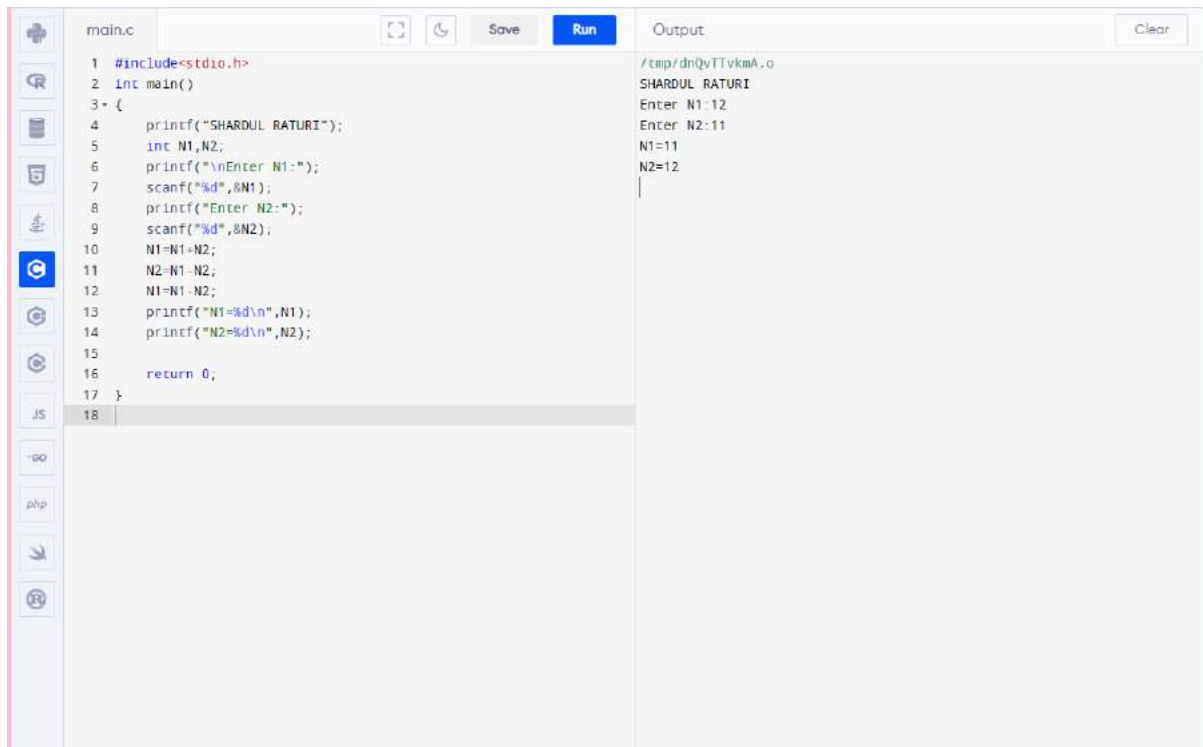
```
/tmp/dnQvTTykmA.o
SHARDUL RATURI
Enter N1:12
Enter N2:23
N1=23
N2=12
```

❖ PROGRAM 8:- TO SWAP TWO VARIABLES WITHOUT USING 3rd VARIABLE

```
#include<stdio.h>

int main()
{
    printf("SHARDUL RATURI");
    int N1,N2;
    printf("\nEnter N1:");
    scanf("%d",&N1);
    printf("Enter N2:");
    scanf("%d",&N2);
    N1=N1+N2;
    N2=N1-N2;
    N1=N1-N2;
    printf("N1=%d\n",N1);
    printf("N2=%d\n",N2);

    return 0;
}
```



```
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int N1,N2;
6     printf("\nEnter N1:");
7     scanf("%d",&N1);
8     printf("Enter N2:");
9     scanf("%d",&N2);
10    N1=N1+N2;
11    N2=N1-N2;
12    N1=N1-N2;
13    printf("N1=%d\n",N1);
14    printf("N2=%d\n",N2);
15
16    return 0;
17 }
```

Output

```
/tmp/dnQyTtykMA.o
SHARDUL RATURI
Enter N1:12
Enter N2:11
N1=23
N2=12
```

❖ PROGRAM 9:- TO SWAP THREE VARIABLES WITHOUT USING 3rd VARIABLE

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int N1,N2,N3;
```

```
    printf("\nEnter N1:");
```

```
    scanf("%d",&N1);
```

```
    printf("Enter N2:");
```

```
    scanf("%d",&N2);
```

```

printf("Enter N3:");

scanf("%d",&N3);

N1=N1+N2+N3;

N2=N1-(N2+N3);

N3=N1-(N2+N3);

N1=N1-(N2+N3);

printf("N1=%d\n",N1);

printf("N2=%d\n",N2);

printf("N3=%d\n",N3);

return 0;

}

```

```

main.c
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int N1,N2,N3;
6     printf("\nEnter N1:");
7     scanf("%d",&N1);
8     printf("Enter N2:");
9     scanf("%d",&N2);
10    printf("Enter N3:");
11    scanf("%d",&N3);
12    N1=N1+N2+N3;
13    N2=N1-(N2+N3);
14    N3=N1-(N2+N3);
15    N1=N1-(N2+N3);
16    printf("N1=%d\n",N1);
17    printf("N2=%d\n",N2);
18    printf("N3=%d\n",N3);
19    return 0;
20 }
21

```

Output

```

/cmp/dnQvTTvkMA.o
SHARDUL RATURI
Enter N1:12
Enter N2:13
Enter N3:14
N1=14
N2=12
N3=13

```


❖ PROGRAM 10:- TO FIND AREA OF RECTANGLE

```
#include<stdio.h>

int main()
{
    printf("SHARDUL RATURI");
    float Length,Breadth,Area;
    printf("\nENTER LENGTH:");
    scanf("%f",&Length);
    printf("ENTER BREADTH:");
    scanf("%f",&Breadth);
    Area=Length*Breadth;
    printf("THE AREA IS:%.2f\n",Area);
    return 0;
}
```

```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float Length,Breadth,Area;
6     printf("\nENTER LENGTH:");
7     scanf("%f",&Length);
8     printf("ENTER BREADTH:");
9     scanf("%f",&Breadth);
10    Area=Length*Breadth;
11    printf("THE AREA IS: %.2f\n",Area);
12    return 0;
13 }
14
```

Output

```
/tmp/dnQvTIVkMA.o
SHARDUL RATURI
ENTER LENGTH:12
ENTER BREADTH:13
THE AREA IS:156.00
```

❖ PROGRAM 11:- TO FIND AREA OF SQUARE

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    Printf("SHARDUL RATURI");
```

```
    float Side,Area;
```

```
    printf("\nENTER SIDE:");
```

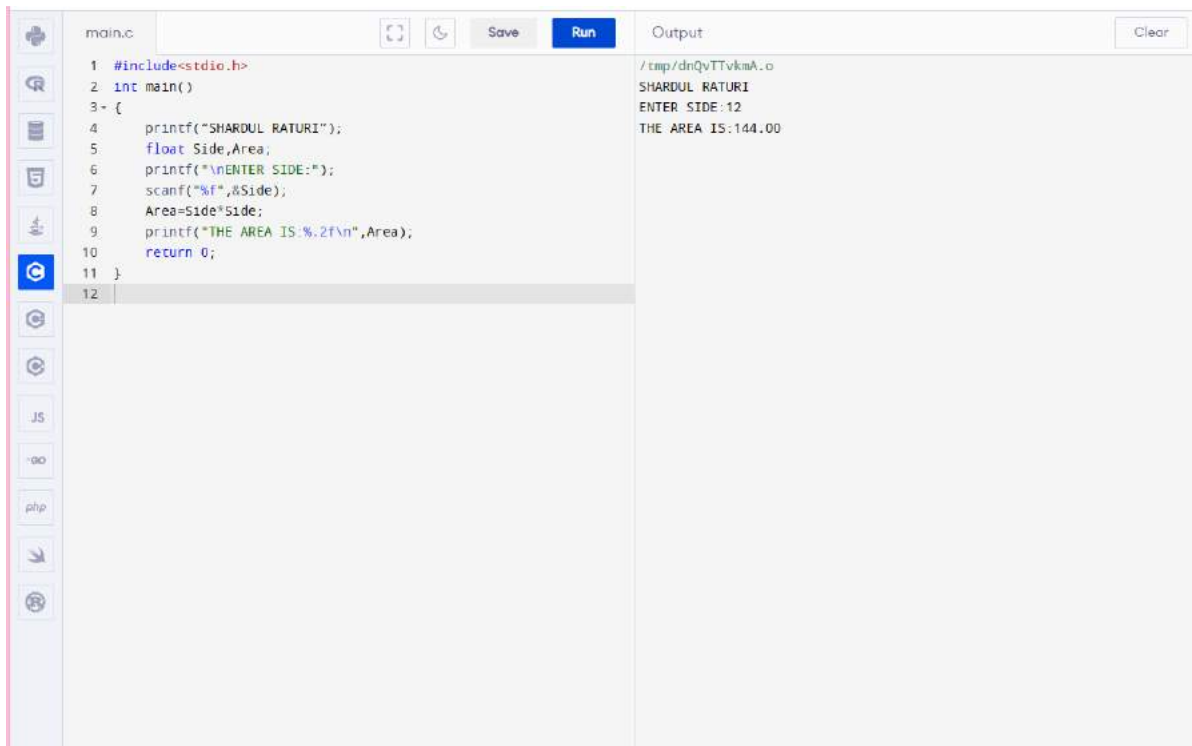
```
    scanf("%f",&Side);
```

```
    Area=Side*Side;
```

```
    printf("THE AREA IS: %.2f\n",Area);
```

```
    return 0;
```

}



The screenshot shows a code editor with a file named 'main.c'. The code is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float Side,Area;
6     printf("\nENTER SIDE:");
7     scanf("%f",&Side);
8     Area=Side*Side;
9     printf("THE AREA IS:%.2f\n",Area);
10    return 0;
11 }
12
```

The 'Output' window on the right shows the following text:

```
/tmp/dnQvTTvkMA.o
SHARDUL RATURI
ENTER SIDE:12
THE AREA IS:144.00
```

❖ PROGRAM 12:- TO FIND AREA OF RIGHT ANGLED TRIANGLE

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    float Base,Height,Area;
```

```
    printf("\nENTER THE VALUE OF BASE:");
```

```
    scanf("%f",&Base);
```

```
    printf("ENTER THE VALUE OF HEIGHT:");
```

```
    scanf("%f",&Height);
```

```

Area=0.5*Base*Height;

printf("THE AREA OF RIGHT ANGLED TRIANGLE IS:%.2f\n",Area);

return 0;
}

```

The screenshot shows a C program in a code editor with the following code:

```

1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float Base,Height,Area;
6     printf("\nENTER THE VALUE OF BASE:");
7     scanf("%f",&Base);
8     printf("ENTER THE VALUE OF HEIGHT:");
9     scanf("%f",&Height);
10    Area=0.5*Base*Height;
11    printf("THE AREA OF RIGHT ANGLED TRIANGLE IS:%.2f\n",Area);
12    return 0;
13 }
14

```

The output window shows the following execution steps:

```

/tmp/dnqvTTvkMA.o
SHARDUL RATURI
ENTER THE VALUE OF BASE:12
ENTER THE VALUE OF HEIGHT:13
THE AREA OF RIGHT ANGLED TRIANGLE IS:78.00

```

❖ PROGRAM 13:- TO FIND AREA OF ISOSCLES TRIANGLE

```

#include<stdio.h>

int main()
{
    printf("SHARDUL RATURI");
}

```

```

float Base,Height,Area;

printf("\nENTER THE VALUE OF BASE:");

scanf("%f",&Base);

printf("ENTER THE VALUE OF HEIGHT:");

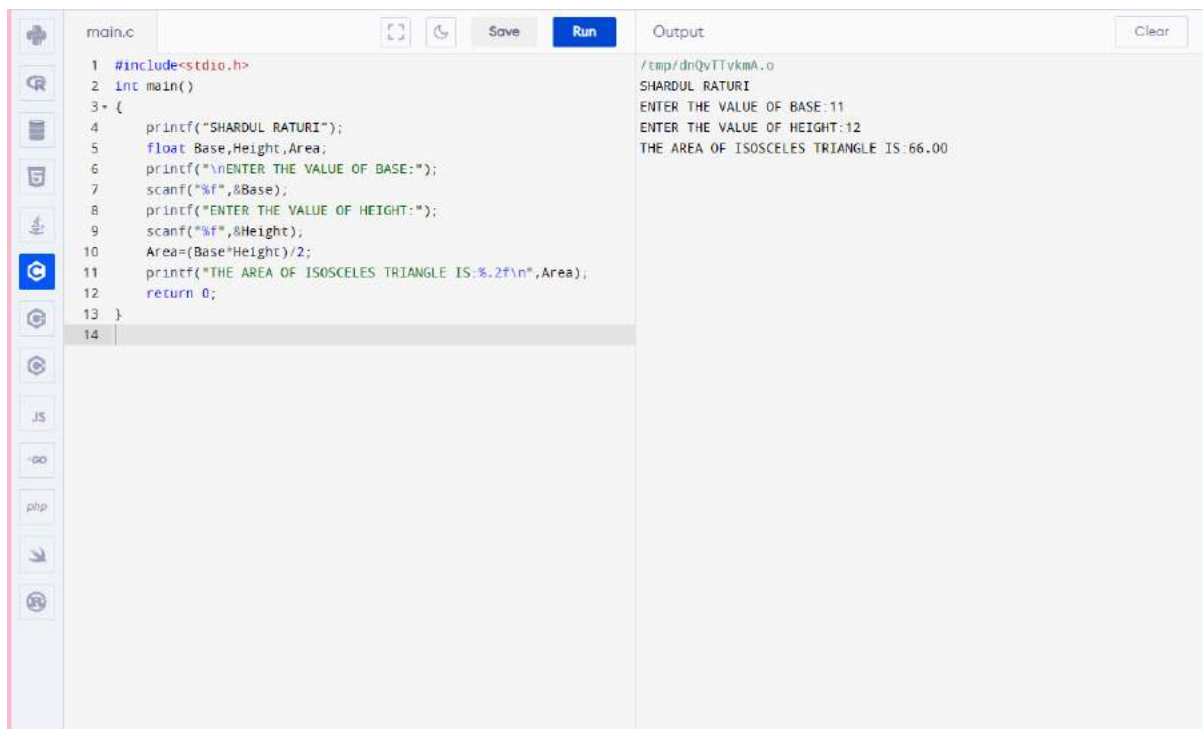
scanf("%f",&Height);

Area=(Base*Height)/2;

printf("THE AREA OF ISOSCELES TRIANGLE IS:%.2f\n",Area);

return 0;
}

```



The screenshot shows a C programming IDE with a file named 'main.c'. The code in the editor is as follows:

```

1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float Base,Height,Area;
6     printf("\nENTER THE VALUE OF BASE:");
7     scanf("%f",&Base);
8     printf("ENTER THE VALUE OF HEIGHT:");
9     scanf("%f",&Height);
10    Area=(Base*Height)/2;
11    printf("THE AREA OF ISOSCELES TRIANGLE IS:%.2f\n",Area);
12    return 0;
13 }
14

```

The IDE has a 'Run' button and a 'Save' button. The output window on the right shows the following text:

```

/tmp/dnQvTTvkMA.o
SHARDUL RATURI
ENTER THE VALUE OF BASE:11
ENTER THE VALUE OF HEIGHT:12
THE AREA OF ISOSCELES TRIANGLE IS:66.00

```

❖ PROGRAM 14:- TO FIND AREA OF TRIANGLE WITH ANY THREE SIDES

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    float side1, side2, side3;
```

```
    printf("\nEnter the length of side 1: ");
```

```
    scanf("%f", &side1);
```

```
    printf("Enter the length of side 2: ");
```

```
    scanf("%f", &side2);
```

```
    printf("Enter the length of side 3: ");
```

```
    scanf("%f", &side3);
```

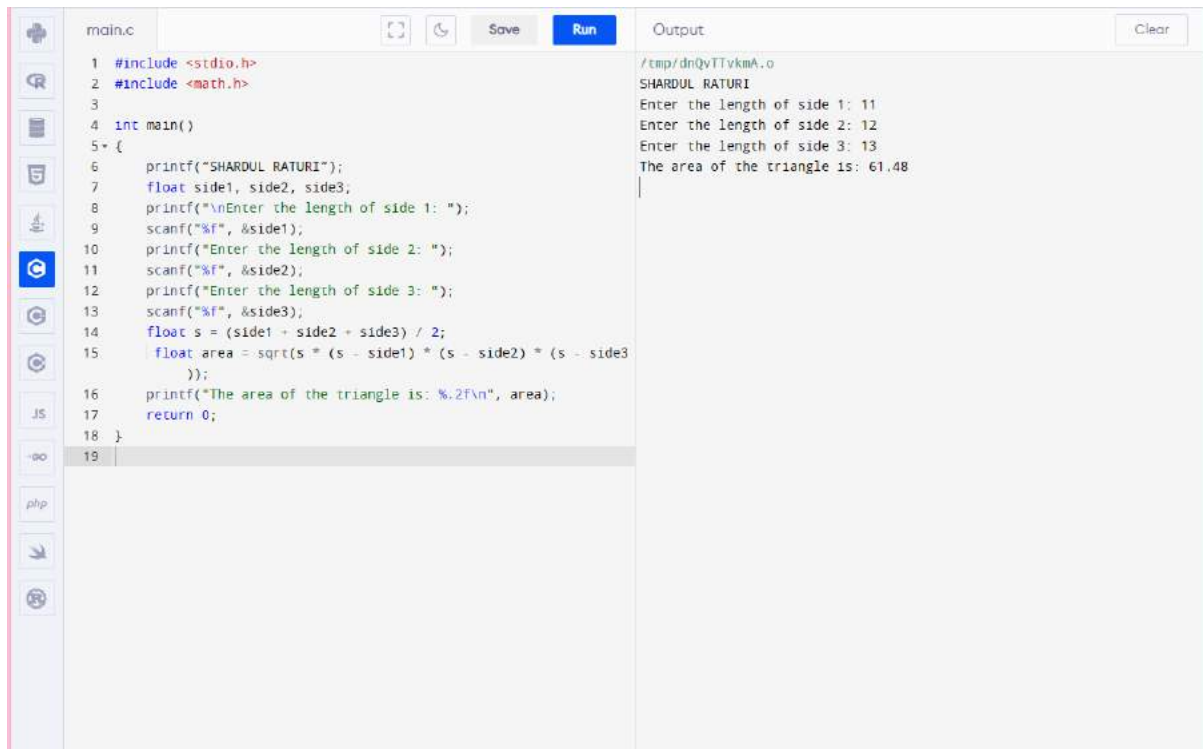
```
    float s = (side1 + side2 + side3) / 2;
```

```
    float area = sqrt(s * (s - side1) * (s - side2) * (s - side3));
```

```
    printf("The area of the triangle is: %.2f\n", area);
```

```
    return 0;
```

```
}
```



The screenshot shows a C++ IDE with a file named 'main.c'. The code is as follows:

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main()
5 {
6     printf("SHARDUL RATURI");
7     float side1, side2, side3;
8     printf("\nEnter the length of side 1: ");
9     scanf("%f", &side1);
10    printf("Enter the length of side 2: ");
11    scanf("%f", &side2);
12    printf("Enter the length of side 3: ");
13    scanf("%f", &side3);
14    float s = (side1 + side2 + side3) / 2;
15    float area = sqrt(s * (s - side1) * (s - side2) * (s - side3));
16    printf("The area of the triangle is: %.2f\n", area);
17    return 0;
18 }
```

The output window on the right shows the following text:

```
/tmp/dnQyT7ykmA.o
SHARDUL RATURI
Enter the length of side 1: 11
Enter the length of side 2: 12
Enter the length of side 3: 13
The area of the triangle is: 61.48
```

PROGRAM 15:- TO FIND AREA AND VOLUME OF CUBE

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    float Side,Area,Volume;
```

```
    printf("\nENTER SIDE OF A CUBE:");
```

```
    scanf("%f",&Side);
```

```
    Area=6*Side*Side;
```

```
    Volume=Side*Side*Side;
```

```

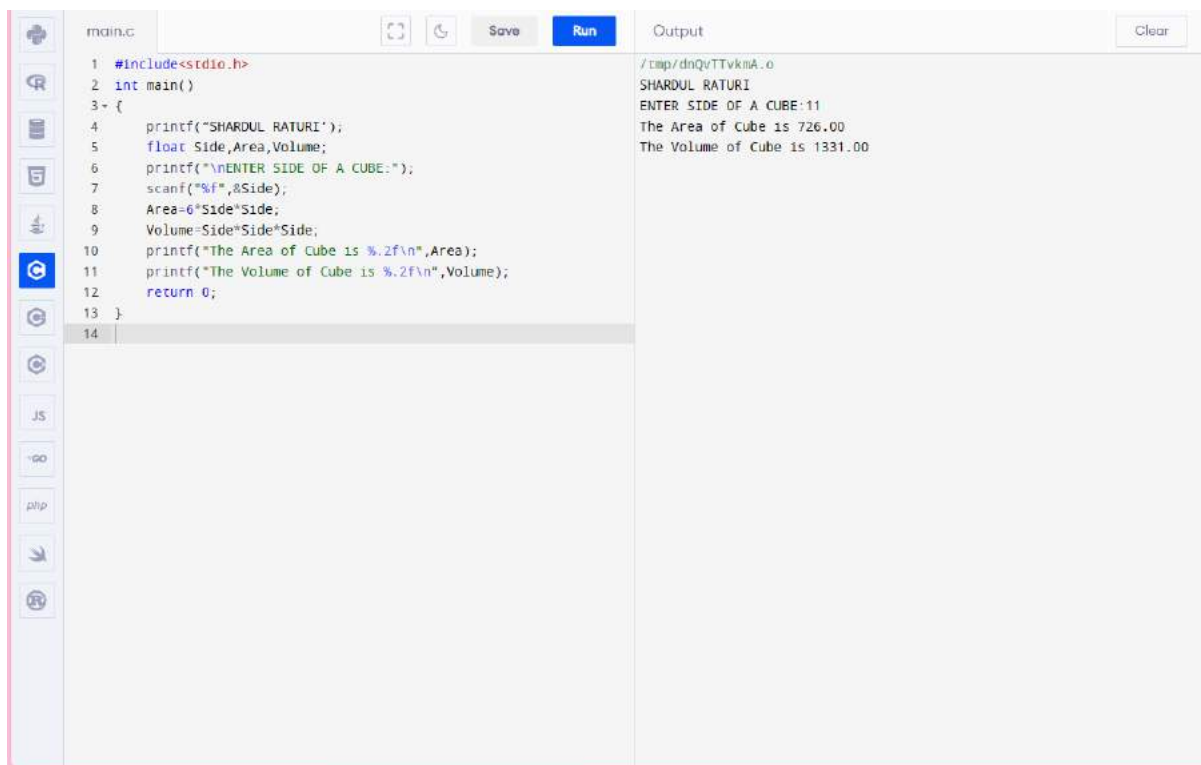
printf("The Area of Cube is %.2f\n",Area);

printf("The Volume of Cube is %.2f\n",Volume);

return 0;

}

```



The screenshot shows a C++ IDE with a file named `main.cpp`. The code in the editor is as follows:

```

1  #include<stdio.h>
2  int main()
3  {
4      printf("SHARDUL RATURI");
5      float Side,Area,Volume;
6      printf("\nENTER SIDE OF A CUBE:");
7      scanf("%f",&Side);
8      Area=6*Side*Side;
9      Volume=Side*Side*Side;
10     printf("The Area of Cube is %.2f\n",Area);
11     printf("The Volume of Cube is %.2f\n",Volume);
12     return 0;
13 }
14

```

The Output window on the right shows the following text:

```

/tmp/dnQvTTvknA.o
SHARDUL RATURI
ENTER SIDE OF A CUBE:11
The Area of Cube is 726.00
The Volume of Cube is 1331.00

```

❖ PROGRAM 16: -TO FIND AREA AND VOLUME OF CUBOID

```

#include<stdio.h>

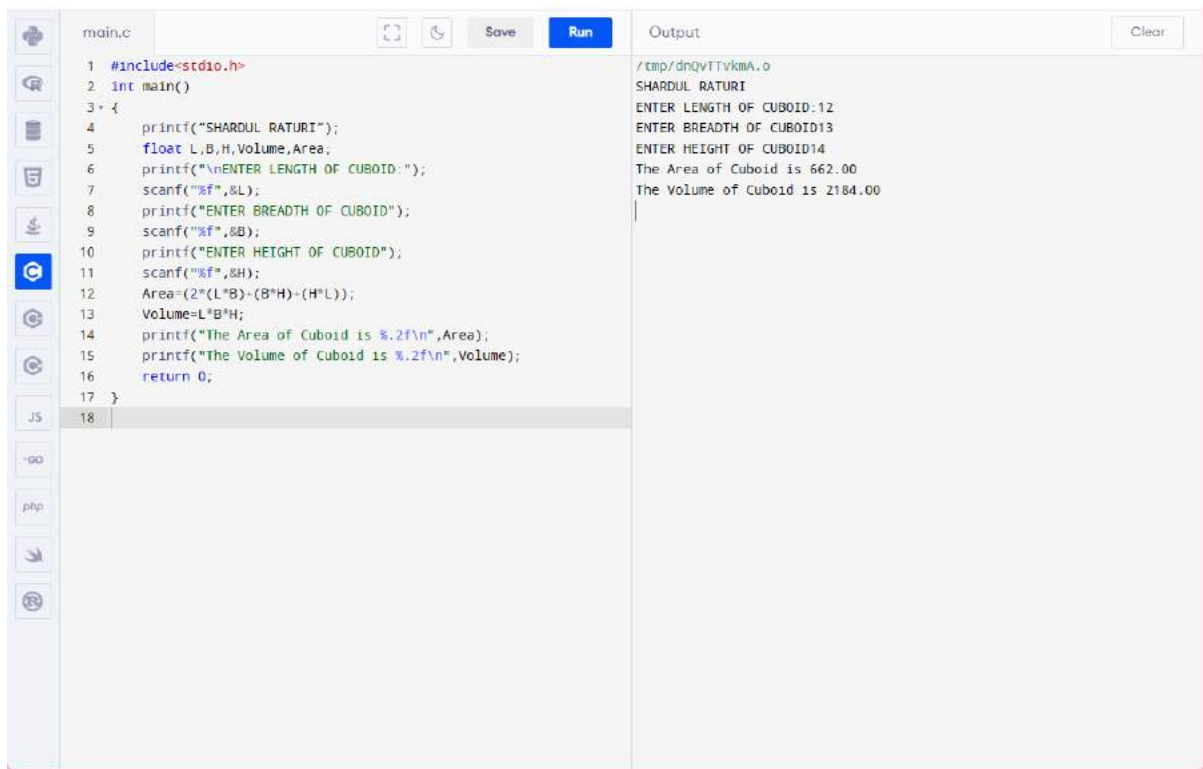
int main()

{

```



```
printf("SHARDUL RATURI");  
  
float L,B,H,Volume,Area;  
  
printf("\nENTER LENGTH OF CUBOID:");  
  
scanf("%f",&L);  
  
printf("ENTER BREADTH OF CUBOID");  
  
scanf("%f",&B);  
  
printf("ENTER HEIGHT OF CUBOID");  
  
scanf("%f",&H);  
  
Area=(2*(L*B)+(B*H)+(H*L));  
  
Volume=L*B*H;  
  
printf("The Area of Cuboid is %.2f\n",Area);  
  
printf("The Volume of Cuboid is %.2f\n",Volume);  
  
return 0;  
  
}
```



```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     float L,B,H,Volume,Area;
6     printf("\nEnter LENGTH OF CUBOID:");
7     scanf("%f",&L);
8     printf("ENTER BREADTH OF CUBOID");
9     scanf("%f",&B);
10    printf("ENTER HEIGHT OF CUBOID");
11    scanf("%f",&H);
12    Area=(2*(L*B)+(B*H)+(H*L));
13    Volume=L*B*H;
14    printf("The Area of Cuboid is %.2f\n",Area);
15    printf("The Volume of Cuboid is %.2f\n",Volume);
16    return 0;
17 }
18
```

Output

```
/tmp/dnQvTtVkmA.o
SHARDUL RATURI
ENTER LENGTH OF CUBOID:12
ENTER BREADTH OF CUBOID:13
ENTER HEIGHT OF CUBOID:14
The Area of Cuboid is 662.00
The Volume of Cuboid is 2184.00
```

❖ PROGRAM 17: - TO FIND LARGEST NUMBER USING LOGICAL AND OPERATOR

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int num1, num2;
```

```
    printf("\nEnter the first number: ");
```

```
    scanf("%d", &num1);
```

```
    printf("Enter the second number: ");
```

```

scanf("%d", &num2);

if (num1 > num2 && num1 != num2) {
    printf("The largest number is: %d\n", num1);
} else if (num2 > num1 && num1 != num2) {
    printf("The largest number is: %d\n", num2);
} else {
    printf("Both numbers are equal.\n");
}

return 0;
}

```

The screenshot shows a C programming IDE with a file named 'main.c'. The code in the editor is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num1, num2;
6     printf("\nEnter the first number: ");
7     scanf("%d", &num1);
8     printf("Enter the second number: ");
9     scanf("%d", &num2);
10    if (num1 > num2 && num1 != num2) {
11        printf("The largest number is: %d\n", num1);
12    } else if (num2 > num1 && num1 != num2) {
13        printf("The largest number is: %d\n", num2);
14    } else {
15        printf("Both numbers are equal.\n");
16    }
17    return 0;
18 }
19

```

The IDE has a 'Run' button and a 'Clear' button. The output window on the right shows the following text:

```

/tmp/dnQvTTvkmA.o
SHARDUL RATURI
Enter the first number: 12
Enter the second number: 13
The largest number is: 13

```

❖ PROGRAM 18: - TO VALIDATE THE USERNAME AND PASSWORD ENTERED BY USER IS CORRECT OR NOT USING THE PREDEFINED USERNAME AND PASSWORD

```
#include <stdio.h>

#include <string.h>

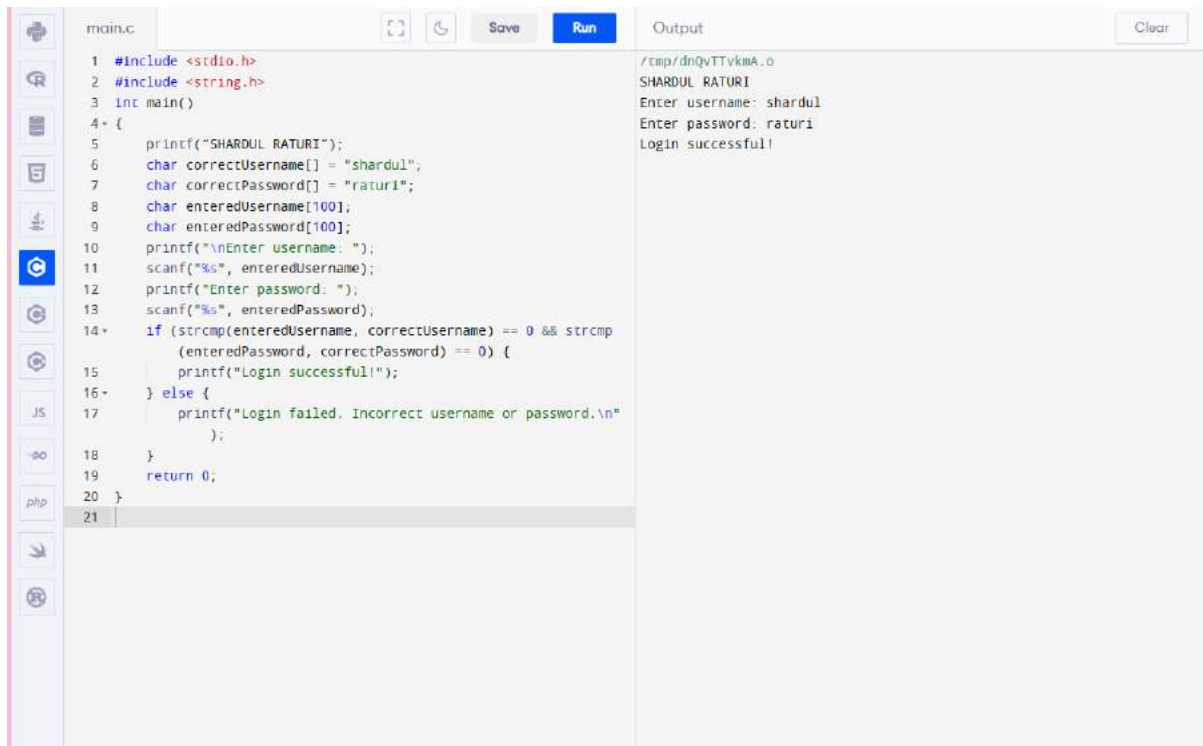
int main()
{
    printf("SHARDUL RATURI");
    char correctUsername[] = "HIMANSHUMAWDI";
    char correctPassword[] = "MAWDIHIMANSHU";
    char enteredUsername[100];
    char enteredPassword[100];
    printf("\nEnter username: ");
    scanf("%s", enteredUsername);
    printf("Enter password: ");
    scanf("%s", enteredPassword);
    if (strcmp(enteredUsername, correctUsername) == 0 &&
    strcmp(enteredPassword, correctPassword) == 0) {
        printf("Login successful!");
    } else {
```

```

        printf("Login failed. Incorrect username or password.\n");
    }

    return 0;
}

```



The screenshot shows a C++ IDE with a file named 'main.cpp'. The code implements a login system. It includes `<stdio.h>` and `<string.h>`. In the `main` function, it prints 'SHARDUL RATURI'. It then prompts the user for a username and password. The correct credentials are 'shardul' and 'raturi'. The program checks if the entered credentials match the correct ones using `strcmp`. If they match, it prints 'Login successful!'. Otherwise, it prints 'Login failed. Incorrect username or password.\n'. The output window on the right shows the execution results: the program name, the user's name, the entered username and password, and the 'Login successful!' message.

```

main.cpp
1 #include <stdio.h>
2 #include <string.h>
3 int main()
4 {
5     printf("SHARDUL RATURI");
6     char correctUsername[] = "shardul";
7     char correctPassword[] = "raturi";
8     char enteredUsername[100];
9     char enteredPassword[100];
10    printf("\nEnter username: ");
11    scanf("%s", enteredUsername);
12    printf("Enter password: ");
13    scanf("%s", enteredPassword);
14    if (strcmp(enteredUsername, correctUsername) == 0 && strcmp
        (enteredPassword, correctPassword) == 0) {
15        printf("Login successful!");
16    } else {
17        printf("Login failed. Incorrect username or password.\n"
            );
18    }
19    return 0;
20 }
21
Output
/cmp/dnQvTTvkMA.o
SHARDUL RATURI
Enter username: shardul
Enter password: raturi
Login successful!

```

❖ PROGRAM 19: - TO INPUT A POSITIVE NUMBER FROM THE USER AND PERFORM THE LEFT SHIFT OPERATOR

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
printf("SHARDUL RATURI");

int number, shift, result;

printf("\nEnter a number: ");

scanf("%d", &number);

printf("Enter the number of positions to shift left: ");

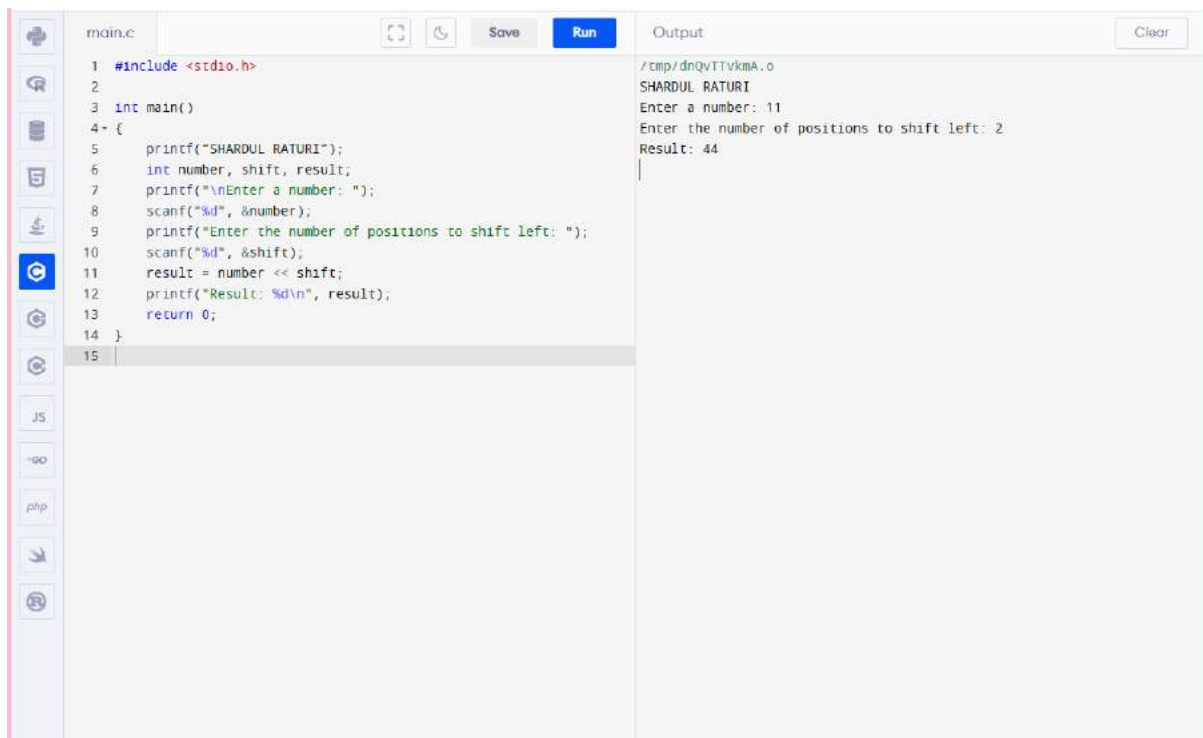
scanf("%d", &shift);

result = number << shift;

printf("Result: %d\n", result);

return 0;

}
```



The screenshot shows a code editor with a file named 'main.c'. The code is a C program that prints 'SHARDUL RATURI', prompts for a number and a shift value, and then prints the result of a left shift operation. The code is as follows:

```
1 #include <stdio.h>
2
3 int main()
4 {
5     printf("SHARDUL RATURI");
6     int number, shift, result;
7     printf("\nEnter a number: ");
8     scanf("%d", &number);
9     printf("Enter the number of positions to shift left: ");
10    scanf("%d", &shift);
11    result = number << shift;
12    printf("Result: %d\n", result);
13    return 0;
14 }
15
```

The IDE has a 'Run' button and a 'Clear' button. The output window shows the following text:

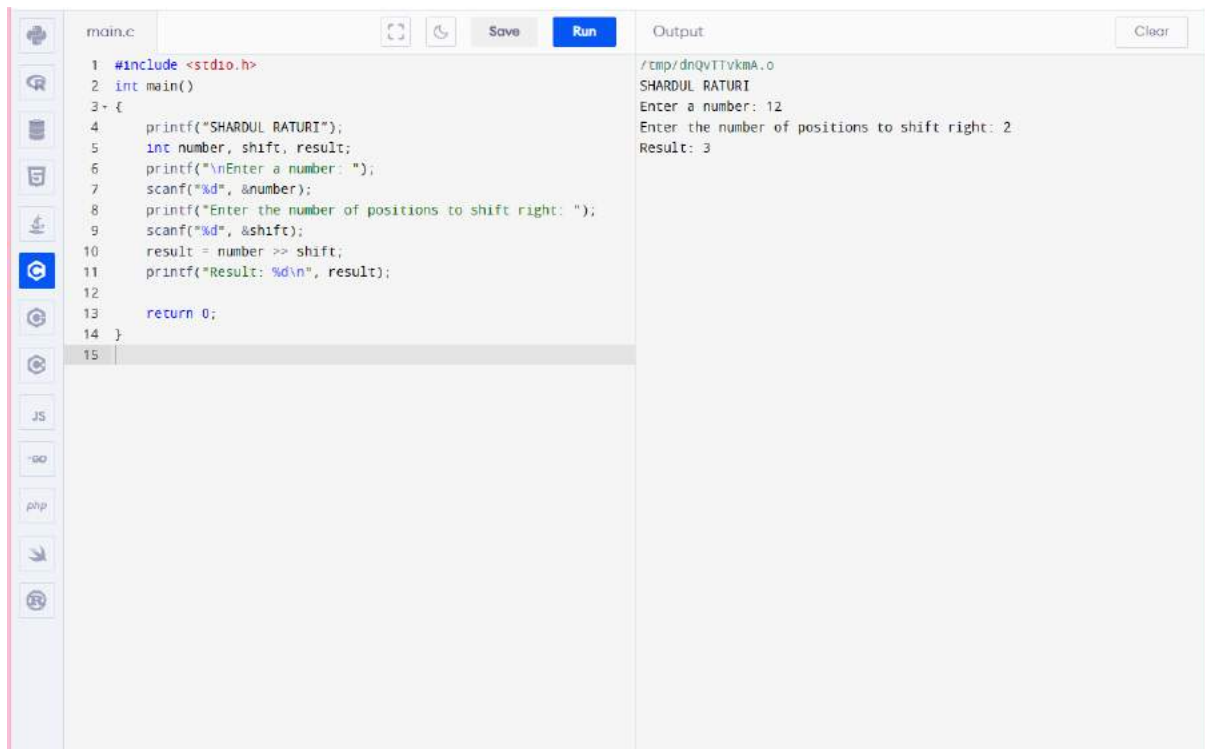
```
/tmp/dnQvTTvKmA.o
SHARDUL RATURI
Enter a number: 11
Enter the number of positions to shift left: 2
Result: 44
```

❖ PROGRAM 20: - TO INPUT THE POSITIVE NUMBER FROM THE USER TO PERFORM THE RIGHT SHIFT OPERATOR

```
#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");
    int number, shift, result;
    printf("\nEnter a number: ");
    scanf("%d", &number);
    printf("Enter the number of positions to shift right: ");
    scanf("%d", &shift);
    result = number >> shift;
    printf("Result: %d\n", result);

    return 0;
}
```



The screenshot shows a code editor with a file named 'main.c'. The code is a C program that prints 'SHARDUL RATURI', prompts for a number and a shift value, and then prints the result of a right shift operation. The output window on the right shows the program's execution: it prints 'SHARDUL RATURI', prompts for a number (12) and a shift value (2), and then prints 'Result: 3'.

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int number, shift, result;
6     printf("\nEnter a number: ");
7     scanf("%d", &number);
8     printf("Enter the number of positions to shift right: ");
9     scanf("%d", &shift);
10    result = number >> shift;
11    printf("Result: %d\n", result);
12
13    return 0;
14 }
```

Output:

```
/tmp/dnQvTtVkmA.o
SHARDUL RATURI
Enter a number: 12
Enter the number of positions to shift right: 2
Result: 3
```

❖ PROGRAM 21: - TO PERFORM THE PRE-INCREMENT AND PRE-DECREMENT OPERATOR ON TWO INTEGER AND PRINT BOTH ORIGINAL VALUE AND UPDATE VALUE

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int num1, num2;
```

```
    printf("\nEnter the first integer: ");
```



```

scanf("%d", &num1);

printf("Enter the second integer: ");

scanf("%d", &num2);

int updatedNum1 = ++num1;

int updatedNum2 = --num2;

printf("Original num1: %d, Updated num1: %d\n", num1 - 1,
updatedNum1);

printf("Original num2: %d, Updated num2: %d\n", num2 + 1,
updatedNum2);

return 0;
}

```

The screenshot shows a C programming IDE with a file named `main.c`. The code in the editor is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num1, num2;
6     printf("\nEnter the first integer: ");
7     scanf("%d", &num1);
8     printf("Enter the second integer: ");
9     scanf("%d", &num2);
10    int updatedNum1 = ++num1;
11    int updatedNum2 = --num2;
12    printf("Original num1: %d, Updated num1: %d\n", num1 - 1,
        updatedNum1);
13    printf("Original num2: %d, Updated num2: %d\n", num2 + 1,
        updatedNum2);
14    return 0;
15 }
16
17

```

The IDE has a `Run` button, which has been clicked. The output window on the right shows the following text:

```

/cmp/dnQvTTvkmA.o
SHARDUL RATURI
Enter the first integer: 12
Enter the second integer: 23
Original num1: 12, Updated num1: 13
Original num2: 23, Updated num2: 22

```

❖ PROGRAM 22: - TO PERFORM THE POST-INCREMENT AND POST-DECREMENT OPERATOR ON TWO INTEGER AND PRINT BOTH ORIGINAL VALUE AND UPDATE VALUE

```
#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");

    int num1, num2;

    printf("\nEnter the first integer: ");
    scanf("%d", &num1);

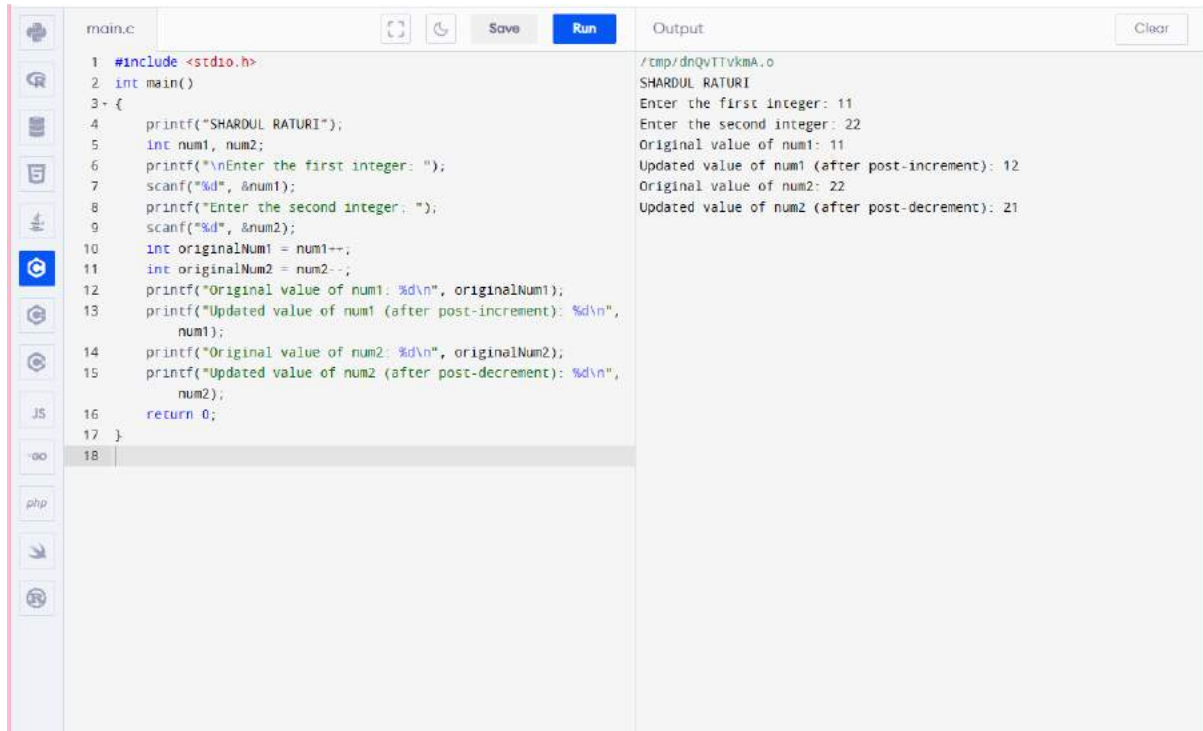
    printf("Enter the second integer: ");
    scanf("%d", &num2);

    int originalNum1 = num1++;
    int originalNum2 = num2--;

    printf("Original value of num1: %d\n", originalNum1);
    printf("Updated value of num1 (after post-increment): %d\n",
num1);

    printf("Original value of num2: %d\n", originalNum2);
    printf("Updated value of num2 (after post-decrement): %d\n",
num2);
```

```
    return 0;
}
```



```
main.c
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num1, num2;
6     printf("\nEnter the first integer: ");
7     scanf("%d", &num1);
8     printf("Enter the second integer: ");
9     scanf("%d", &num2);
10    int originalNum1 = num1++;
11    int originalNum2 = num2--;
12    printf("Original value of num1: %d\n", originalNum1);
13    printf("Updated value of num1 (after post-increment): %d\n",
14           num1);
15    printf("Original value of num2: %d\n", originalNum2);
16    printf("Updated value of num2 (after post-decrement): %d\n",
17           num2);
18    return 0;
19 }
```

Output

```
/tmp/dnQvTtVkmA.o
SHARDUL RATURI
Enter the first integer: 11
Enter the second integer: 22
Original value of num1: 11
Updated value of num1 (after post-increment): 12
Original value of num2: 22
Updated value of num2 (after post-decrement): 21
```

❖ PROGRAM 23: - TO CHECK WHETHER FOR AN INTEGER NUMBER THAT IT IS DIVISIBLE BY 9 OR 7 USING OR OPERATOR.

```
#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");
```

```
int number;

printf("\nEnter an integer: ");

scanf("%d", &number);

if (number % 9 == 0 || number % 7 == 0) {

    printf("%d is divisible by either 9 or 7.\n", number);

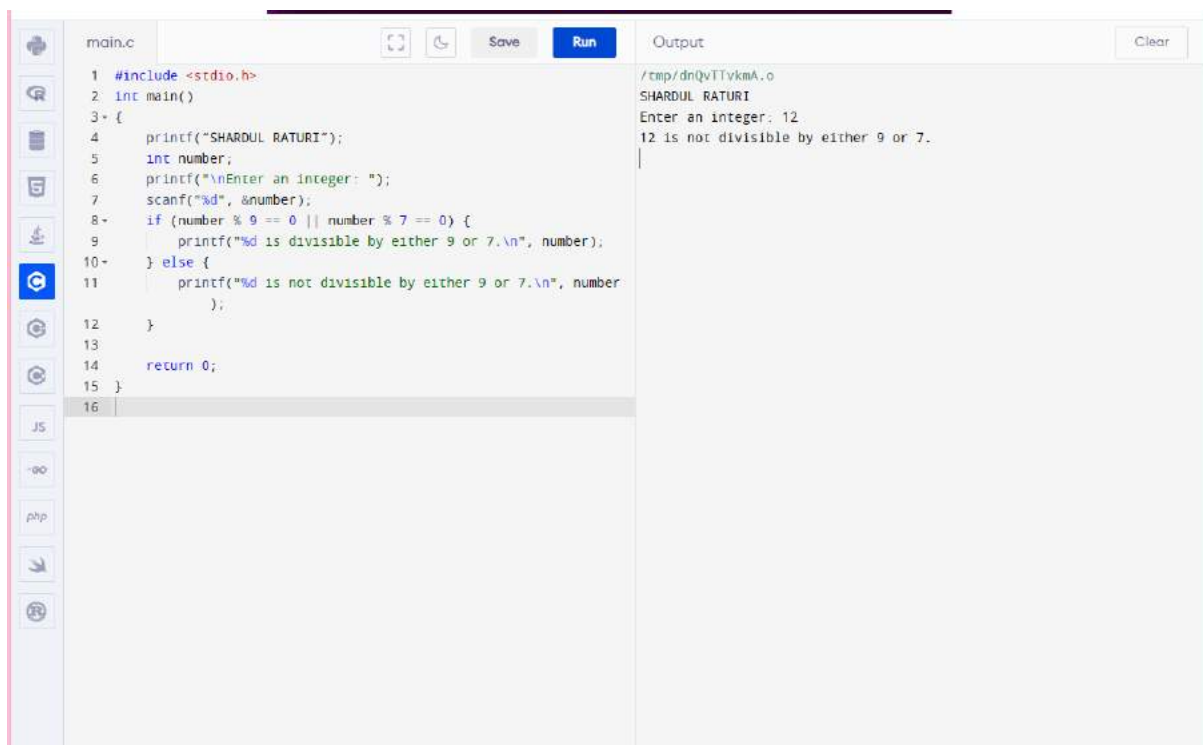
} else {

    printf("%d is not divisible by either 9 or 7.\n", number);

}

return 0;

}
```



The screenshot shows a code editor with a file named 'main.c'. The code is a C program that prompts the user to enter an integer and checks if it is divisible by 9 or 7. The code is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int number;
6     printf("\nEnter an integer: ");
7     scanf("%d", &number);
8     if (number % 9 == 0 || number % 7 == 0) {
9         printf("%d is divisible by either 9 or 7.\n", number);
10    } else {
11        printf("%d is not divisible by either 9 or 7.\n", number);
12    }
13
14    return 0;
15 }
16
```

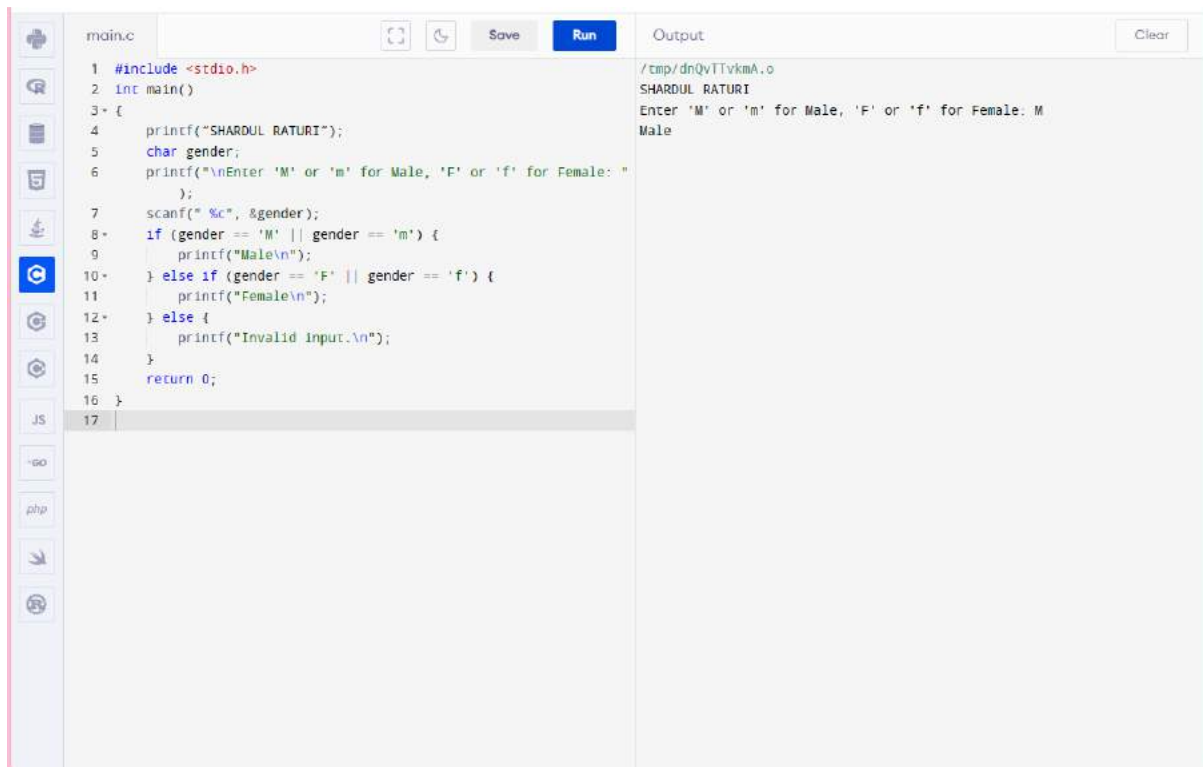
The output window on the right shows the following text:

```
/tmp/dnQvTIVkmA.o
SHARDUL RATURI
Enter an integer: 12
12 is not divisible by either 9 or 7.
```

❖ **PROGRAM 24: - TO IDENTIFY GENDER
IN SINGLE CHARACTER AND PRINT FULL
GENDER(EX: IF INPUT IS 'M' OR 'm'-IT
SHOULD PRINT "MALE"**

```
#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");
    char gender;
    printf("\nEnter 'M' or 'm' for Male, 'F' or 'f' for Female: ");
    scanf(" %c", &gender);
    if (gender == 'M' || gender == 'm') {
        printf("Male\n");
    } else if (gender == 'F' || gender == 'f') {
        printf("Female\n");
    } else {
        printf("Invalid input.\n");
    }
    return 0;
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     char gender;
6     printf("\nEnter 'M' or 'm' for Male, 'F' or 'f' for Female: ");
7     scanf("%c", &gender);
8     if (gender == 'M' || gender == 'm') {
9         printf("Male\n");
10    } else if (gender == 'F' || gender == 'f') {
11        printf("Female\n");
12    } else {
13        printf("Invalid input.\n");
14    }
15    return 0;
16 }
17
```

Output

```
/tmp/dnQvTtykmA.o
SHARDUL RATURI
Enter 'M' or 'm' for Male, 'F' or 'f' for Female: M
Male
```

❖ PROGRAM 25:- TO PRINT ALL NATURAL NUMBERS IN REVERSE (FROM n TO 1)

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int n;
```

```
    printf("\nEnter the value of N:");
```

```
    scanf("%d",&n);
```

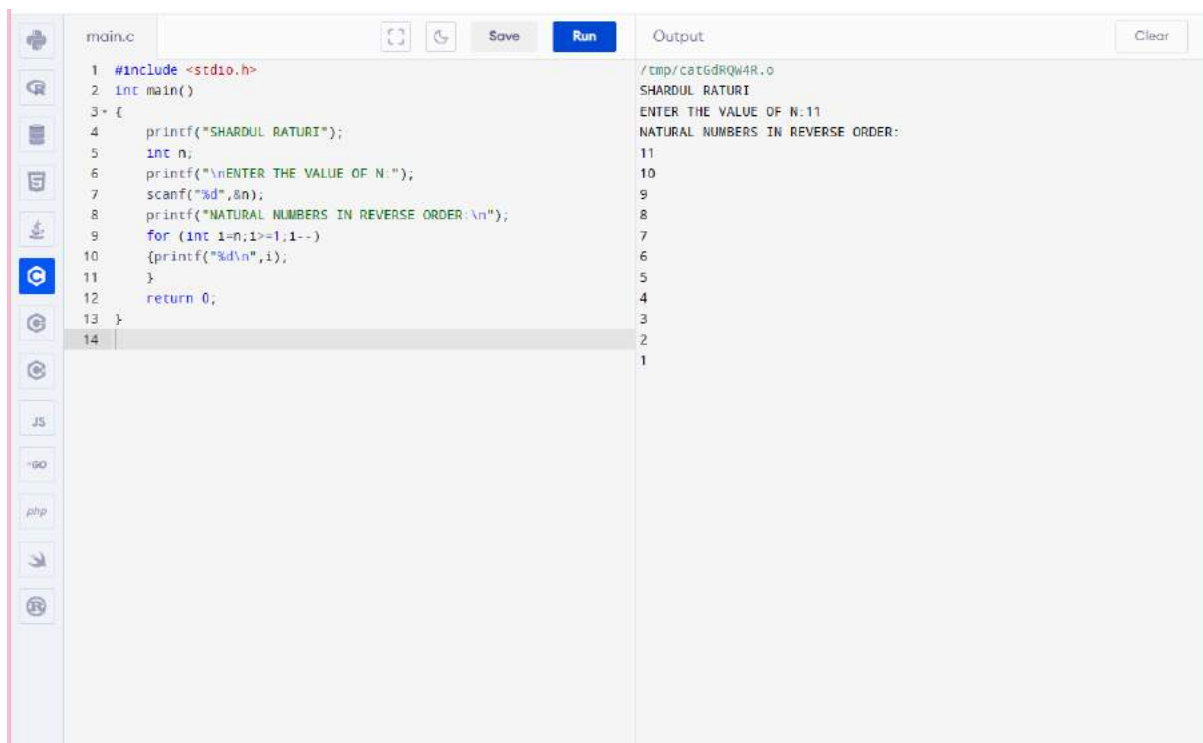
```
    printf("Natural numbers in reverse order:\n");
```

```

for (int i=n;i>=1;i--)
{printf("%d\n",i);
}

return 0;
}

```



The screenshot shows a C++ IDE with a file named 'main.cpp'. The code in the editor is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int n;
6     printf("\nENTER THE VALUE OF N:");
7     scanf("%d",&n);
8     printf("NATURAL NUMBERS IN REVERSE ORDER:\n");
9     for (int i=n;i>=1;i--)
10     {printf("%d\n",i);
11     }
12     return 0;
13 }
14

```

The output window on the right shows the following text:

```

/tmp/cc1GdRQW4R.o
SHARDUL RATURI
ENTER THE VALUE OF N:11
NATURAL NUMBERS IN REVERSE ORDER:
11
10
9
8
7
6
5
4
3
2
1

```

❖ PROGRAM 26:- TO PRINT ALL ALPHABET FROM a TO z

```

#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");
    char a;
}

```

```

printf("\nALPHABETS FROM a TO z :\n");

for (a='a';a<='z';a++)

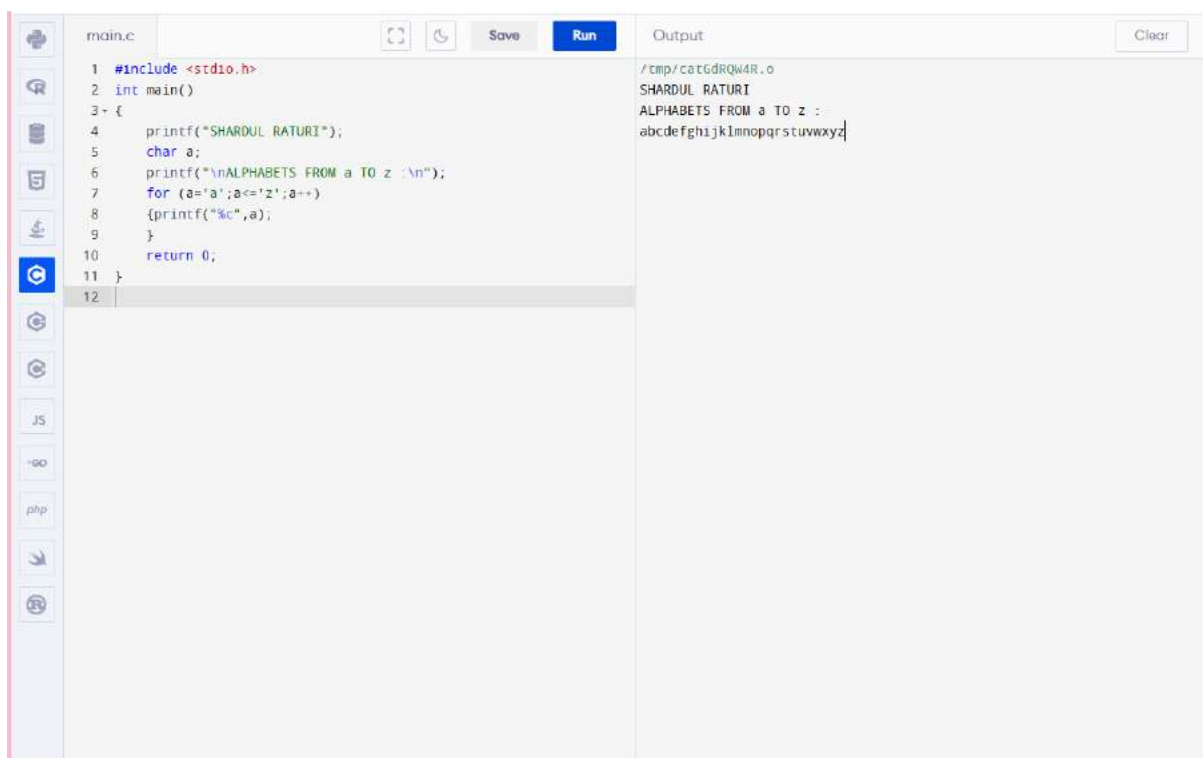
{printf("%c",a);

}

return 0;

}

```



```

main.c
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     char a;
6     printf("\nALPHABETS FROM a TO z :\n");
7     for (a='a';a<='z';a++)
8     {printf("%c",a);
9     }
10    return 0;
11 }
12
Output
/tmp/catGdRQW4R.o
SHARDUL RATURI
ALPHABETS FROM a TO z :
abcdefghijklmnopqrstuvwxyz

```

❖ PROGRAM 27:- TO PRINT ALL NATURAL NUMBERS FROM 1 TO n

```

#include <stdio.h>

int main()

{

    printf("SHARDUL RATURI");

```



```

int n;

printf("\nENTER THE VALUE OF N:");

scanf("%d",&n);

printf("NATURAL NUMBERS ARE:\n",n);

for (int i=1;i<=n;i++)

{printf("%d\n",i);

}

return 0;

}

```

The screenshot shows a C++ IDE with a file named 'main.c'. The code in the editor is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int n;
6     printf("\nENTER THE VALUE OF N:");
7     scanf("%d",&n);
8     printf("NATURAL NUMBERS ARE:\n",n);
9     for (int i=1;i<=n;i++)
10     {printf("%d\n",i);
11     }
12     return 0;
13 }
14

```

The IDE has a 'Run' button highlighted in blue. To the right of the code editor is an 'Output' window showing the program's execution:

```

/tmp/catGdRQW4R.o
SHARDUL RATURI
ENTER THE VALUE OF N:22
NATURAL NUMBERS ARE:
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22

```

❖ PROGRAM 28:- TO PRINT ALL EVEN NUMBERS BETWEEN 1 TO 100

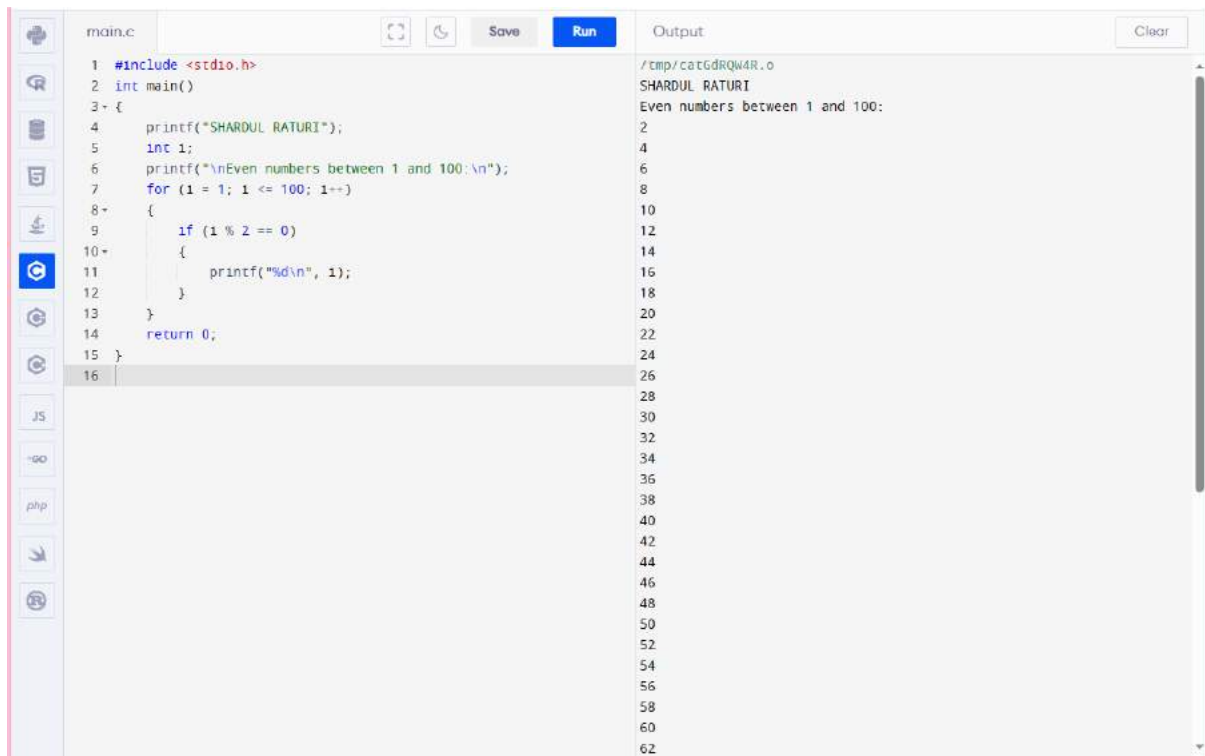
```
#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");

    int i;

    printf("\nEven numbers between 1 and 100:\n");
    for (i = 1; i <= 100; i++)
    {
        if (i % 2 == 0)
        {
            printf("%d\n", i);
        }
    }

    return 0;
}
```



The screenshot shows a C program in a code editor. The code is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int i;
6     printf("\nEven numbers between 1 and 100:\n");
7     for (i = 1; i <= 100; i++)
8     {
9         if (i % 2 == 0)
10        {
11            printf("%d\n", i);
12        }
13    }
14    return 0;
15 }
```

The output of the program is displayed on the right side of the editor:

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Even numbers between 1 and 100:
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
```

PROGRAM 29:- TO PRINT ALL ODD NUMBER BETWEEN 1 TO 100

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int i;
```

```
    printf("\nEven numbers between 1 and 100:\n");
```

```
    for (i = 1; i <= 100; i++)
```

```
    {
```

```
        if (i % 2 != 0)
```

```
        {
```

```
            printf("%d\n", i);
```

```

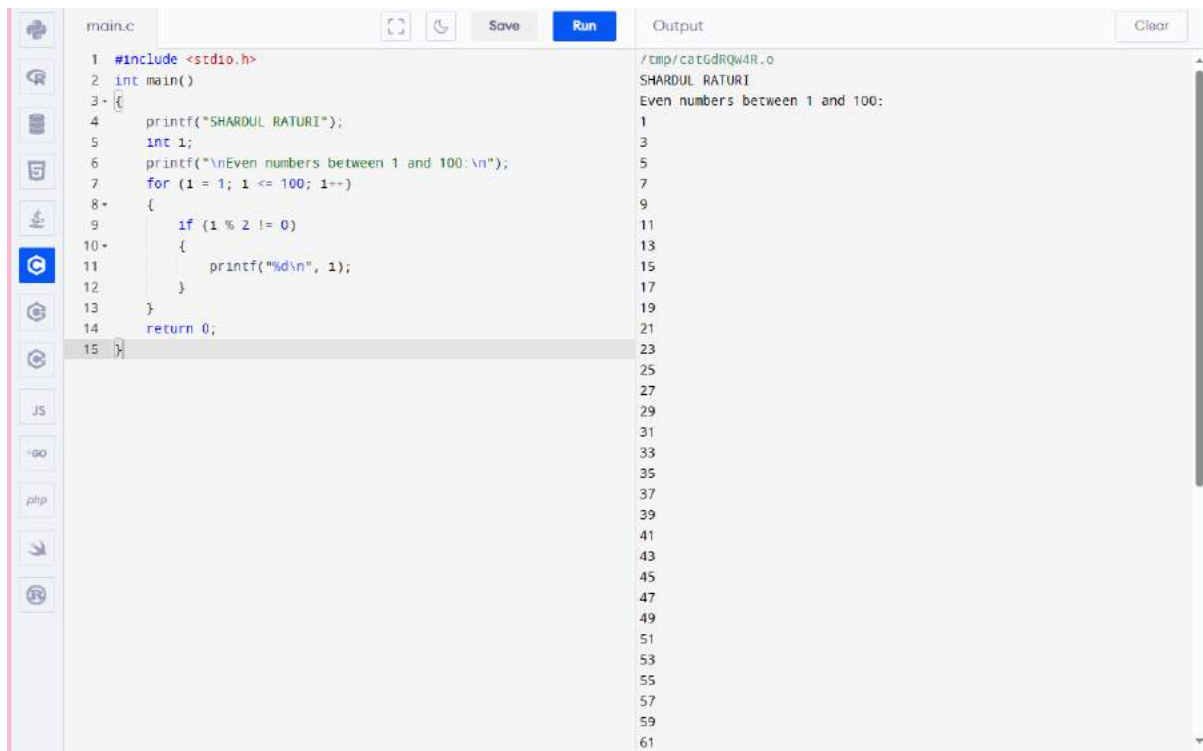
    }

}

return 0;

}

```



```

main.c
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int i;
6     printf("\nEven numbers between 1 and 100:\n");
7     for (i = 1; i <= 100; i++)
8     {
9         if (i % 2 != 0)
10        {
11            printf("%d\n", i);
12        }
13    }
14    return 0;
15 }

```

Output

```

/tmp/catGdRQW4R.o
SHARDUL RATURI
Even numbers between 1 and 100:
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
64
66
68
70
72
74
76
78
80
82
84
86
88
90
92
94
96
98
100

```

❖ PROGRAM 30:- TO FIND SUM OF ALL NATURAL NUMBERS BETWEEN 1 TO N

```

#include<stdio.h>

int main()
{
    printf("SHARDUL RATURI");
}

```

```
int n;

long sum = 0;

printf("\nEnter a value of n: ");

scanf("%d", &n);

if (n < 0) {

    printf("Please enter a positive integer.\n");

}

else {

    for (int i = 1; i <= n; i++)

    {

        sum += i;

    }

    printf("Sum of natural numbers from 1 to %d is %d.\n", n,
sum);

}

return 0;

}
```

The screenshot shows a C++ IDE with a file named 'main.c'. The code is as follows:

```
1 #include<stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int n;
6     long sum = 0;
7     printf("\nEnter a value of n: ");
8     scanf("%d", &n);
9     if (n < 0) {
10         printf("Please enter a positive integer.\n");
11     }
12     else {
13         for (int i = 1; i <= n; i++)
14         {
15             sum += i;
16         }
17         printf("Sum of natural numbers from 1 to %d is %d.\n", n, sum);
18     }
19     return 0;
20 }
21
22
```

The output window on the right shows the following text:

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter a value of n: 11
Sum of natural numbers from 1 to 11 is 66.
```

❖ PROGRAM 31:- TO FIND SUM AF ALL EVEN NUMBER BETWEEN 1 TO N

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int n, sum = 0;
```

```
    printf("\nEnter value of n: ");
```

```
    scanf("%d", &n);
```

```
    if (n < 1)
```

```
    {
```

```
        printf("Invailid\n");
```

```
        return 0;
```

```

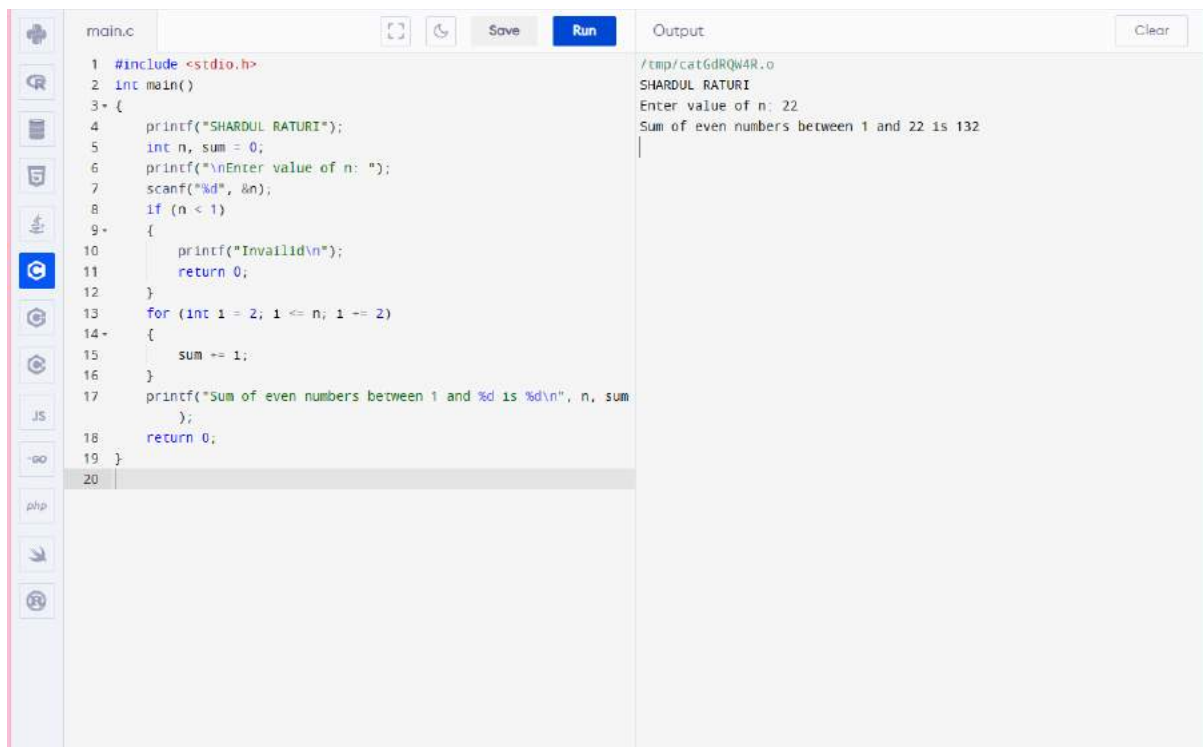
    }

    for (int i = 2; i <= n; i += 2)
    {
        sum += i;
    }

    printf("Sum of even numbers between 1 and %d is %d\n", n,
sum);

    return 0;
}

```



The screenshot shows a C++ IDE with a file named `main.cpp`. The code is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int n, sum = 0;
6     printf("\nEnter value of n: ");
7     scanf("%d", &n);
8     if (n < 1)
9     {
10        printf("Invalid\n");
11        return 0;
12    }
13    for (int i = 2; i <= n; i += 2)
14    {
15        sum += i;
16    }
17    printf("Sum of even numbers between 1 and %d is %d\n", n, sum);
18    return 0;
19 }
20

```

The Output window shows the following text:

```

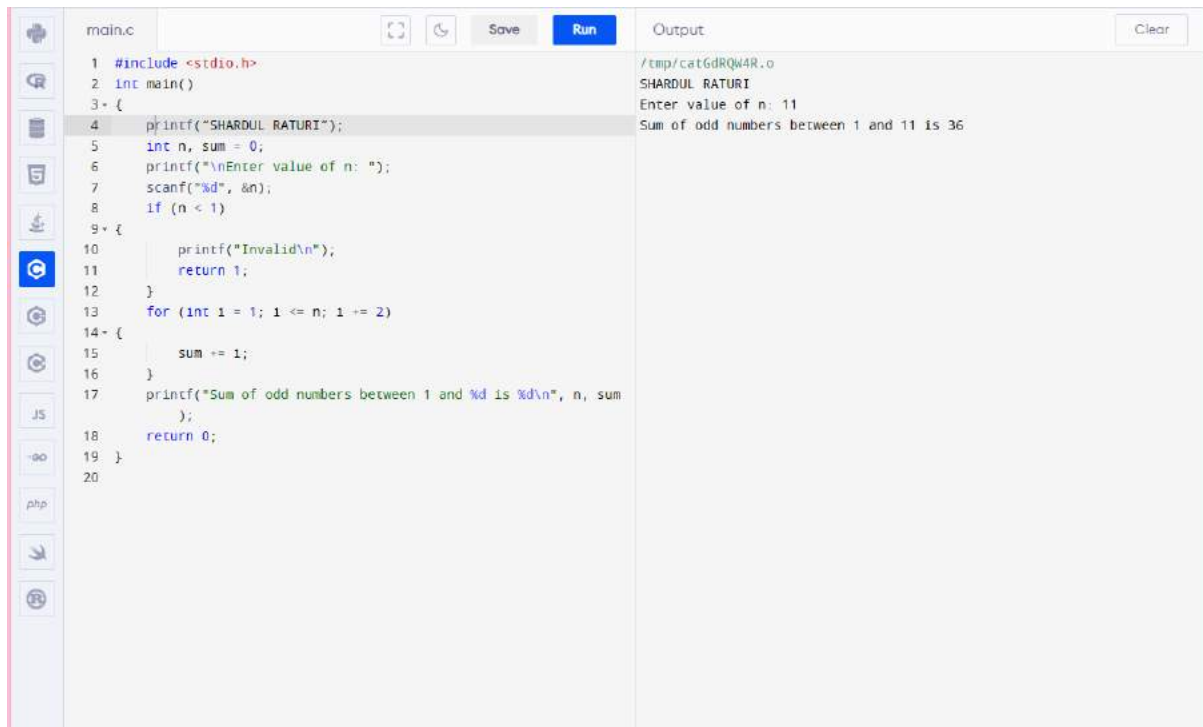
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter value of n: 22
Sum of even numbers between 1 and 22 is 132

```

❖ PROGRAM 32:- TO FIND SUM OF ALL ODD NUMBER BETWEEN 1 TO N

```
#include <stdio.h>

int main()
{
    Printf("SHARDUL RATURI");
    int n, sum = 0;
    printf("\nEnter value of n: ");
    scanf("%d", &n);
    if (n < 1)
    {
        printf("Invalid\n");
        return 1;
    }
    for (int i = 1; i <= n; i += 2)
    {
        sum += i;
    }
    printf("Sum of odd numbers between 1 and %d is %d\n", n, sum);
    return 0;
}
```

```
main.c
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int n, sum = 0;
6     printf("\nEnter value of n: ");
7     scanf("%d", &n);
8     if (n < 1)
9     {
10        printf("Invalid\n");
11        return 1;
12    }
13    for (int i = 1; i <= n; i += 2)
14    {
15        sum += i;
16    }
17    printf("Sum of odd numbers between 1 and %d is %d\n", n, sum);
18    return 0;
19 }
20
```

Output

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter value of n: 11
Sum of odd numbers between 1 and 11 is 36
```

❖ PROGRAM 33:- TO PRINT MULTIPLICATION TABLE OF ANY NUMBER

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int num;
```

```
    printf("\nEnter an integer to get its multiplication table: ");
```

```
    scanf("%d", &num);
```

```
    printf("Multiplication Table for %d:\n", num);
```

```
    for (int i = 1; i <= 10; i++)
```

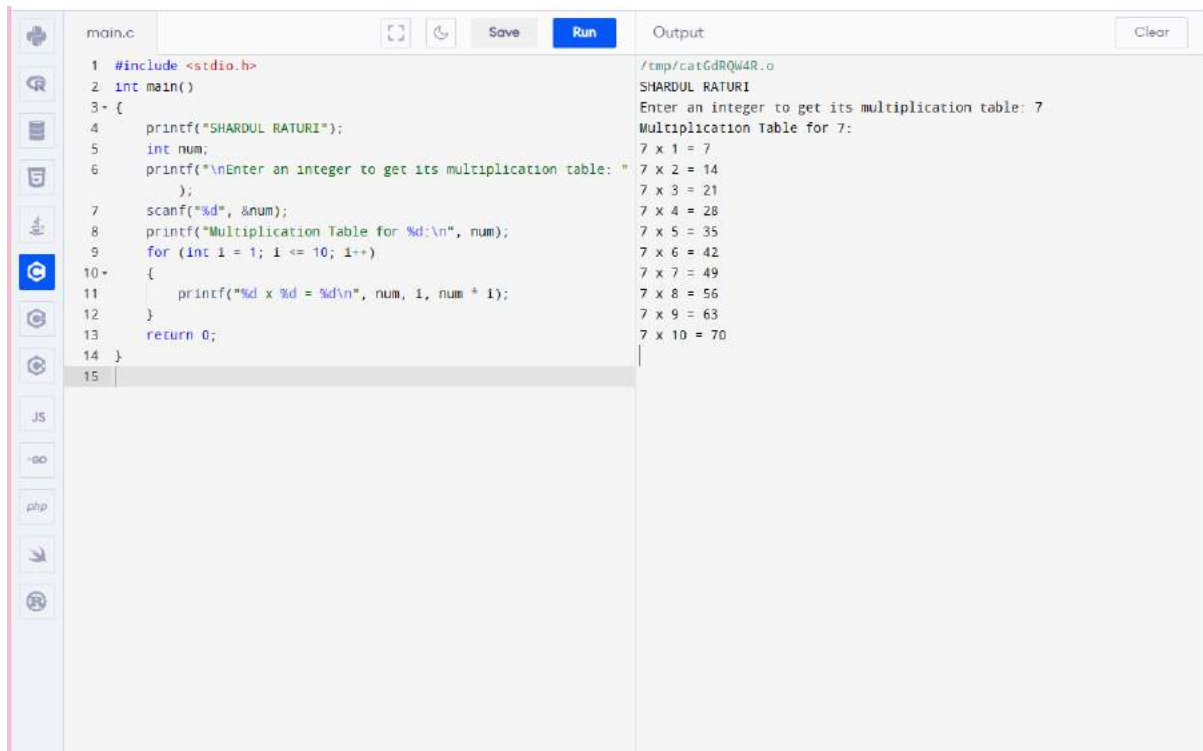
```
    {
```

```

        printf("%d x %d = %d\n", num, i, num * i);
    }

    return 0;
}

```



```

main.cpp
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num;
6     printf("\nEnter an integer to get its multiplication table: ");
7     scanf("%d", &num);
8     printf("Multiplication Table for %d:\n", num);
9     for (int i = 1; i <= 10; i++)
10    {
11        printf("%d x %d = %d\n", num, i, num * i);
12    }
13    return 0;
14 }
15
Output
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter an integer to get its multiplication table: 7
Multiplication Table for 7:
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
7 x 6 = 42
7 x 7 = 49
7 x 8 = 56
7 x 9 = 63
7 x 10 = 70

```

❖ PROGRAM 34:- TO COUNT NUMBERS OF DIGITS IN A NUMBER

```

#include <stdio.h>

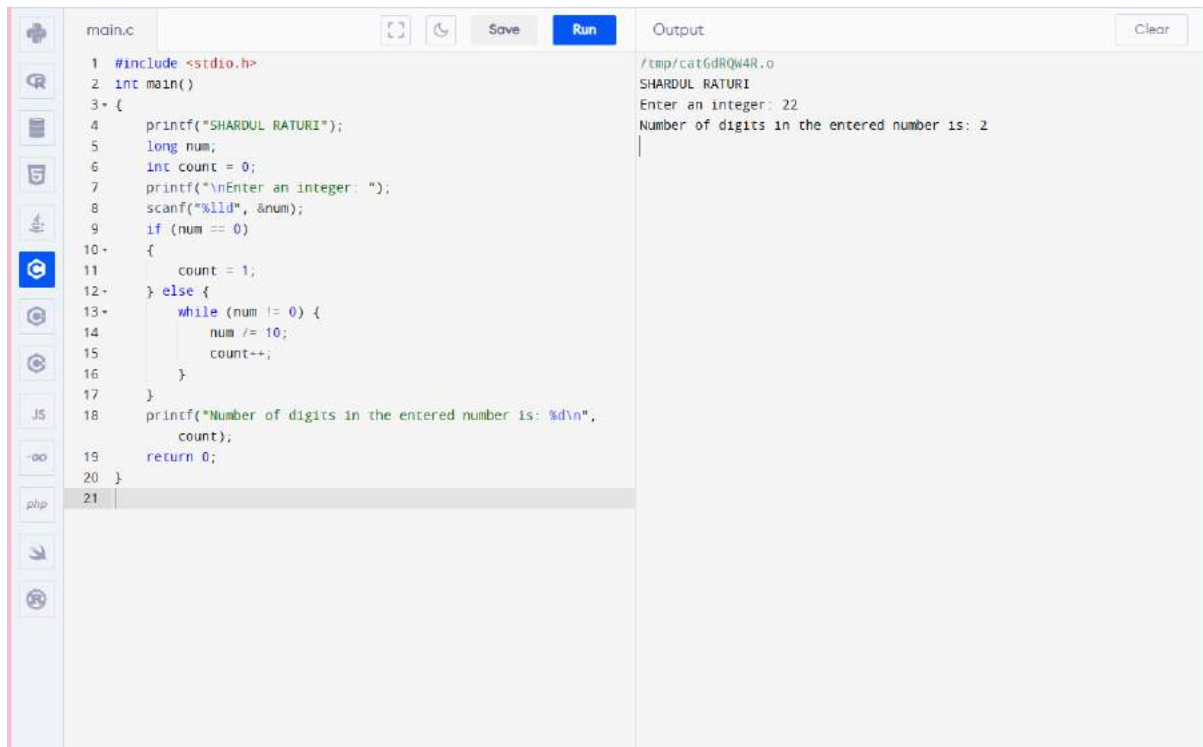
int main()
{
    printf("SHARDUL RATURI");

    long num;

    int count = 0;

```

```
printf("\nEnter an integer: ");
scanf("%lld", &num);
if (num == 0)
{
    count = 1;
} else {
    while (num != 0) {
        num /= 10;
        count++;
    }
}
printf("Number of digits in the entered number is: %d\n", count);
return 0;
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     long num;
6     int count = 0;
7     printf("\nEnter an integer: ");
8     scanf("%lld", &num);
9     if (num == 0)
10    {
11        count = 1;
12    } else {
13        while (num != 0) {
14            num /= 10;
15            count++;
16        }
17    }
18    printf("Number of digits in the entered number is: %d\n",
19          count);
19    return 0;
20 }
21
```

Output

```
/tmp/cat6dRQW4R.o
SHARDUL RATURI
Enter an integer: 22
Number of digits in the entered number is: 2
```

❖ PROGRAM 35:- TO FIND FIRST AND LAST DIGIT OF A NUMBER

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int num, firstDigit, lastDigit;
```

```
    printf("\nEnter an integer: ");
```

```
    scanf("%d", &num);
```

```
    lastDigit = num % 10;
```

```
while (num >= 10)
{
    num /= 10;
}
firstDigit = num;
printf("First digit: %d\n", firstDigit);
printf("Last digit: %d\n", lastDigit);
return 0;
}
```

The screenshot shows a C program in a code editor. The code is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num, firstDigit, lastDigit;
6     printf("\nEnter an integer: ");
7     scanf("%d", &num);
8     lastDigit = num % 10;
9     while (num >= 10)
10    {
11        num /= 10;
12    }
13    firstDigit = num;
14    printf("First digit: %d\n", firstDigit);
15    printf("Last digit: %d\n", lastDigit);
16    return 0;
17 }
```

The output window shows the following text:

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter an integer: 12
First digit: 1
Last digit: 2
```

PROGRAM 36:- TO FIND SUM OF FIRST AND LAST DIGIT OF A NUMBER

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int num, firstDigit, lastDigit, sum;
```

```
    printf("\nEnter an integer: ");
```

```
    scanf("%d", &num);
```

```
    lastDigit = num % 10;
```

```
    while (num >= 10) {
```

```
        num /= 10;
```

```
}
```

```

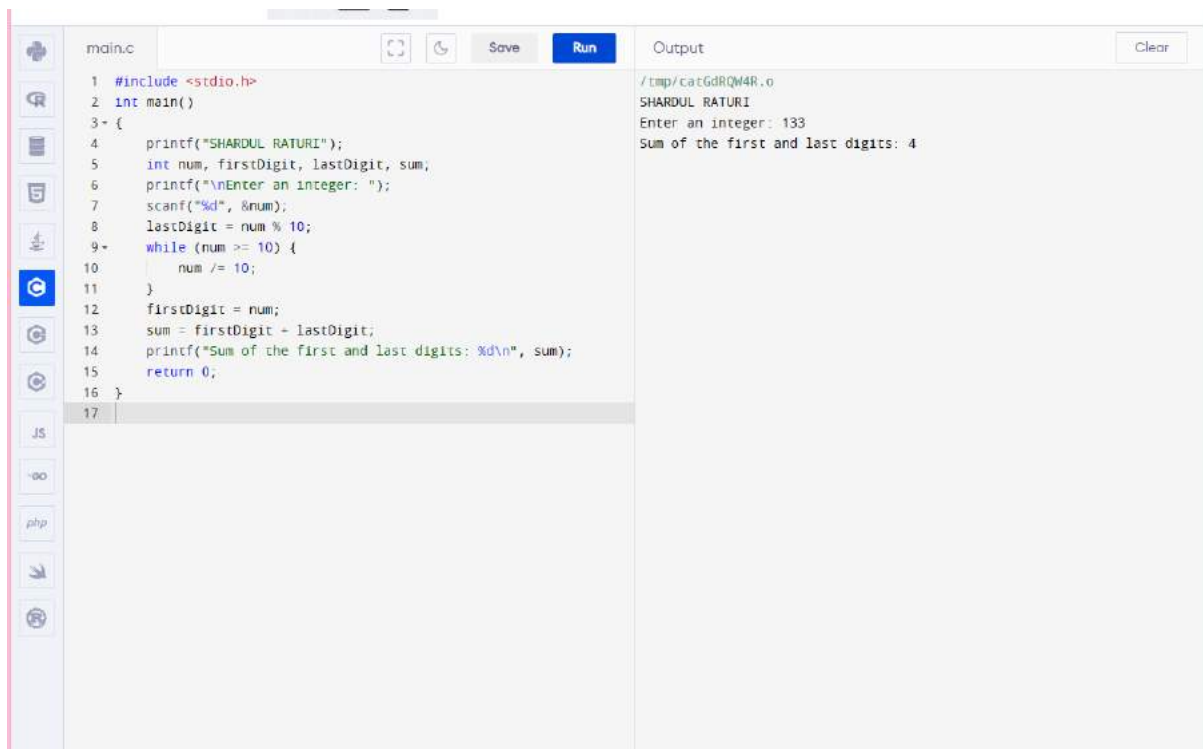
firstDigit = num;

sum = firstDigit + lastDigit;

printf("Sum of the first and last digits: %d\n", sum);

return 0;
}

```



```

main.cpp
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num, firstDigit, lastDigit, sum;
6     printf("\nEnter an integer: ");
7     scanf("%d", &num);
8     lastDigit = num % 10;
9     while (num >= 10) {
10        num /= 10;
11    }
12    firstDigit = num;
13    sum = firstDigit + lastDigit;
14    printf("Sum of the first and last digits: %d\n", sum);
15    return 0;
16 }
17

```

Output

```

/tmp/catGdRQM4R.o
SHARDUL RATURI
Enter an integer: 133
Sum of the first and last digits: 4

```

❖ PROGRAM 38:- TO CALCULATE SUM OF DIGITS IN A NUMBER

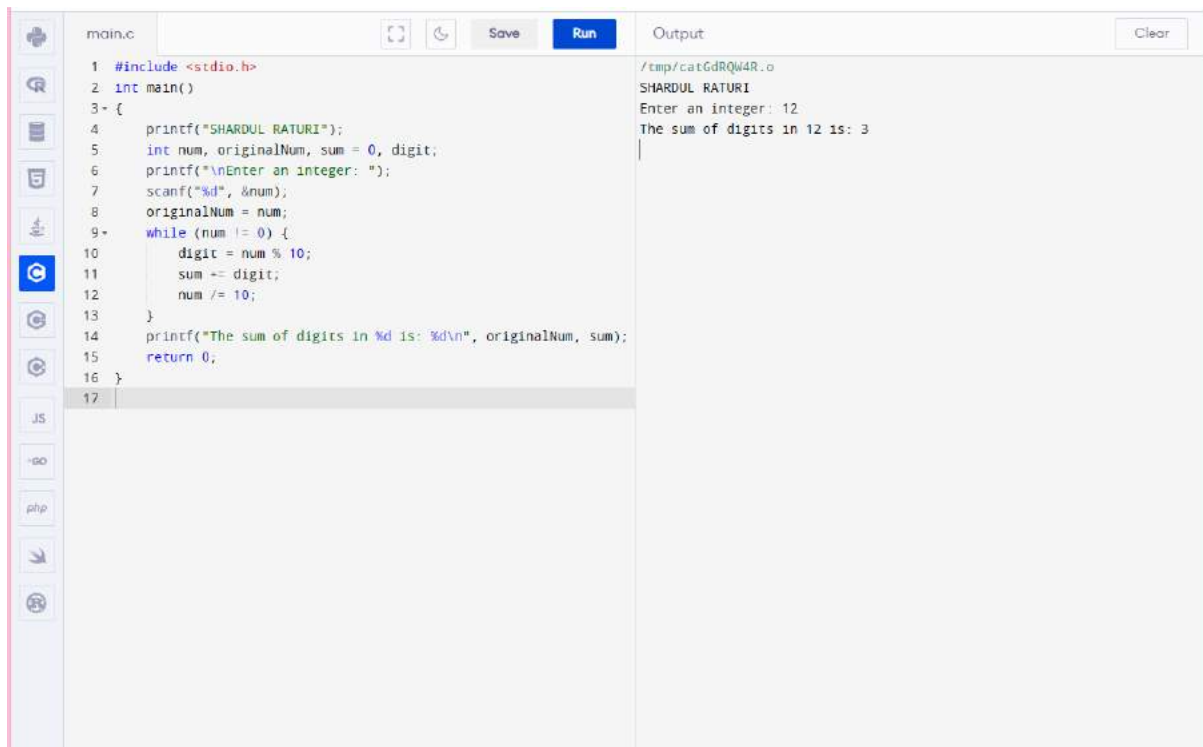
```

#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");
}

```

```
int num, originalNum, sum = 0, digit;
printf("\nEnter an integer: ");
scanf("%d", &num);
originalNum = num;
while (num != 0) {
    digit = num % 10;
    sum += digit;
    num /= 10;
}
printf("The sum of digits in %d is: %d\n", originalNum, sum);
return 0;
}
```

The screenshot shows a C++ IDE with a file named 'main.c'. The code is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num, originalNum, sum = 0, digit;
6     printf("\nEnter an integer: ");
7     scanf("%d", &num);
8     originalNum = num;
9     while (num != 0) {
10         digit = num % 10;
11         sum += digit;
12         num /= 10;
13     }
14     printf("The sum of digits in %d is: %d\n", originalNum, sum);
15     return 0;
16 }
```

The output window on the right shows the following text:

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter an integer: 12
The sum of digits in 12 is: 3
```

❖ PROGRAM 39:- TO CALCULATE PRODUCT OF DIGITS OF A NUMBER

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int num, originalNum, product = 1, digit;
```

```
    printf("\nEnter an integer: ");
```

```
    scanf("%d", &num);
```

```
    originalNum = num;
```

```
    while (num != 0) {
```

```
        digit = num % 10;
```

```

        product *= digit;

        num /= 10;

    }

    printf("The product of digits in %d is: %d\n", originalNum,
product);

    return 0;

}

```

The screenshot shows a C++ IDE with a file named 'main.c'. The code in the editor is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num, originalNum, product = 1, digit;
6     printf("\nEnter an integer: ");
7     scanf("%d", &num);
8     originalNum = num;
9     while (num != 0) {
10        digit = num % 10;
11        product *= digit;
12        num /= 10;
13    }
14    printf("The product of digits in %d is: %d\n", originalNum,
product);
15    return 0;
16 }
17

```

The 'Output' window on the right shows the following text:

```

/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter an integer: 12
The product of digits in 12 is: 2

```

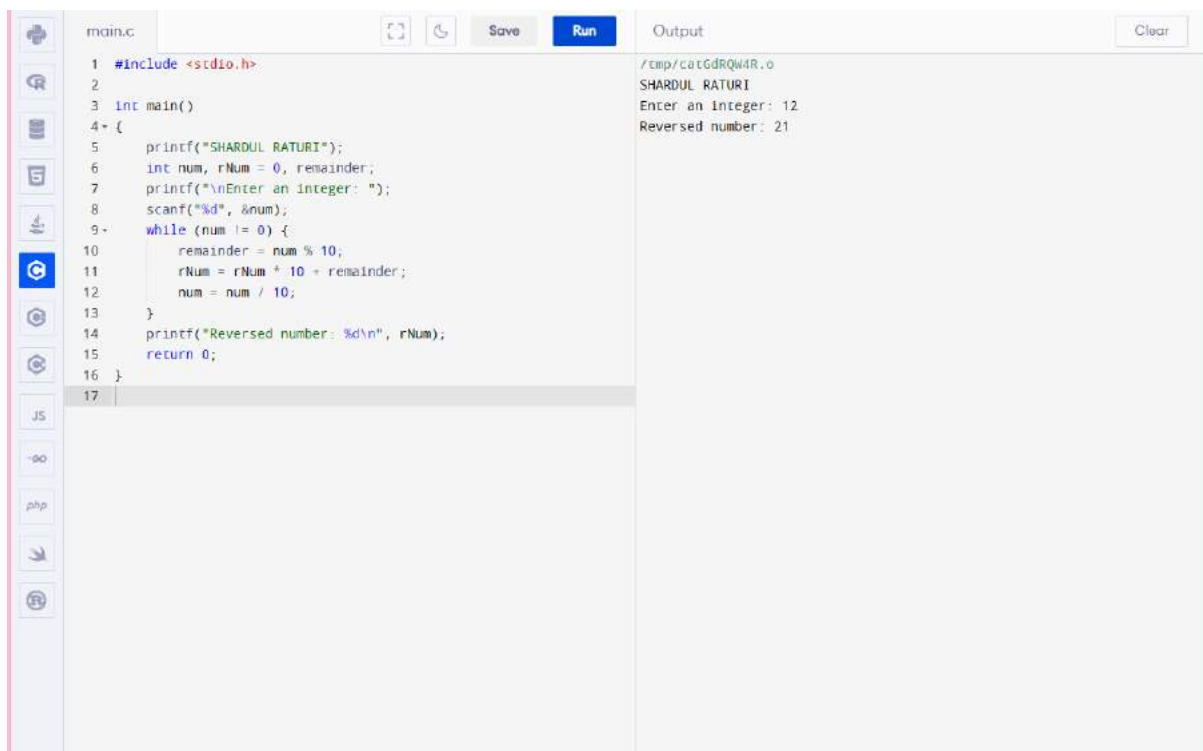
❖ PROGRAM 40:- TO ENTER A NUMBER AND PRINT ITS REVERSE

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
printf("SHARDUL RATURI");  
  
int num, rNum = 0, remainder;  
  
printf("\nEnter an integer: ");  
  
scanf("%d", &num);  
  
while (num != 0) {  
  
    remainder = num % 10;  
  
    rNum = rNum * 10 + remainder;  
  
    num = num / 10;  
  
}  
  
printf("Reversed number: %d\n", rNum);  
  
return 0;  
  
}
```



The screenshot shows a code editor with a file named 'main.c'. The code is a C program that prints 'SHARDUL RATURI', prompts the user to enter an integer, and then reverses the number using a while loop. The output window shows the program's execution: it prints 'SHARDUL RATURI', prompts 'Enter an Integer: 12', and then prints 'Reversed number: 21'.

```
1 #include <stdio.h>  
2  
3 int main()  
4 {  
5     printf("SHARDUL RATURI");  
6     int num, rNum = 0, remainder;  
7     printf("\nEnter an integer: ");  
8     scanf("%d", &num);  
9     while (num != 0) {  
10        remainder = num % 10;  
11        rNum = rNum * 10 + remainder;  
12        num = num / 10;  
13    }  
14    printf("Reversed number: %d\n", rNum);  
15    return 0;  
16 }  
17
```

Output

```
/tmp/catGdRQw4R.o  
SHARDUL RATURI  
Enter an Integer: 12  
Reversed number: 21
```

❖ PROGRAM 41:- TO CHECK WHETHER THE NUMBER IS PALINDROME OR NOT

```
#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");

    int num, originalNum, rNum = 0, remainder;

    printf("\nEnter an integer: ");

    scanf("%d", &num);

    originalNum = num;

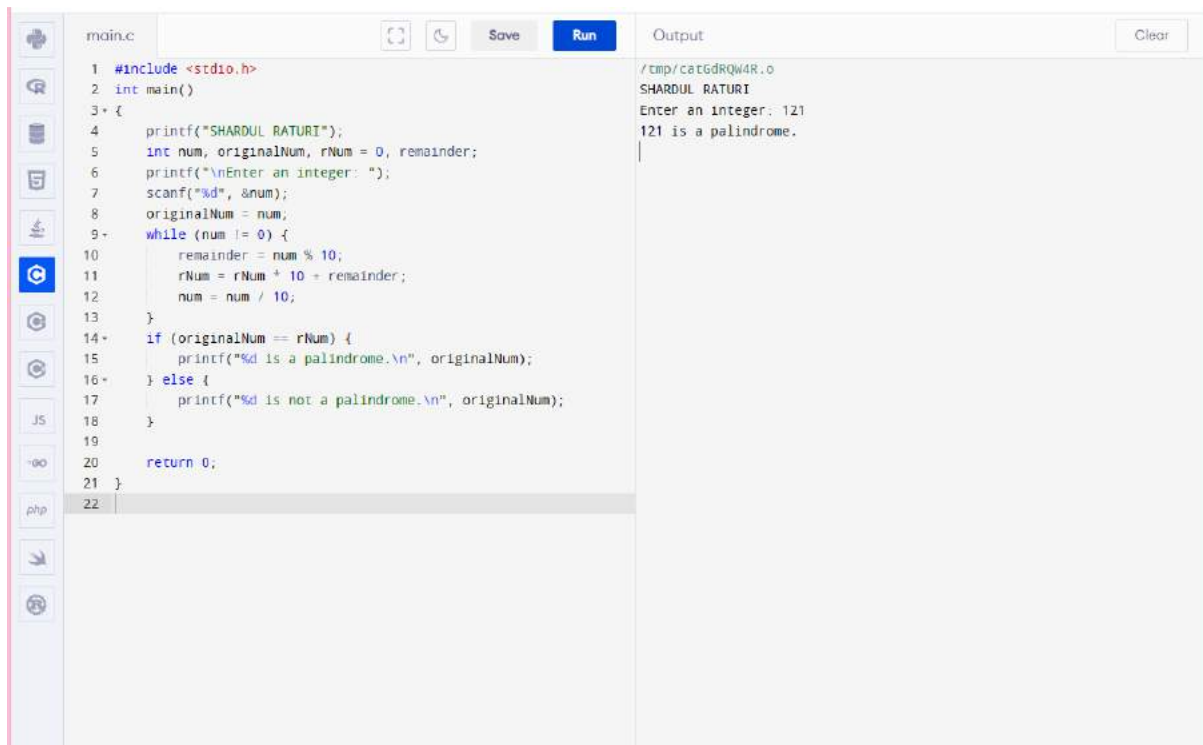
    while (num != 0) {
        remainder = num % 10;

        rNum = rNum * 10 + remainder;

        num = num / 10;
    }

    if (originalNum == rNum) {
        printf("%d is a palindrome.\n", originalNum);
    } else {
        printf("%d is not a palindrome.\n", originalNum);
    }

    return 0;
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     printf("SHARDUL RATURI");
5     int num, originalNum, rNum = 0, remainder;
6     printf("\nEnter an integer: ");
7     scanf("%d", &num);
8     originalNum = num;
9     while (num != 0) {
10        remainder = num % 10;
11        rNum = rNum * 10 + remainder;
12        num = num / 10;
13    }
14    if (originalNum == rNum) {
15        printf("%d is a palindrome.\n", originalNum);
16    } else {
17        printf("%d is not a palindrome.\n", originalNum);
18    }
19
20    return 0;
21 }
22
```

Output

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter an integer: 121
121 is a palindrome.
```

PROGRAM 42:- TO FIND ALL FACTORS OF A NUMBER

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("SHARDUL RATURI");
```

```
    int num;
```

```
    printf("\nEnter an integer: ");
```

```
    scanf("%d", &num);
```

```
    printf("Factors of %d are: ", num);
```

```

for (int i = 1; i <= num; i++) {
    if (num % i == 0) {
        printf("%d ", i);
    }
}

printf("\n");

return 0;
}

```

```

main.c
1 #include <stdio.h>
2
3 int main()
4 {
5     printf("SHARDUL RATURI");
6     int num;
7     printf("\nEnter an integer: ");
8     scanf("%d", &num);
9     printf("Factors of %d are: ", num);
10    for (int i = 1; i <= num; i++) {
11        if (num % i == 0) {
12            printf("%d ", i);
13        }
14    }
15    printf("\n");
16    return 0;
17 }
18

```

Output

```

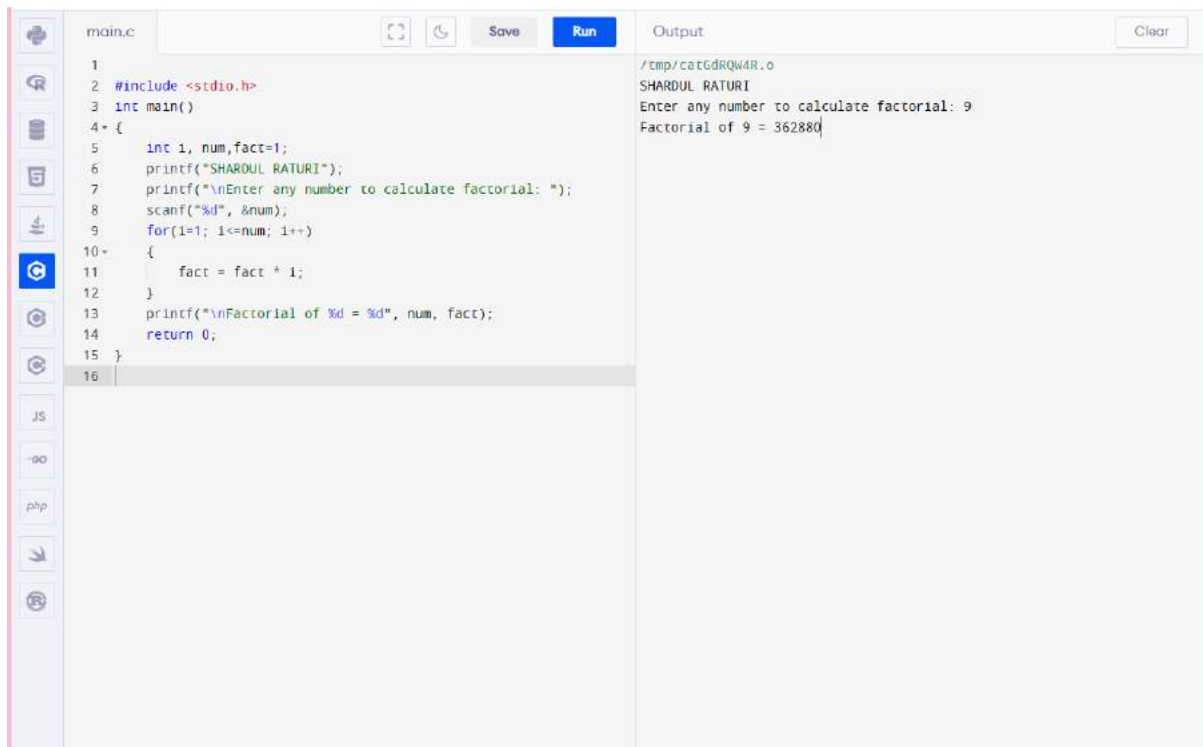
/tmp/catGdRQw4R.o
SHARDUL RATURI
Enter an integer: 12
Factors of 12 are: 1 2 3 4 6 12

```

PROGRAM 43:- TO FIND FACTORIAL OF A NUMBER

```
#include <stdio.h>
```

```
int main()
{
    int i, num, fact=1;
    printf("SHARDUL RATURI");
    printf("\nEnter any number to calculate factorial: ");
    scanf("%d", &num);
    for(i=1; i<=num; i++)
    {
        fact = fact * i;
    }
    printf("\nFactorial of %d = %d", num, fact);
    return 0;
}
```



The screenshot shows a C++ IDE with a file named 'main.c'. The code is a program to calculate the factorial of a number. It includes `<stdio.h>`, defines `main()`, and uses `printf` and `scanf` for input/output. A `for` loop calculates the factorial. The output window shows the program's execution: it prints 'SHARDUL RATURI', prompts for a number, and calculates the factorial of 9 as 362880.

```
1
2 #include <stdio.h>
3 int main()
4 {
5     int i, num, fact=1;
6     printf("SHARDUL RATURI");
7     printf("\nEnter any number to calculate factorial: ");
8     scanf("%d", &num);
9     for(i=1; i<=num; i++)
10    {
11        fact = fact * i;
12    }
13    printf("\nFactorial of %d = %d", num, fact);
14    return 0;
15 }
```

Output: /tmp/catGdRQW4R.o
SHARDUL RATURI
Enter any number to calculate factorial: 9
Factorial of 9 = 362880

PROGRAM 44:- TO FIND HCF OF TWO NUMBERS

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int i, num1, num2, min, hcf=1;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter any two numbers to find HCF: ");
```

```
    scanf("%d%d", &num1, &num2);
```

```
    min = (num1<num2) ? num1 : num2;
```

```
    for(i=1; i<=min; i++)
```

```
    {
```

```
        if(num1%i==0 && num2%i==0)
```



```
{  
  
}
```

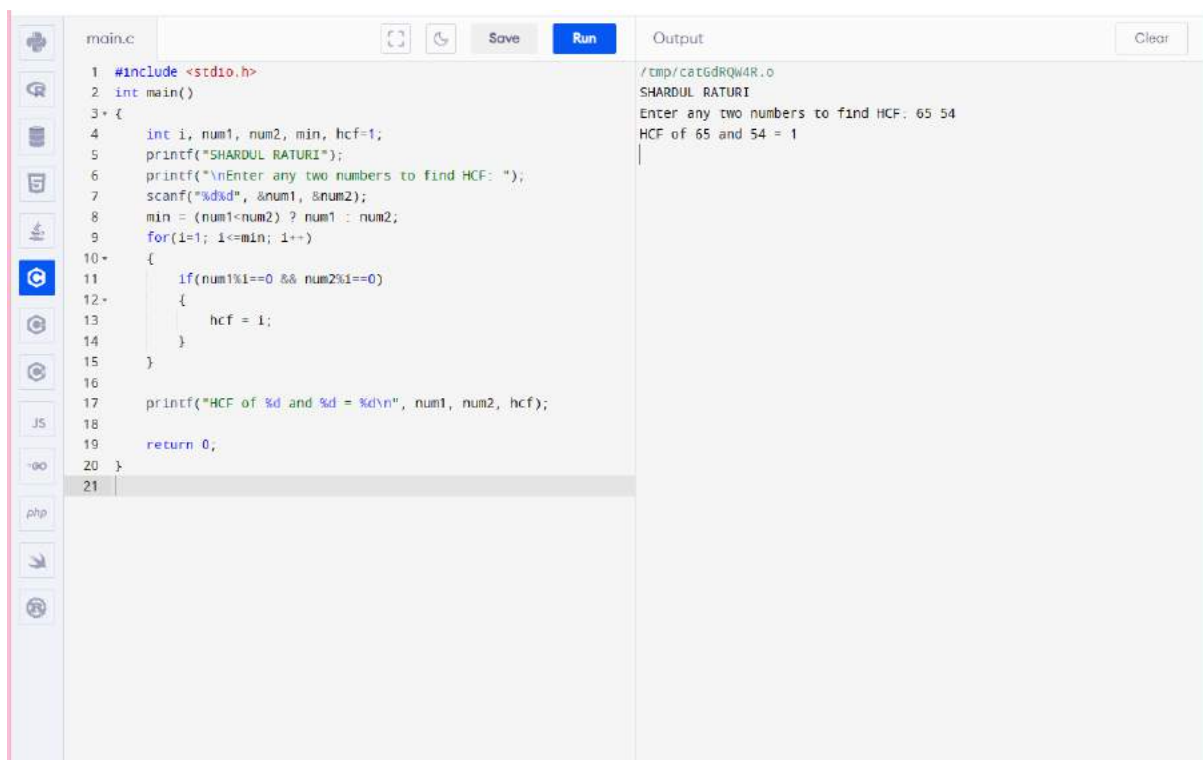
```
hcf = i;
```

```
}
```

```
printf("HCF of %d and %d = %d\n", num1, num2, hcf);
```

```
return 0;
```

```
}
```



The screenshot shows a C program in a code editor. The code is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     int i, num1, num2, min, hcf=1;
5     printf("SHARDUL RATURI");
6     printf("\nEnter any two numbers to find HCF: ");
7     scanf("%d%d", &num1, &num2);
8     min = (num1<num2) ? num1 : num2;
9     for(i=1; i<=min; i++)
10     {
11         if(num1%i==0 && num2%i==0)
12         {
13             hcf = i;
14         }
15     }
16     printf("HCF of %d and %d = %d\n", num1, num2, hcf);
17     return 0;
18 }
19
20
21
```

The output of the program is shown on the right:

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter any two numbers to find HCF: 65 54
HCF of 65 and 54 = 1
```

PROGRAM 45:- TO FIND LCM OF TWO NUMBERS

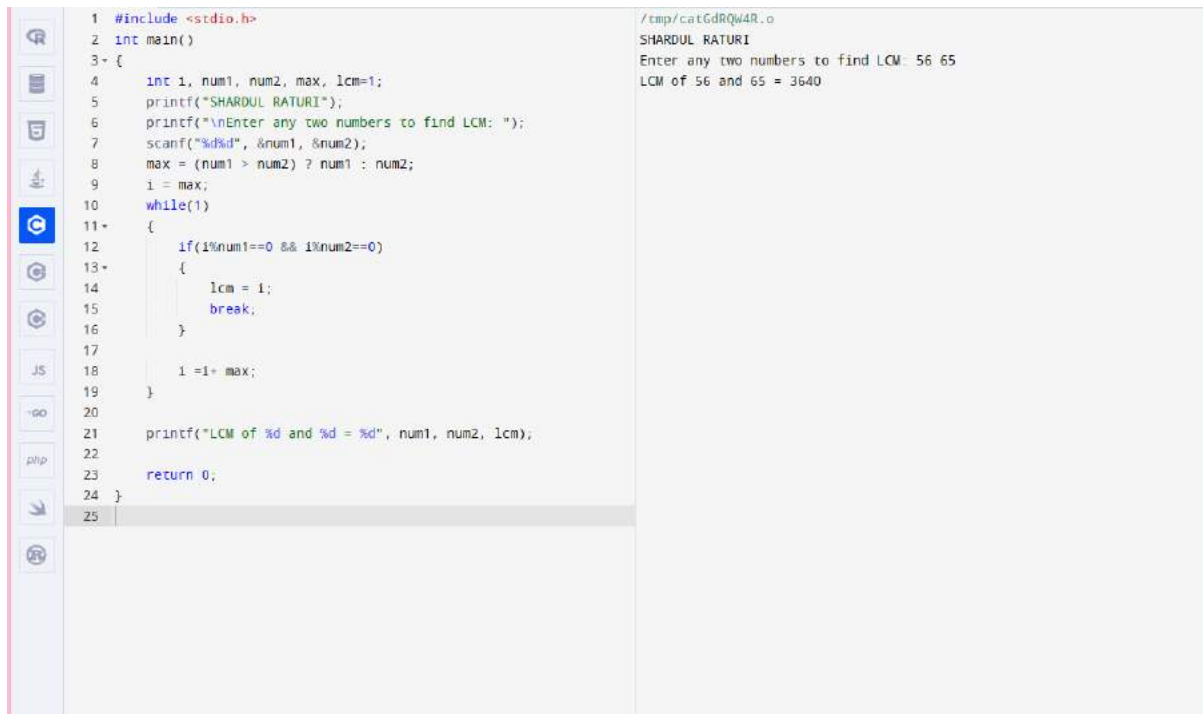
```
#include <stdio.h>
```

```
int main()
{
    int i, num1, num2, max, lcm=1;
    printf("SHARDUL RATURI");
    printf("\nEnter any two numbers to find LCM: ");
    scanf("%d%d", &num1, &num2);
    max = (num1 > num2) ? num1 : num2;
    i = max;
    while(1)
    {
        if(i%num1==0 && i%num2==0)
        {
            lcm = i;
            break;
        }

        i =i+ max;
    }

    printf("LCM of %d and %d = %d", num1, num2, lcm);

    return 0;
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     int i, num1, num2, max, lcm=1;
5     printf("SHARDUL RATURI");
6     printf("\nEnter any two numbers to find LCM: ");
7     scanf("%d%d", &num1, &num2);
8     max = (num1 > num2) ? num1 : num2;
9     i = max;
10    while(1)
11    {
12        if(1%num1==0 && 1%num2==0)
13        {
14            lcm = i;
15            break;
16        }
17        i = i * max;
18    }
19    printf("LCM of %d and %d = %d", num1, num2, lcm);
20    return 0;
21 }
```

/tmp/catGdRQw4R.o
SHARDUL RATURI
Enter any two numbers to find LCM: 56 65
LCM of 56 and 65 = 3640

PROGRAM 46:- TO CHECK WHETHER THE NUMBER IS PRIME OR NOT

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int i, num, temp = 0;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter any number to Check for Prime: ");
```

```
    scanf("%d", &num);
```

```
    for (i = 2; i <= num / 2; i++)
```

```
    {
```

```
        if (num % i == 0)
```

```
        {
```

```
            temp++;
```

```

        break;
    }
}
if (temp == 0 && num != 1)
{
    printf("%d is a Prime number", num);
}
else
{
    printf("%d is not a Prime number", num);
}
return 0;
}

```

The screenshot shows a C++ IDE with a file named `main.c`. The code implements a prime number checker. It includes `<stdio.h>` and defines a `main` function. Inside `main`, it declares `int i, num, temp = 0;`. It prints "SHARDUL RATURI" and prompts the user to "Enter any number to Check for Prime: ". The user has entered 45. A `for` loop starts at `i = 2` and goes up to `num / 2`. Inside the loop, it checks if `num % i == 0`. If true, it increments `temp` and breaks the loop. After the loop, it uses an `if-else` statement to print whether the number is a prime or not. The output window shows the program's execution: the name, the prompt, the input 45, and the result "45 is not a Prime number".

```

main.c
1 #include <stdio.h>
2 int main()
3 {
4     int i, num, temp = 0;
5     printf("SHARDUL RATURI");
6     printf("\nEnter any number to Check for Prime: ");
7     scanf("%d", &num);
8     for (i = 2; i <= num / 2; i++)
9     {
10         if (num % i == 0)
11         {
12             temp++;
13             break;
14         }
15     }
16     if (temp == 0 && num != 1)
17     {
18         printf("%d is a Prime number", num);
19     }
20     else
21     {
22         printf("%d is not a Prime number", num);
23     }
24     return 0;
25 }
26
Output
/tmp/catGdRQw4R.o
SHARDUL RATURI
Enter any number to Check for Prime: 45
45 is not a Prime number

```

PROGRAM 47:- TO PRINT ALL PRIME NUMBER BETWEEN 1 to n

```
#include<stdio.h>

int main()
{
    int num,i,count,n;
    printf("SHARDUL RATURI");
    printf("\nEnter num: ");
    scanf("%d",&n);
    for(num = 1;num<=n;num++)
    {
        count = 0;
        for(i=2;i<=num/2;i++)
        {
            if(num%i==0)
            {
                count++;
                break;
            }
        }
        if(count==0 && num!= 1)
        {
            printf("%d ",num);
        }
    }
}
```

```

    }

}

return 0;

}

```

```

1  #include<stdio.h>
2  int main()
3  {
4      int num,i,count,n;
5      printf("SHARDUL RATURI");
6      printf("\nEnter num: ");
7      scanf("%d",&n);
8      for(num = 1; num<=n; num++)
9      {
10         count = 0;
11         for(i=2; i<=num/2; i++)
12         {
13             if(num%i==0)
14             {
15                 count++;
16                 break;
17             }
18         }
19         if(count==0 && num!= 1)
20         {
21             printf("%d ",num);
22         }
23     }
24     return 0;
25 }

```

Output: /tmp/catGdRQw4R.o
SHARDUL RATURI
Enter num: 12
2 3 5 7 11 |

PROGRAM 48:-TO FIND SUM OF ALL PRIME NUMBER BETWEEN 1 to n

```

#include <stdio.h>

int main()
{
    int j, N, i, isPrime, Sum = 0;

    printf("SHARDUL RATURI");

    printf("\nEnter a Number\n");

```

```
scanf("%d", &N);
```

```
for(j = 2; j <= N; j++) {
```

```
    isPrime = 1;
```

```
    for(i = 2; i <=(j/2); ++i) {
```

```
        if(j%i==0) {
```

```
            isPrime = 0;
```

```
            break;
```

```
        }
```

```
    }
```

```
    if(isPrime==1)
```

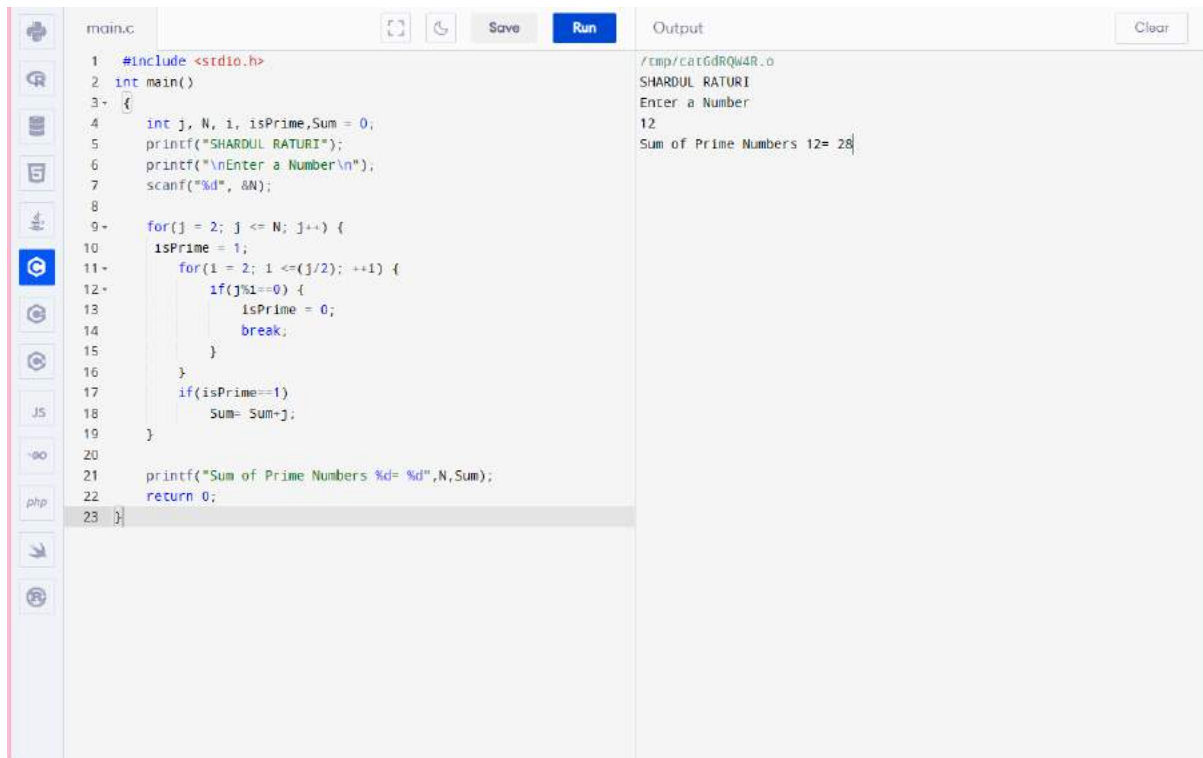
```
        Sum= Sum+j;
```

```
}
```

```
printf("Sum of Prime Numbers %d= %d",N,Sum);
```

```
return 0;
```

```
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     int j, N, i, isPrime, Sum = 0;
5     printf("SHARDUL RATURI");
6     printf("\nEnter a Number\n");
7     scanf("%d", &N);
8
9     for(j = 2; j <= N; j++) {
10         isPrime = 1;
11         for(i = 2; i <= (j/2); ++i) {
12             if(j%i==0) {
13                 isPrime = 0;
14                 break;
15             }
16         }
17         if(isPrime==1)
18             Sum= Sum+j;
19     }
20
21     printf("Sum of Prime Numbers %d= %d",N,Sum);
22     return 0;
23 }
```

Output

```
/tmp/catGdRQw4R.o
SHARDUL RATURI
Enter a Number
12
Sum of Prime Numbers 12= 28
```

PROGRAM 49:- TO FIND ALL PRIME FACTORS OF A NUMBER

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    int n,i;
```

```
    printf("Enter a positive integer\n");
```

```
    scanf("%d", &n);
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nPrime Factors of %d are\n", n);
```

```
    for(i = 2; n > 1; i++)
```

```
    {
```



```

while(n % i == 0)
{
    printf("%d ", i);
    n = n / i;
}
}

printf("\n");

return 0;
}

```

The screenshot shows a C++ IDE with a source code editor on the left and a terminal on the right. The source code is as follows:

```

1 #include<stdio.h>
2 int main()
3 {
4     int n,i;
5     printf("Enter a positive integer\n");
6     scanf("%d", &n);
7     printf("SHARDUL RATURI");
8     printf("\nPrime Factors of %d are\n", n);
9     for(i = 2; n > 1; i++)
10    {
11        while(n % i == 0)
12        {
13            printf("%d ", i);
14            n = n / i;
15        }
16    }
17    printf("\n");
18    return 0;
19 }
20

```

The terminal output on the right shows the program's execution:

```

/tmp/catGdRQW4R.o
Enter a positive integer
43
SHARDUL RATURI
Prime Factors of 43 are
43

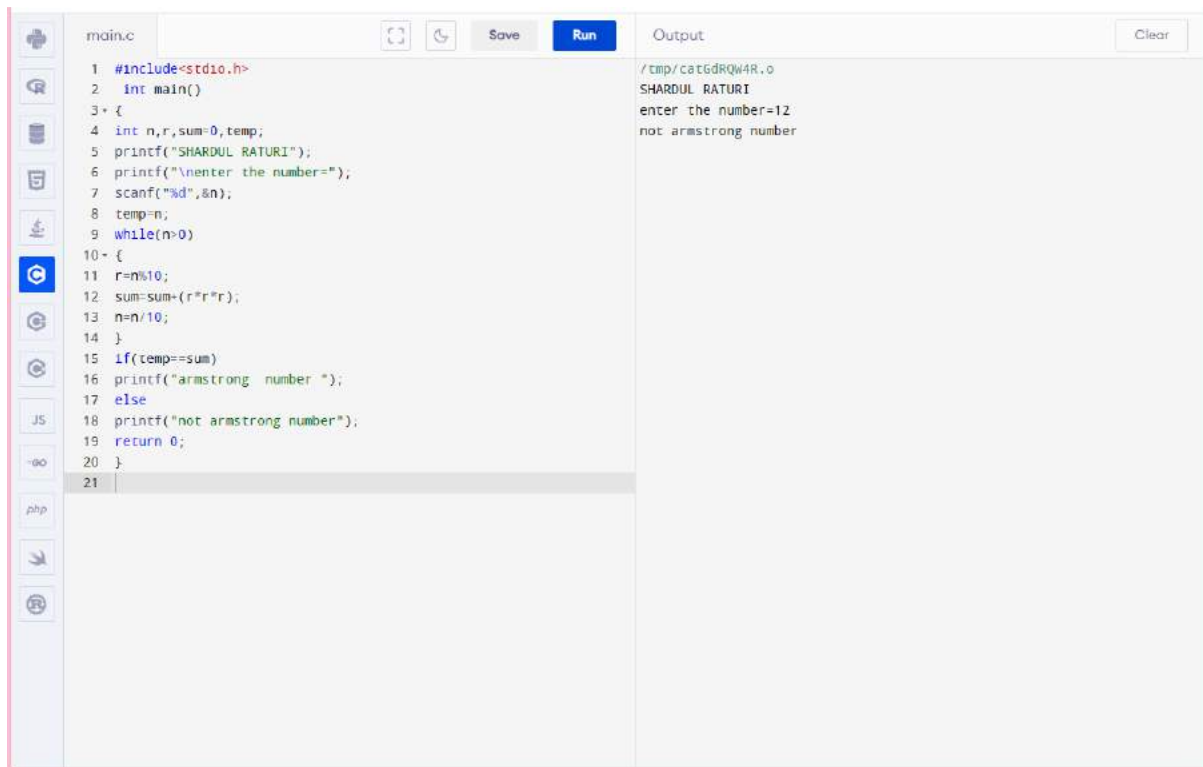
```

PROGRAM 50:-TO CHECK WHETHER A NUMBER IS ARMSTRONG OR NOT

```
#include<stdio.h>
```

```
int main()
```

```
{  
int n,r,sum=0,temp;  
printf("SHARDUL RATURI");  
printf("\nenter the number=");  
scanf("%d",&n);  
temp=n;  
while(n>0)  
{  
r=n%10;  
sum=sum+(r*r*r);  
n=n/10;  
}  
if(temp==sum)  
printf("armstrong number ");  
else  
printf("not armstrong number");  
return 0;  
}
```



```
1 #include<stdio.h>
2 int main()
3 {
4     int n,r,sum=0,temp;
5     printf("SHARDUL RATURI");
6     printf("\nenter the number=");
7     scanf("%d",&n);
8     temp=n;
9     while(n>0)
10    {
11        r=n%10;
12        sum=sum+(r*r*r);
13        n=n/10;
14    }
15    if(temp==sum)
16        printf("armstrong number ");
17    else
18        printf("not armstrong number");
19    return 0;
20 }
21
```

Output

```
/tmp/cat6dRQW4R.o
SHARDUL RATURI
enter the number=12
not armstrong number
```

PROGRAM 51:- TO PRINT ALL ARMSTRONG NUMBER BETWEEN 1 to n

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
int digit,n,sum=0,number;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nThe armstrong numbers are-");
```

```
scanf("%d",&n);
```

```
for(number = 1; number <=n; number++)
```

```
{
```

```
int temporary = number;
```

```
sum = 0;
```

```

while(temporary > 0)
{
    digit = temporary % 10;

    sum = sum + (digit * digit * digit);

    temporary = temporary / 10;
}

if(sum == number)

    printf("%d ",number);
}

return 0;
}

```

The screenshot shows a C++ IDE with a file named 'main.cpp'. The code is as follows:

```

1 #include<stdio.h>
2 int main()
3 {
4     int digit,n,sum=0,number;
5     printf("SHARDUL RATURI");
6     printf("\nThe armstrong numbers are-");
7     scanf("%d",&n);
8     for(number = 1; number <=n; number++)
9     {
10         int temporary = number;
11         sum = 0;
12         while(temporary > 0)
13         {
14             digit = temporary % 10;
15             sum = sum + (digit * digit * digit);
16             temporary = temporary / 10;
17         }
18         if(sum == number)
19             printf("%d ",number);
20     }
21     return 0;
22 }
23

```

The output window shows the following text:

```

/tmp/catGdRQW4R.o
SHARDUL RATURI
The armstrong numbers are-100
1

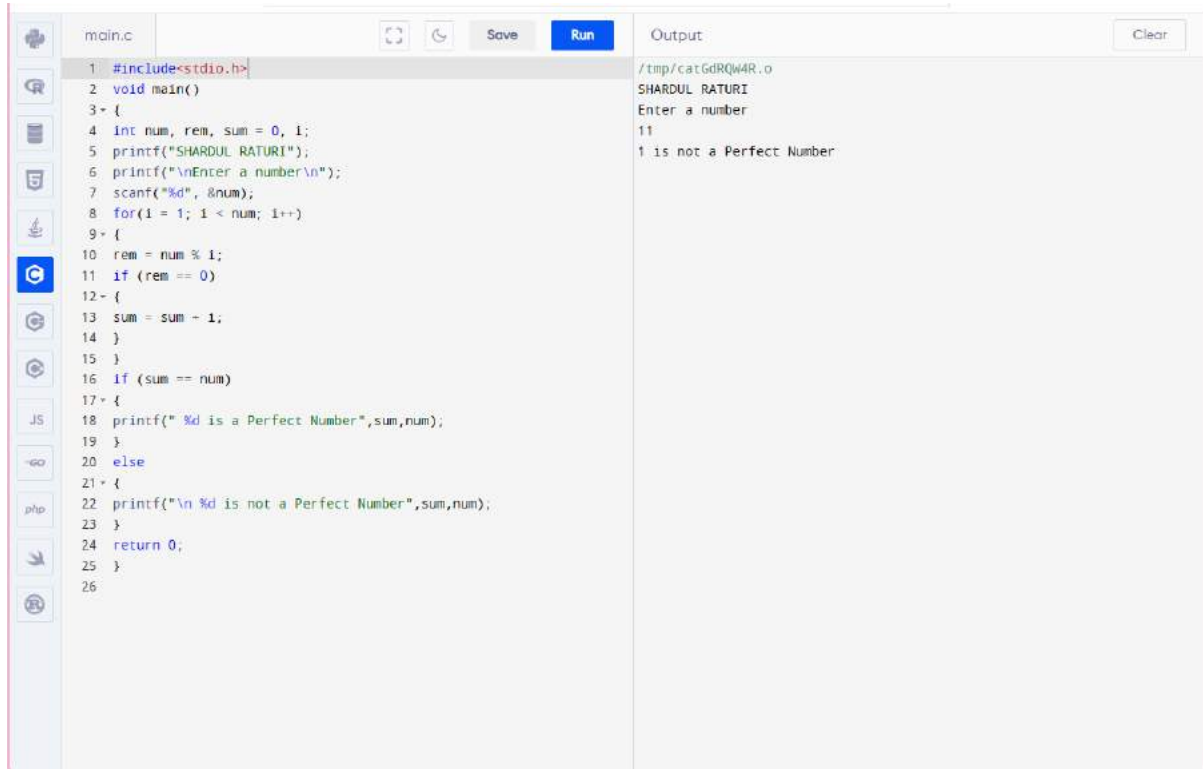
```

**PROGRAM 52:- TO CHECK WHETHER
NUMBER IS PERFECT NUMBER OR NOT**

```
#include<stdio.h>
#include<conio.h>
void main()
{
int num, rem, sum = 0, i;
printf("SHARDUL RATURI");
printf("\nEnter a number\n");
scanf("%d", &num);
for(i = 1; i < num; i++)
{
rem = num % i;
if (rem == 0)
{
sum = sum + i;
}
}
if (sum == num)
{
printf(" %d is a Perfect Number",sum,num);
}
else
{
printf("\n %d is not a Perfect Number",sum,num);
}
```

```
return 0;

}
```



```
1 #include<stdio.h>
2 void main()
3 {
4     int num, rem, sum = 0, i;
5     printf("SHARDUL RATURI");
6     printf("\nEnter a number\n");
7     scanf("%d", &num);
8     for(i = 1; i < num; i++)
9     {
10         rem = num % i;
11         if (rem == 0)
12         {
13             sum = sum + i;
14         }
15     }
16     if (sum == num)
17     {
18         printf(" %d is a Perfect Number", sum, num);
19     }
20     else
21     {
22         printf("\n %d is not a Perfect Number", sum, num);
23     }
24     return 0;
25 }
26
```

Output:

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter a number
11
1 is not a Perfect Number
```

PROGRAM 53:- TO PRINT ALL PERFECT NUMBER BETWEEN 1 to n

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int i, j, n, sum;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter limit: ");
```

```
    scanf("%d", &n);
```

```
    printf("\nAll Perfect numbers between 1 to %d:\n", n);
```

```
for(i=1; i<=n; i++)
{
    sum = 0;
    for(j=1; j<i; j++)
    {
        if(i % j == 0)
        {
            sum += j;
        }
    }
    if(sum == i)
    {
        printf("%d, ", i);
    }
}
return 0;
}
```

```
1 #include <stdio.h>
2 int main()
3 {
4     int i, j, n, sum;
5     printf("SHARDUL RATURI");
6     printf("\nEnter limit: ");
7     scanf("%d", &n);
8     printf("\nAll Perfect numbers between 1 to %d:\n", n);
9     for(i=1; i<=n; i++)
10    {
11        sum = 0;
12        for(j=1; j<i; j++)
13        {
14            if(i % j == 0)
15            {
16                sum += j;
17            }
18        }
19        if(sum == i)
20        {
21            printf("%d, ", i);
22        }
23    }
24    return 0;
25 }
26
```

Output

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter limit: 9
All Perfect numbers between 1 to 9:
6, |
```

PROGRAM 54:- Write a C program to check whether a number is Strong number or not.

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    int number, original, rem, sum=0, fact, i;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("/nEnter number: ");
```

```
    scanf("%d", &number);
```

```
    original = number;
```

```
    while(number != 0)
```

```
    {
```



```
    rem = number%10;
    fact = 1;
    for(i=1; i<=rem; i++)
    {
        fact = fact*i;
    }

    sum = sum + fact;
    number = number/10;
}
if(sum == original)
{
    printf("%d is STRONG.", original);
}
else
{
    printf("%d is NOT STRONG.", original);
}

return 0;
}
```

```
1 #include<stdio.h>
2 int main()
3 {
4     int number, original, rem, sum=0, fact, i;
5     printf("SHARDUL RATURI");
6     printf("\nEnter number: ");
7     scanf("%d", &number);
8     original = number;
9     while(number != 0)
10    {
11        rem = number%10;
12        fact = 1;
13        for(i=1; i<=rem; i++)
14        {
15            fact = fact*i;
16        }
17        sum = sum + fact;
18        number = number/10;
19    }
20    if(sum == original)
21    {
22        printf("%d is STRONG.", original);
23    }
24    else
25    {
26        printf("%d is NOT STRONG.", original);
27    }
28 }
29
30 return 0;
31 }
```

Output

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter number: 12
12 is NOT STRONG.
```

PROGRAM 54:- TO PRINT ALL STRONG NUMBERS BETWEEN 1 to n

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int factorial(int f) {
```

```
    int mul = 1;
```

```
    for (int i = 1; i <= f; i++) {
```

```
        mul = mul * i;
```

```
    }
```

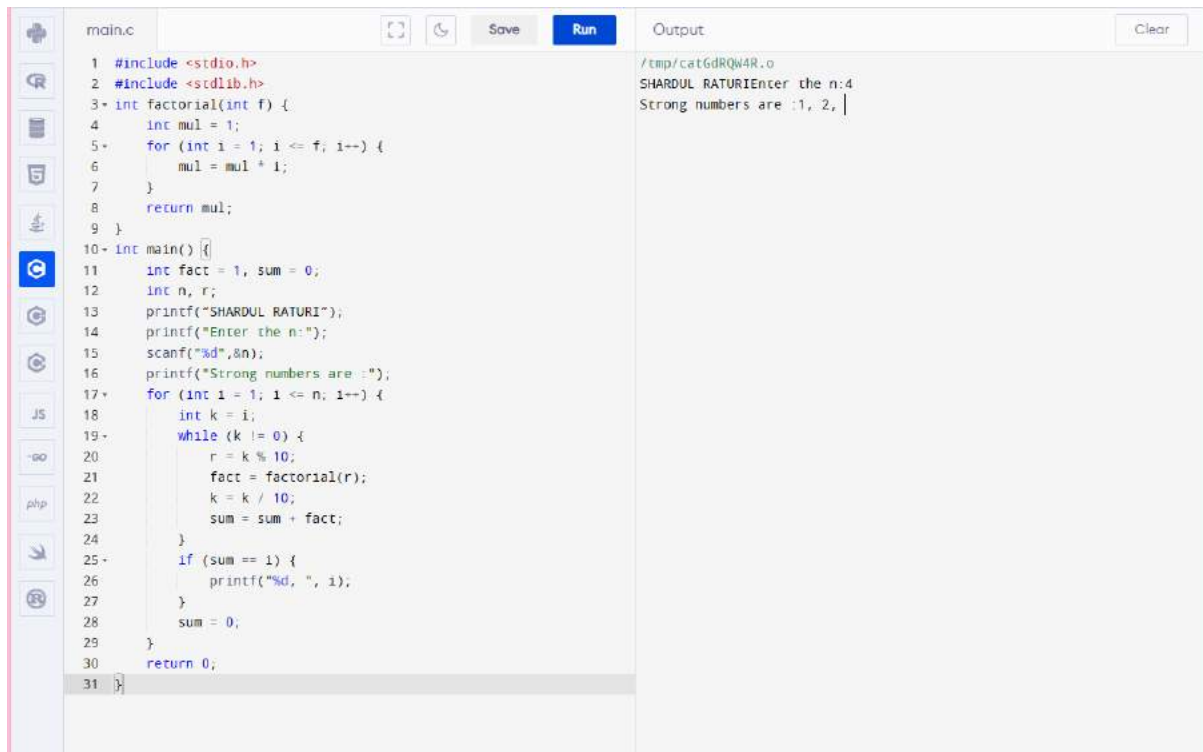
```
    return mul;
```

```
}
```

```
int main() {
```

```
    int fact = 1, sum = 0;
```

```
int n, r;
printf("SHARDUL RATURI");
printf("Enter the n:");
scanf("%d",&n);
printf("Strong numbers are :");
for (int i = 1; i <= n; i++) {
    int k = i;
    while (k != 0) {
        r = k % 10;
        fact = factorial(r);
        k = k / 10;
        sum = sum + fact;
    }
    if (sum == i) {
        printf("%d, ", i);
    }
    sum = 0;
}
return 0;
}
```



The screenshot shows a C++ IDE with a file named 'main.c'. The code defines a 'factorial' function and a 'main' function. The 'main' function prints 'SHARDUL RATURI', prompts for 'n', and then prints 'Strong numbers are :'. The output window shows the execution result: 'SHARDUL RATURIEnter the n:4' followed by 'Strong numbers are :1, 2,'.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int factorial(int f) {
4     int mul = 1;
5     for (int i = 1; i <= f; i++) {
6         mul = mul * i;
7     }
8     return mul;
9 }
10 int main() {
11     int fact = 1, sum = 0;
12     int n, r;
13     printf("SHARDUL RATURI");
14     printf("Enter the n:");
15     scanf("%d",&n);
16     printf("Strong numbers are :");
17     for (int i = 1; i <= n; i++) {
18         int k = i;
19         while (k != 0) {
20             r = k % 10;
21             fact = factorial(r);
22             k = k / 10;
23             sum = sum + fact;
24         }
25         if (sum == 1) {
26             printf("%d, ", i);
27         }
28         sum = 0;
29     }
30     return 0;
31 }
```

Output: /tmp/catGdRQW4R.o
SHARDUL RATURIEnter the n:4
Strong numbers are :1, 2, |

PROGRAM 55:- TO PRINT FIBONACCI SERIES UPTO n Terms

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
int n1=0,n2=1,n3,i,num;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter the number of elements:");
```

```
scanf("%d",&num);
```

```
printf("\n%d %d",n1,n2);
```

```
for(i=2;i<num;++i)
```

```
{
```

```
n3=n1+n2;
```

```
printf(" %d",n3);
```

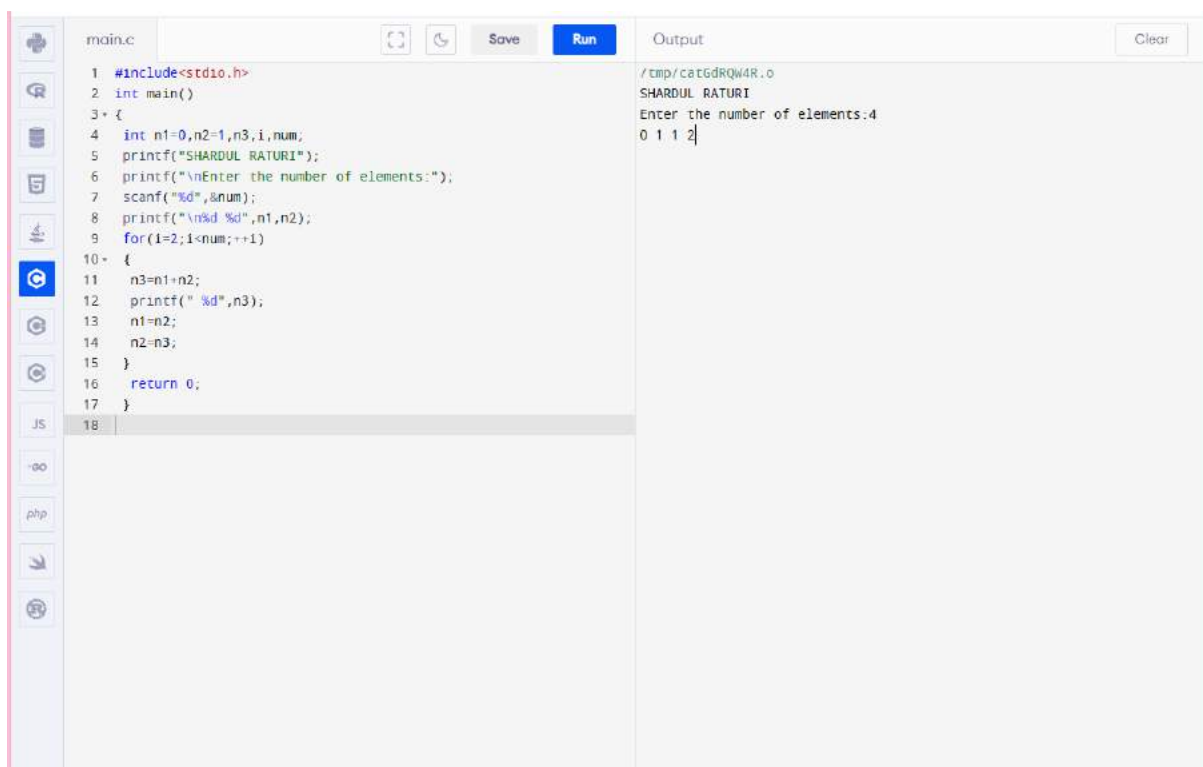
```
n1=n2;
```

```
n2=n3;
```

```
}
```

```
return 0;
```

```
}
```



```
main.c
1 #include<stdio.h>
2 int main()
3 {
4     int n1=0,n2=1,n3,i,num;
5     printf("SHARDUL RATURI");
6     printf("\nEnter the number of elements:");
7     scanf("%d",&num);
8     printf("\n%d %d",n1,n2);
9     for(i=2;i<num;++i)
10    {
11        n3=n1+n2;
12        printf(" %d",n3);
13        n1=n2;
14        n2=n3;
15    }
16    return 0;
17 }
18
```

Output

```
/tmp/catGdRQW4R.o
SHARDUL RATURI
Enter the number of elements:4
0 1 1 2
```

PROGRAM 56:- TO FIND ONEs COMPLEMENT OF A BINARY NUMBER.

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main()
{
    char binaryNumber[100], onesComplement[100];
    int counter, error=0, digitCount;
    printf("SHARDUL RATURI");
    printf("\nEnter a Binary Number\n");
    scanf("%s", binaryNumber);

    digitCount = strlen(binaryNumber);

    for(counter=0; counter < digitCount; counter++)
    {
        if(binaryNumber[counter]=='1')
        {
            onesComplement[counter] = '0';
        }
        else if(binaryNumber[counter]=='0')
        {
            onesComplement[counter] = '1';
        }
        else
        {
            printf("Error :( ");
            return 1;
        }
    }
}
```

```

    }
}

onesComplement[digitCount] = '\0';

printf("Ones Complement : %s", onesComplement);

return 0;
}

```

The screenshot shows an online C compiler interface with the following components:

- Header:** "Programiz" logo, "IFLYTEK Open Platform", "Translate API for Free", and buttons for "OPEN" and "Certification".
- File Explorer:** A sidebar on the left showing a file named "main.c".
- Code Editor:** The main area contains C code for finding the ones complement. The code includes headers for `<stdio.h>` and `<string.h>`, defines a `main` function, and uses a loop to iterate through the binary number string, flipping 0s to 1s and 1s to 0s. It also includes error handling for non-binary characters.
- Output:** A panel on the right shows the program's execution output:


```

/tmp/w5ktaVbwn1.c
SHARDUL RATURI
Enter a Binary Number
1000
Ones Complement : 0111
      
```

PROGAM 57:- TO FIND TWO's COMPLEMENT OF A BINARY NUMBER

```
#include<stdio.h>
```

```
#include<stdlib.h>
```

```
#define SIZE 8
```

```
int main()
```

```
{
```

```
    int i, carry = 1;
```

```
    char num[SIZE + 1], one[SIZE + 1], two[SIZE + 1];
```

```
printf("SHARDUL RATURI");  
printf("\nEnter the binary number");  
gets(num);  
for(i = 0; i < SIZE; i++)  
{  
    if(num[i] == '0')  
    {  
        one[i] = '1';  
    }  
    else if(num[i] == '1')  
    {  
        one[i] = '0';  
    }  
}  
one[SIZE] = '\0';  
printf("\nOnes' complement of binary number %s is %s",num,  
one);  
for(i = SIZE - 1; i >= 0; i--)  
{  
    if(one[i] == '1' && carry == 1)  
    {  
        two[i] = '0';  
    }  
    else if(one[i] == '0' && carry == 1)
```



```

{
    two[i] = '1';
    carry = 0;
}
else{
    two[i] = one[i];
}
}
two[SIZE] = '\0';
printf("Two's complement of binary number %s is %s", num, two);
return 0;
}

```

The screenshot displays an online C compiler interface. The code editor on the left contains the following C code:

```

1 #include<stdio.h>
2 #include<stdlib.h>
3 #define SIZE 8
4 int main()
5 {
6     int i, carry = 1;
7     char num[SIZE + 1], one[SIZE + 1], two[SIZE + 1];
8     printf("SHARDUL RATURI");
9     printf("\nEnter the binary number");
10    gets(num);
11    for(i = 0; i < SIZE; i++)
12    {
13        if(num[i] == '0')
14        {
15            one[i] = '1';
16        }
17        else if(num[i] == '1')
18        {
19            one[i] = '0';
20        }
21    }
22    one[SIZE] = '\0';
23    printf("\nOnes' complement of binary number %s is %s", num, one);
24    for(i = SIZE - 1; i >= 0; i--)
25    {
26        if(one[i] == '1' && carry == 1)
27        {
28            two[i] = '0';
29        }
30        else if(one[i] == '0' && carry == 1)
31        {
32            two[i] = '1';
33            carry = 0;
34        }
35        else{
36            two[i] = one[i];
37        }
38    }
39    two[SIZE] = '\0';
40    printf("Two's complement of binary number %s is %s", num, two);
41    return 0;
42 }

```

The output window on the right shows the following text:

```

/tmp/QPvbc6511e.o
SHARDUL RATURI
Enter the binary number1000
Ones' complement of binary number 1000 is 01118
Two's complement of binary number 1000 is 10000

```

PROGRAM 58:- TO CONVERT BINARY NUMBER TO OCTAL NUMBER

```
#include <stdio.h>
```

```
#include<math.h>
```

```
int main()
```

```
{
```

```
    int i, octal = 0, decimal = 0;
```

```
    long binary;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter the Binary Number = ");
```

```
    scanf("%ld", &binary);
```

```
    i = 0;
```

```
    while(binary != 0)
```

```
    {
```

```
        decimal = decimal + (binary % 10) * pow(2, i);
```

```
        i++;
```

```
        binary = binary/10;
```

```
    }
```

```
    i = 1;
```

```
    while(decimal != 0)
```

```
    {
```

```
        octal = octal + (decimal % 8) * i;
```

```
        decimal = decimal / 8;
```

```
        i = i * 10;
```

```
}
```

```
printf("The octal Value = %d\n", octal);
```

```
return 0;
```



```
main.c
1 #include <stdio.h>
2
3 #include <math.h>
4
5
6 int main()
7 {
8     int i, octal = 0, decimal = 0;
9     long binary;
10    printf("SHARDUL RATURI");
11    printf("\nEnter the Binary Number = ");
12    scanf("%ld", &binary);
13
14    i = 0;
15    while(binary != 0)
16    {
17        decimal = decimal + (binary % 10) * pow(2, i);
18        i++;
19        binary = binary/10;
20    }
21
22    i = 1;
23    while(decimal != 0)
24    {
25        octal = octal + (decimal % 8) * i;
26        decimal = decimal / 8;
27        i = i * 10;
28    }
29    printf("The octal Value = %d\n", octal);
30    return 0;
31 }
```

Output

```
/tmp/QPvbC6511e.o
SHARDUL RATURI
Enter the Binary Number = 10000
The octal Value = 20
```

PROGRAM 59:- TO CONVERT BINARY NUMBER TO DECIMAL NUMBER

```
#include <stdio.h>
```


PROGRAM 62:- TO CONVERT OCTAL NUMBER TO BINARY NUMBER

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main()
```

```
{
```

```
    int i, octal, decimal = 0;
```

```
    long binary = 0;
```

```
    i = 0;
```

```
    printf("Enter the Octal Number = ");
```

```
    scanf("%d",&octal);
```

```
    while(octal != 0)
```

```
    {
```

```
        decimal = decimal + (octal % 10) * pow(8, i);
```

```
        i++;
```

```

        octal = octal / 10;
    }
    i = 1;
    while(decimal != 0)
    {
        binary += ((decimal % 2) * i);
        decimal = decimal / 2;
        i = i * 10;
    }

    printf("The Binay Value = %ld\n", binary);

```

The screenshot shows the Programiz online C compiler interface. The code in the editor is as follows:

```

1  #include <stdio.h>
2
3  #include <math.h>
4
5
6  int main()
7  {
8      int i, octal, decimal = 0;
9      long binary = 0;
10     i = 0;
11
12     printf("Enter the Octal Number = ");
13     scanf("%d", &octal);
14
15     while(octal != 0)
16     {
17         decimal = decimal + (octal % 10) * pow
18             (8, i);
19         i++;
20         octal = octal / 10;
21     }
22     i = 1;
23     while(decimal != 0)
24     {
25         binary += ((decimal % 2) * i);
26         decimal = decimal / 2;
27         i = i * 10;
28     }
29     printf("The Binay Value = %ld\n", binary);
30     return 0;
31 }

```

The output window shows the following text:

```

/tmp/QPvbc6511e.o
Enter the Octal Number = 15
The Binay Value = 1101

```

PROGRAM 63:- TO CONERT OCTAL NUMBER TO DECIMAL NUMBER

```
#include <stdio.h>
```

```
#include <math.h>

int main()
{
    int octal, decimal = 0;
    int i = 0;
    printf("    \n");
    printf("Enter the Octal Number = ");
    scanf("%d",&octal);

    while(octal != 0)
    {
        decimal = decimal + (octal % 10) * pow(8, i++);
        octal = octal / 10;
    }

    printf("The Decimal Value = %d\n", decimal);
    return 0;
}
```

Programiz
C Online Compiler
IFLYTEK Open Platform
Translate API for Free
OPEN >
C Certification >

```
main.c  
1 #include <stdio.h>  
2  
3 #include <math.h>  
4  
5 int main()  
6 {  
7     int octal, decimal = 0;  
8     int i = 0;  
9     printf("\n");  
10    printf("Enter the Octal Number = ");  
11    scanf("%d", &octal);  
12  
13    while(octal != 0)  
14    {  
15        decimal = decimal + (octal % 10) * pow  
16        (8, i++);  
17        octal = octal / 10;  
18    }  
19    printf("The Decimal Value = %d\n", decimal);  
20    return 0;  
21 }  
22  
23
```

Output
/tmp/QPvbC6511e.o
Enter the Octal Number = 13
The Decimal Value = 11
Clear

PROGRAM 65:- TO CONVERT DECIMAL NUMBER TO BINARY NUMBER

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    long long decimal, tempDecimal, binary;
```

```
    int rem, place = 1;
```

```
    binary = 0;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter any decimal number: ");
```

```
    scanf("%lld", &decimal);
```

```
    tempDecimal = decimal;
```

```
    while(tempDecimal > 0)
```

```
    {
```

```
        rem = tempDecimal % 2;
```



```
binary = (rem * place) + binary;
```

```
tempDecimal /= 2;
```

```
place *= 10;
```

```
}
```

```
printf("Decimal number = %lld\n", decimal);
```

```
printf("Binary number = %lld", binary);
```

```
return 0;
```

```
}
```

The screenshot shows a web-based C compiler interface. At the top, there's a banner for 'Programiz' with the tagline 'TAKE CONTROL OF YOUR DATA'. Below the banner, the code editor shows a C program for converting a decimal number to binary. The code includes `<stdio.h>`, defines `main()`, and uses `long long` for `decimal`, `tempDecimal`, and `binary`. It prompts the user to enter a decimal number, reads it, and then uses a `while` loop to convert it to binary by repeatedly dividing by 2 and building the binary string. The output window shows the execution results for the input 13, displaying the decimal number and its binary equivalent, 1101.

```
main.c
1 #include <stdio.h>
2 int main()
3 {
4     long long decimal, tempDecimal, binary;
5     int rem, place = 1;
6     binary = 0;
7     printf("SHARDUL RATURI");
8     printf("\nEnter any decimal number: ");
9     scanf("%lld", &decimal);
10    tempDecimal = decimal;
11    while(tempDecimal > 0)
12    {
13        rem = tempDecimal % 2;
14        binary = (rem * place) + binary;
15        tempDecimal /= 2;
16        place *= 10;
17    }
18    printf("Decimal number = %lld\n", decimal);
19    printf("Binary number = %lld", binary);
20    return 0;
21 }
```

Output

```
/tmp/h668pv02vh.o
SHARDUL RATURI
Enter any decimal number: 13
Decimal number = 13
Binary number = 1101
```

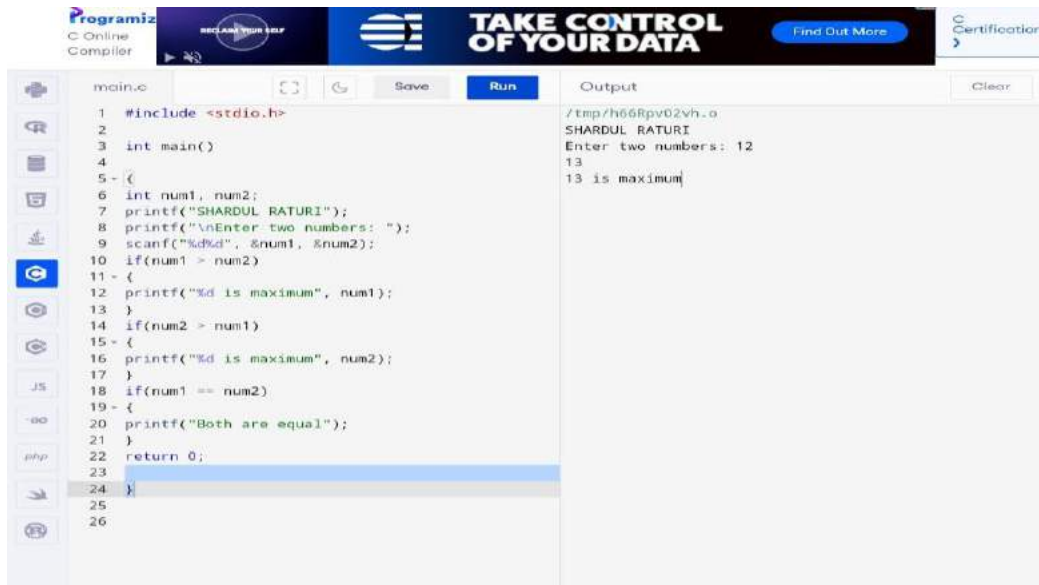
If...Else Exercises

PROGRAM 66. TO FIND MAXIMUM BETWEEN TWO NUMBERS.

```
#include <stdio.h>

int main()
{
    int num1, num2;
    printf("SHARDUL RATURI");
    printf("\nEnter two numbers: ");
    scanf("%d%d", &num1, &num2);
    if(num1 > num2)
    {
        printf("%d is maximum", num1);
    }
    if(num2 > num1)
    {
        printf("%d is maximum", num2);
    }
    if(num1 == num2)
    {
        printf("Both are equal");
    }
    return 0;
```

}



The screenshot shows the Programiz C Online Compiler interface. The code in main.c is as follows:

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int num1, num2;
6     printf("SHARDUL RATURI");
7     printf("\nEnter two numbers: ");
8     scanf("%d%d", &num1, &num2);
9     if(num1 > num2)
10    {
11        printf("%d is maximum", num1);
12    }
13    if(num2 > num1)
14    {
15        printf("%d is maximum", num2);
16    }
17    if(num1 == num2)
18    {
19        printf("Both are equal");
20    }
21    return 0;
22 }
```

The output on the right shows the execution results:

```
/tmp/h66Rpv02vh.o
SHARDUL RATURI
Enter two numbers: 12
13
13 is maximum
```

PROGRAM 67. TO FIND MAXIMUM BETWEEN THREE NUMBERS.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int num1, num2, num3, max;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter three numbers: ");
```

```
scanf("%d%d%d", &num1, &num2, &num3);
```

```
if(num1 > num2)
```

```
{
```

```
if(num1 > num3)
```

```
{
```

```
max = num1;
```

```
}  
else  
{  
    max = num3;  
}  
}  
else  
{  
    if(num2 > num3)  
    {  
        max = num2;  
    }  
    else  
    {  
        max = num3;  
    }  
}  
printf("Maximum among all three numbers = %d", max);  
return 0;  
}
```

The screenshot shows the Programiz online C compiler interface. The code in the editor is as follows:

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int num1, num2, num3, max;
6     printf("SHARDUL RATURI");
7     printf("\nEnter three numbers: ");
8     scanf("%d%d%d", &num1, &num2, &num3);
9     if(num1 > num2)
10     {
11         if(num1 > num3)
12         {
13             max = num1;
14         }
15         else
16         {
17             max = num3;
18         }
19     }
20     else if(num2 > num3)
21     {
22         max = num2;
23     }
24     else
25     {
26         max = num3;
27     }
28     printf("Maximum among all three numbers = %d",
29           max);
30     return 0;
31 }
```

The output window on the right shows the following text:

```
/tmp/h66Rpv02v6.o
SHARDUL RATURI
Enter three numbers: 13
14
15
Maximum among all three numbers = 15
```

PROGRAM 68. TO CHECK WHETHER A NUMBER IS NEGATIVE, POSITIVE OR ZERO.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int num;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter any number: ");
```

```
scanf("%d", &num);
```

```
if(num > 0)
```

```
{
```

```
printf("Number is POSITIVE");
```

```
}
```

```
else if(num < 0)
```

```

{
printf("Number is NEGATIVE");
}

else

{
printf("Number is ZERO");
}

return 0;
}

```

The screenshot shows the Programiz C Online Compiler interface. The code editor contains a C program that checks if a number is positive, negative, or zero. The program prompts the user to enter a number, and the output shows the result for the input -15.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int num;
6     printf("SHARDUL RATURI");
7     printf("\nEnter any number: ");
8     scanf("%d", &num);
9     if(num > 0)
10    {
11        printf("Number is POSITIVE");
12    }
13    else if(num < 0)
14    {
15        printf("Number is NEGATIVE");
16    }
17    else
18    {
19        printf("Number is ZERO");
20    }
21    return 0;
22 }
23
24
25

```

The output of the program is:

```

/tmp/h66Rpv02vh.o
SHARDUL RATURI
Enter any number: -15
Number is NEGATIVE

```

PROGRAM 69. TO CHECK WHETHER A NUMBER IS DIVISIBLE BY 5 AND 11 OR NOT.

```
#include <stdio.h>
```

```
int main()
```

```

{
int num;

printf("SHARDUL RATURI");

printf("\nEnter any number: ");

scanf("%d", &num);

if((num % 5 == 0) && (num % 11 == 0))
{
printf("Number is divisible by 5 and 11");
}
else
{
printf("Number is not divisible by 5 and 11");
}

return 0;
}

```



The screenshot shows the Programiz C Online Compiler interface. The code editor on the left contains the following C code:

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int num;
6     printf("SHARDUL RATURI");
7     printf("\nEnter any number: ");
8     scanf("%d", &num);
9     if((num % 5 == 0) && (num % 11 == 0))
10    {
11        printf("Number is divisible by 5 and 11");
12    }
13    else
14    {
15        printf("Number is not divisible by 5 and 11");
16    }
17    return 0;
18 }
19
20
21

```

The output window on the right shows the following output:

```

/tmp/h66Rpv02vh.o
SHARDUL RATURI
Enter any number: 18
Number is not divisible by 5 and 11

```

PROGRAM 70. TO CHECK WHETHER A NUMBER IS EVEN OR ODD.

```
#include <stdio.h>

int main()
{
    int num;

    printf("SHARDUL RATURI");
    printf("\nEnter any number to check even or odd: ");
    scanf("%d", &num);
    if(num % 2 == 0)
    {
        printf("Number is Even.");
    }
    else
    {
        printf("Number is Odd.");
    }
    return 0;
}
```




The screenshot shows the Programiz Online Compiler interface. At the top, it says "Programiz iFLYTEK Open Platform" and "Translate API for Free". There is an "OPEN >" button and a "Certification" link. The main area is divided into two panes. The left pane, titled "main.c", contains the following C code:

```
1 #include <stdio.h>
2
3 int main()
4 {
5     {
6     int num;
7     printf("SHARDUL RATURI");
8     printf("\nEnter any number to check even or odd
9         : ");
10    scanf("%d", &num);
11    if(num % 2 == 0)
12    {
13        printf("Number is Even.");
14    }
15    else
16    {
17        printf("Number is Odd.");
18    }
19    return 0;
20 }
21
```

The right pane, titled "Output", shows the execution results:

```
/tmp/h66Rpv02vh.o
SHARDUL RATURI
Enter any number to check even or odd: 16
Number is Even.
```

**TO CHECK WHETHER A CHARACTER IS
ALPHABET OR NOT**

#include <stdio.h>

int main()

{

char ch;

printf("shardul raturi");

printf("\nEnter any character: ");

```
scanf("%c", &ch);  
if((ch >= 'a' && ch <= 'z') || (ch >= 'A' &&  
ch <= 'Z'))  
{  
printf("Character is an ALPHABET.");  
}  
else  
{  
printf("Character is NOT ALPHABET.");  
}  
return 0;  
  
}
```

The screenshot shows a web browser with two tabs: 'WhatsApp' and 'Online C Compiler'. The address bar shows 'programiz.com/c-programming/online-compiler/'. The page has a header with 'Programiz C Online Compiler' and a 'Decathlon Dehradun' advertisement. The main area is divided into a code editor on the left and an output window on the right. The code editor contains a C program that prints 'shardul raturi', prompts for a character, and checks if it's an alphabet. The output window shows the program's execution: 'shardul raturi', 'Enter any character: e', and 'Character is an ALPHABET.' The Windows taskbar is visible at the bottom.

PROGRAM 73. TO INPUT ANY ALPHABET AND CHECK WHETHER IT IS VOWEL OR CONSONANT.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
char ch;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter any character: ");
```

```
scanf("%c", &ch);
```

```
if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u' ||
```

```
ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U')
```

```

{
printf("%c' is Vowel.", ch);
}

else if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
{
printf("%c' is Consonant.", ch);
}

else
{
printf("%c' is not an alphabet.", ch);
}

return 0;
}

```

The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The page title is "Programiz C Online Compiler". There is a "Certification" button in the top right. The main area is divided into two panes: "main.c" on the left and "Output" on the right. The "main.c" pane contains the following code:

```

1 #include <stdio.h>
2 int main()
3 {
4     char ch;
5     printf("SHARDUL RATURE");
6     printf("\nEnter any character: ");
7     scanf("%c", &ch);
8     if(ch>='a' || ch>='e' || ch>='i' || ch>='o' || ch>='u' ||
9     ch>='A' || ch>='E' || ch>='I' || ch>='O' || ch>='U')
10    {
11        printf("%c' is Vowel.", ch);
12    }
13    else if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
14    {
15        printf("%c' is Consonant.", ch);
16    }
17    else
18    {
19        printf("%c' is not an alphabet.", ch);
20    }
21    return 0;
22 }
23

```

The "Output" pane shows the following text:

```

/rtp/r0uQjJHtZ.0
SHARDUL RATURE
Enter any character: A
'A' is Vowel.

```

The Windows taskbar at the bottom shows the date as 15-12-2023 and the temperature as 9°C Sunny.

PROGRAM 74. TO INPUT ANY CHARACTER AND CHECK WHETHER IT IS ALPHABET, DIGIT OR SPECIAL CHARACTER.

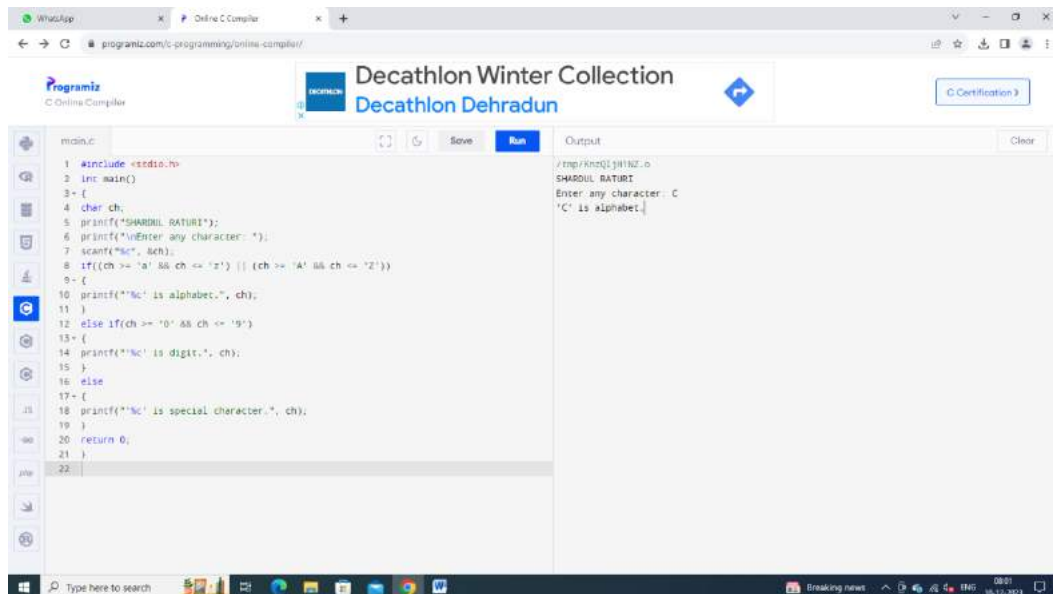
```
#include <stdio.h>

int main()
{
    char ch;

    printf("SHARDUL RATURI");
    printf("\nEnter any character: ");
    scanf("%c", &ch);
    if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
    {
        printf("'%c' is alphabet.", ch);
    }
    else if(ch >= '0' && ch <= '9')
    {
        printf("'%c' is digit.", ch);
    }
    else
    {
        printf("'%c' is special character.", ch);
    }
}
```

return 0;

}



PROGRAM 75. TO CHECK WHETHER A CHARACTER IS UPPERCASE OR LOWERCASE ALPHABET.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
char ch;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter any character: ");
```

```
scanf("%c", &ch);
```

```
if(ch >= 'A' && ch <= 'Z')
```

```
{
```

```
printf("'%c' is uppcase alphabet.", ch);
```

```

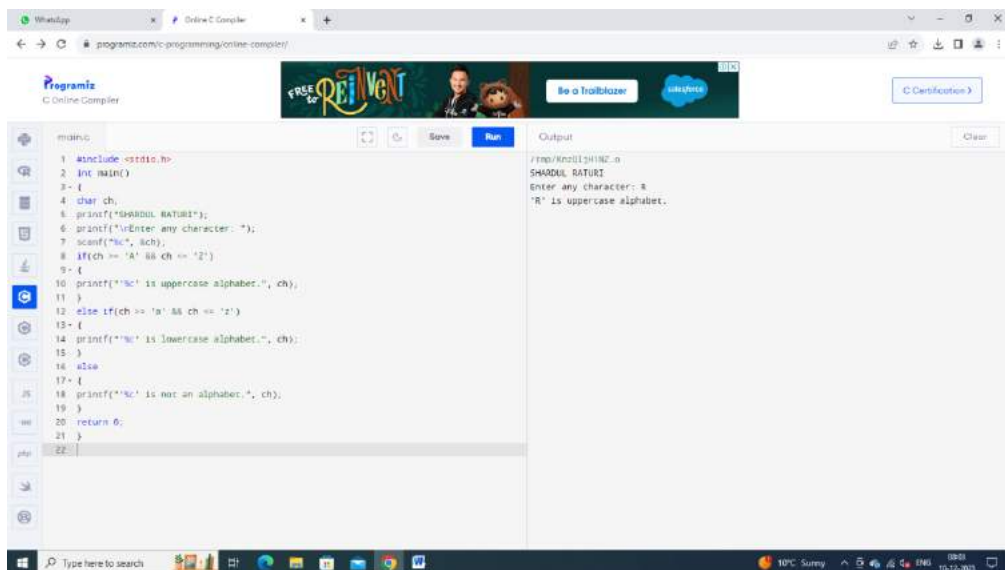
}

else if(ch >= 'a' && ch <= 'z')
{
printf("%c' is lowercase alphabet.", ch);
}

else
{
printf("%c' is not an alphabet.", ch);
}

return 0;
}

```



The screenshot shows the Programiz Online C Compiler interface. The code editor on the left contains the following C program:

```

1 #include <stdio.h>
2 int main()
3 {
4     char ch;
5     printf("SHARAD RATURI");
6     printf("Enter any character: ");
7     scanf("%c", &ch);
8     if(ch >= 'A' && ch <= 'Z')
9     {
10        printf("%c' is uppercase alphabet.", ch);
11    }
12    else if(ch >= 'a' && ch <= 'z')
13    {
14        printf("%c' is lowercase alphabet.", ch);
15    }
16    else
17    {
18        printf("%c' is not an alphabet.", ch);
19    }
20    return 0;
21 }

```

The output window on the right shows the following output:

```

/tmp/Kn0ll1jH1N2.o
SHARAD RATURI
Enter any character: R
R is uppercase alphabet.

```

PROGRAM 76. TO INPUT WEEK NUMBER AND PRINT WEEK DAY.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int week;
printf("SHARDUL RATURI");
printf("\nEnter week number (1-7): ");
scanf("%d", &week);
if(week == 1)
{
printf("Monday");
}
else if(week == 2)
{
printf("Tuesday");
}
else if(week == 3)
{
printf("Wednesday");
}
else if(week == 4)
{
printf("Thursday");
}
else if(week == 5)
{
printf("Friday");
}
```



```

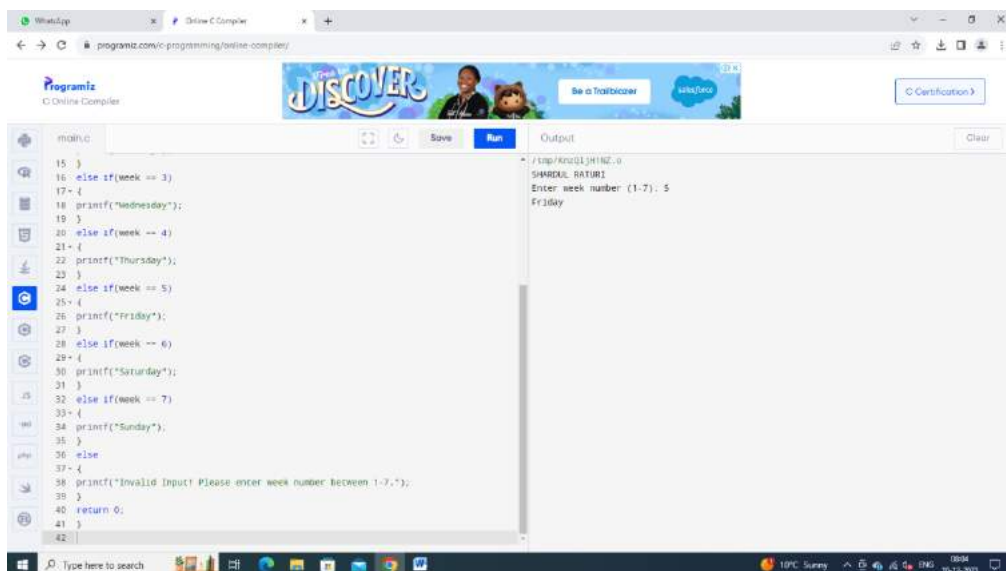
else if(week == 6)
{
printf("Saturday");
}

else if(week == 7)
{
printf("Sunday");
}

else
{
printf("Invalid Input! Please enter week number between 1-7.");
}

return 0;
}

```



The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler`. The page features a banner for "Programiz DISCOVER" and a "Be a Trailblazer" button. Below the banner, the C code from the previous block is pasted into the editor. The code defines a function `main` that takes a week number as input and prints the corresponding day of the week. The code is as follows:

```

15 }
16 else if(week == 3)
17 {
18     printf("Wednesday");
19 }
20 else if(week == 4)
21 {
22     printf("Thursday");
23 }
24 else if(week == 5)
25 {
26     printf("Friday");
27 }
28 else if(week == 6)
29 {
30     printf("Saturday");
31 }
32 else if(week == 7)
33 {
34     printf("Sunday");
35 }
36 else
37 {
38     printf("Invalid Input! Please enter week number between 1-7.");
39 }
40 return 0;
41 }
42

```

The output window on the right shows the following text:

```

/tmp/Hz0L1H1N2.o
SHADEL: SATUR1
Enter week number (1-7): 5
Friday

```

The Windows taskbar at the bottom shows the system clock as 10:00 PM on 10-12-2023.

PROGRAM 77. TO INPUT MONTH NUMBER AND PRINT NUMBER OF DAYS IN THAT MONTH.

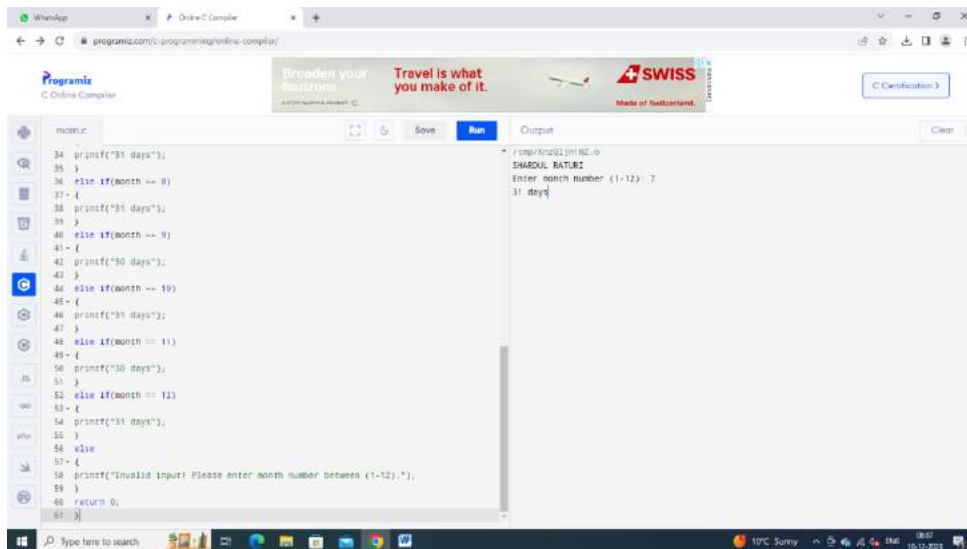
```
#include <stdio.h>

int main()
{
    int month;

    printf("SHARDUL RATURI");
    printf("\nEnter month number (1-12): ");
    scanf("%d", &month);
    if(month == 1)
    {
        printf("31 days");
    }
    else if(month == 2)
    {
        printf("28 or 29 days");
    }
    else if(month == 3)
    {
        printf("31 days");
    }
    else if(month == 4)
```

```
{  
printf("30 days");  
}  
else if(month == 5)  
{  
printf("31 days");  
}  
else if(month == 6)  
{  
printf("30 days");  
}  
else if(month == 7)  
{  
printf("31 days");  
}  
else if(month == 8)  
{  
printf("31 days");  
}  
else if(month == 9)  
{  
printf("30 days");  
}  
else if(month == 10)
```

```
{  
printf("31 days");  
}  
else if(month == 11)  
{  
printf("30 days");  
}  
else if(month == 12)  
{  
printf("31 days");  
}  
else  
{  
printf("Invalid input! Please enter month number between (1-  
12).");  
}  
return 0;  
}
```



PROGRAM 78. TO COUNT TOTAL NUMBER OF NOTES IN GIVEN AMOUNT.

```
#include <stdio.h>
```

```
int main()
```

{

```
int amount;
```

```
int note500, note100, note50, note20, note10, note5, note2,  
note1; note500 = note100 = note50 = note20 = note10 = note5 =  
note2 = note1 = 0;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter amount: ");
```

```
scanf("%d", &amount);
```

```
if(amount >= 500)
```

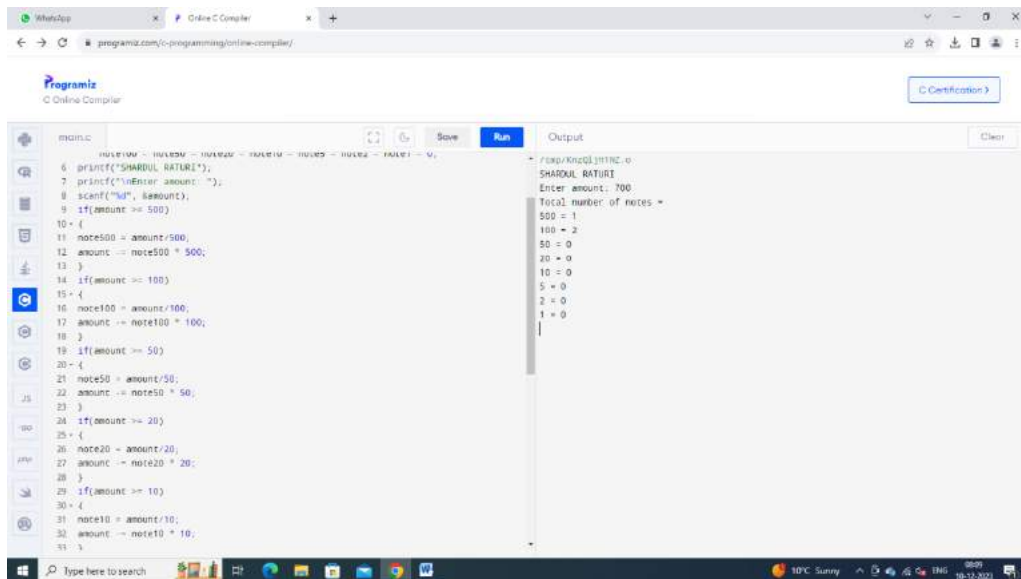
{

```
note500 = amount/500;
```

```
amount -= note500 * 500;
```

```
}  
if(amount >= 100)  
{  
    note100 = amount/100;  
    amount -= note100 * 100;  
}  
if(amount >= 50)  
{  
    note50 = amount/50;  
    amount -= note50 * 50;  
}  
if(amount >= 20)  
{  
    note20 = amount/20;  
    amount -= note20 * 20;  
}  
if(amount >= 10)  
{  
    note10 = amount/10;  
    amount -= note10 * 10;  
}  
if(amount >= 5)  
{  
    note5 = amount/5;
```

```
amount -= note5 * 5;
}
if(amount >= 2)
{
note2 = amount /2;
amount -= note2 * 2;
}
if(amount >= 1)
{
note1 = amount;
}
printf("Total number of notes = \n");
printf("500 = %d\n", note500);
printf("100 = %d\n", note100);
printf("50 = %d\n", note50);
printf("20 = %d\n", note20);
printf("10 = %d\n", note10);
printf("5 = %d\n", note5);
printf("2 = %d\n", note2);
printf("1 = %d\n", note1);
return 0;
}
```



PROGRAM 79. TO INPUT ANGLES OF A TRIANGLE AND CHECK WHETHER TRIANGLE IS VALID OR NOT.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int angle1, angle2, angle3, sum;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter three angles of triangle: \n");
```

```
scanf("%d%d%d", &angle1, &angle2, &angle3);
```

```
sum = angle1 + angle2 + angle3;
```

```
if(sum == 180 && angle1 > 0 && angle2 > 0 && angle3 > 0)
```

```
{
```

```
printf("Triangle is valid.");
```

```
}
```



```

else
{
printf("Triangle is not valid.");
}

return 0;
}

```

The screenshot displays the Programiz Online C Compiler interface. The code editor on the left contains a C program that prompts the user to enter three angles of a triangle. The program calculates the sum of these angles and checks if it equals 180 degrees. If the sum is 180, it prints "Triangle is valid."; otherwise, it prints "Triangle is not valid.". The output window on the right shows the execution results, including the user's input and the program's output.

```

main.c
1 #include <stdio.h>
2 int main()
3 {
4     int angle1, angle2, angle3, sum;
5     printf("SHARDUL RATURI");
6     printf("\nEnter three angles of triangle: \n");
7     scanf("%d%d%d", &angle1, &angle2, &angle3);
8     sum = angle1 + angle2 + angle3;
9     if(sum == 180 && angle1 > 0 && angle2 > 0 && angle3 > 0)
10     {
11         printf("Triangle is valid.");
12     }
13     else
14     {
15         printf("Triangle is not valid.");
16     }
17     return 0;
18 }
19
Output
/tmp/KrzQ19THZ.o
SHARDUL RATURI
Enter three angles of triangle:
50
60
50
Triangle is valid.

```

PROGRAM 80. TO INPUT ALL SIDES OF A TRIANGLE AND CHECK WHETHER TRIANGLE IS VALID OR NOT.

```

#include <stdio.h>

int main()
{
int side1, side2, side3;

printf("SHARDUL RATURI");

printf("\nEnter three sides of triangle: \n");

```

```
scanf("%d%d%d", &side1, &side2, &side3);
if((side1 + side2) > side3)
{
    if((side2 + side3) > side1)
    {
        if((side1 + side3) > side2)
        {
            printf("Triangle is valid.");
        }
    }
    else
    {
        printf("Triangle is not valid.");
    }
}
else
{
    printf("Triangle is not valid.");
}
}
else
{
    printf("Triangle is not valid.");
}
}
return 0;
```

}

PROGRAM 81. TO CHECK WHETHER THE TRIANGLE IS EQUILATERAL, ISOSCELES OR SCALENE TRIANGLE.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int side1, side2, side3;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter three sides of triangle: ");
```

```
scanf("%d%d%d", &side1, &side2, &side3);
```

```
if(side1==side2 && side2==side3)
```

```
{
```

```
printf("Equilateral triangle.");
```

```
}
```

```

else if(side1==side2 || side1==side3 || side2==side3)
{
printf("Isosceles triangle.");
}
else
{
printf("Scalene triangle.");
}

return 0;
}

```

The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The page displays the Programiz C Online Compiler interface. On the left, the source code for `main.c` is shown, containing logic to identify triangle types based on three sides. The code includes headers `<stdio.h>` and `<math.h>`, and uses `printf` and `scanf` for input/output. The logic checks for equilateral, isosceles, and scalene triangles. On the right, the 'Output' pane shows the program's execution: it prompts for three sides, receives the input '12 25', and outputs 'Scalene triangle.'.

PROGRAM 82. TO FIND ALL ROOTS OF A QUADRATIC EQUATION.

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int main()
```

```
{
float a, b, c;

float root1, root2, imaginary;

float discriminant;

printf("SHARDUL RATURI");

printf("\nEnter values of a, b, c of quadratic equation (aX^2 + bX +
c): ");

scanf("%f%f%f", &a, &b, &c);

discriminant = (b * b) - (4 * a * c);

if(discriminant > 0)
{
root1 = (-b + sqrt(discriminant)) / (2*a);
root2 = (-b - sqrt(discriminant)) / (2*a);

printf("Two distinct and real roots exists: %.2f and %.2f", root1,
root2);
}
else if(discriminant == 0)
{
root1 = root2 = -b / (2 * a);

printf("Two equal and real roots exists: %.2f and %.2f", root1,
root2);
}
else if(discriminant < 0)
{
root1 = root2 = -b / (2 * a);
```

```

imaginary = sqrt(-discriminant) / (2 * a);

printf("Two distinct complex roots exists: %.2f + i%.2f and %.2f - i%.2f", root1, imaginary, root2, imaginary);

}

return 0;

}

```

The screenshot shows a web browser window with an online C compiler. The code in the editor is as follows:

```

1 #include <stdio.h>
2
3 float root1, root2, imaginary;
4
5 printf("SHARDUL RATURI");
6 printf("\nEnter values of a, b, c of quadratic equation (ax^2 + bx + c) = 0: ");
7 scanf("%f%f%f", &a, &b, &c);
8
9 discriminant = (b * b) - (4 * a * c);
10
11 if(discriminant > 0)
12 {
13     root1 = (-b + sqrt(discriminant)) / (2 * a);
14     root2 = (-b - sqrt(discriminant)) / (2 * a);
15     printf("Two distinct and real roots exists: %.2f and %.2f", root1, root2);
16 }
17
18 else if(discriminant == 0)
19 {
20     root1 = root2 = -b / (2 * a);
21     printf("Two equal and real roots exists: %.2f and %.2f", root1, root2);
22 }
23
24 else if(discriminant < 0)
25 {
26     root1 = root2 = -b / (2 * a);
27     imaginary = sqrt(-discriminant) / (2 * a);
28     printf("Two distinct complex roots exists: %.2f + i%.2f and %.2f - i%.2f", root1, imaginary, root2, imaginary);
29 }
30
31 return 0;

```

The output window shows the following text:

```

SHARDUL RATURI
Enter values of a, b, c of quadratic equation (ax^2 + bx + c) = 0:
5
2
Two distinct complex roots exists: -0.62 + 10.33i and -0.62 - 10.33i

```

PROGRAM 83. TO CALCULATE PROFIT OR LOSS.

```

#include <stdio.h>

int main()
{
    int cp, sp, amt;

    printf("SHARDUL RATURI");

    printf("\nEnter cost price: ");

    scanf("%d", &cp);

    printf("Enter selling price: ");

```

```
scanf("%d", &sp);  
if(sp > cp)  
{  
    amt = sp - cp;  
    printf("Profit = %d", amt);  
}  
else if(cp > sp)  
{  
    amt = cp - sp;  
    printf("Loss = %d", amt);  
}  
else  
{  
    printf("No Profit No Loss.");  
}  
return 0;  
}
```

The screenshot shows a web browser window with the 'Programiz Online C Compiler' interface. The code in the editor is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     int cp, sp, amt;
5     printf("SHARDUL RATHORI");
6     printf("\nEnter cost price: ");
7     scanf("%d", &cp);
8     printf("\nEnter selling price: ");
9     scanf("%d", &sp);
10    if(sp > cp)
11    {
12        amt = sp - cp;
13        printf("Profit = %d", amt);
14    }
15    else if(cp > sp)
16    {
17        amt = cp - sp;
18        printf("Loss = %d", amt);
19    }
20    else
21    {
22        printf("No Profit No Loss.");
23    }
24    return 0;
25 }
```

The output window on the right shows the following text:

```
/tmp/XXXX/jhNtE.o
SHARDUL RATHORI
Enter cost price: 1500
Enter selling price: 1800
Profit = 300
```

PROGRAM 84. TO INPUT MARKS OF FIVE SUBJECTS PHYSICS, CHEMISTRY, BIOLOGY, MATHEMATICS AND COMPUTER.

CALCULATE PERCENTAGE AND GRADE ACCORDING TO FOLLOWING:

PERCENTAGE \geq 90% : GRADE A

PERCENTAGE \geq 80% : GRADE B

PERCENTAGE \geq 70% : GRADE C

PERCENTAGE \geq 60% : GRADE D

PERCENTAGE \geq 40% : GRADE E

PERCENTAGE $<$ 40% : GRADE F

#include <stdio.h>


```
int main()
{
int phy, chem, bio, math, comp;
float per;
printf("SHARDUL RATURI");
printf("\nEnter five subjects marks: ");
scanf("%d%d%d%d%d", &phy, &chem, &bio, &math, &comp);
per = (phy + chem + bio + math + comp) / 5.0;
printf("Percentage = %.2f\n", per);
if(per >= 90)
{
printf("Grade A");
}
else if(per >= 80)
{
printf("Grade B");
}
else if(per >= 70)
{
printf("Grade C");
}
else if(per >= 60)
{
printf("Grade D");
```

```

}
else if(per >= 40)
{
printf("Grade E");
}
else
{
printf("Grade F");
}
return 0;
}

```

The screenshot displays a web-based C compiler interface. The code editor on the left contains a C program that calculates a grade based on a percentage input. The program uses a series of if-else statements to determine the grade: A (90% and above), B (80% and above), C (70% and above), D (60% and above), E (40% and above), and F (below 40%). The output window on the right shows the execution results, including the user's name (SHARUOL RATORI), the input marks (91), the calculated percentage (78.40), and the resulting grade (C).

```

main.c
10 printf("Percentage = %.2f\n", per);
11 if(per >= 90)
12 {
13 printf("Grade A");
14 }
15 else if(per >= 80)
16 {
17 printf("Grade B");
18 }
19 else if(per >= 70)
20 {
21 printf("Grade C");
22 }
23 else if(per >= 60)
24 {
25 printf("Grade D");
26 }
27 else if(per >= 40)
28 {
29 printf("Grade E");
30 }
31 else
32 {
33 printf("Grade F");
34 }
35 return 0;
36 }
37

```

Output

```

//tmp/kkz01jht102.o
SHARUOL RATORI
Enter five subjects marks: 91
87
56
89
69
Percentage = 78.40
Grade C

```

PROGRAM 85. TO INPUT BASIC SALARY OF AN EMPLOYEE AND CALCULATE ITS GROSS SALARY ACCORDING TO FOLLOWING:

BASIC SALARY <= 10000 : HRA = 20%,

DA = 80% BASIC SALARY <= 20000 : HRA = 25%,

DA = 90% BASIC SALARY > 20000 : HRA = 30%,

DA = 95%

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
float basic, gross, da, hra;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter basic salary of an employee: ");
```

```
scanf("%f", &basic);
```

```
if(basic <= 10000)
```

```
{
```

```
da = basic * 0.8;
```

```
hra = basic * 0.2;
```

```
}
```

```
else if(basic <= 20000)
```

```
{
```

```
da = basic * 0.9;
```

```
hra = basic * 0.25;
```

```

}

else
{

da = basic * 0.95;

hra = basic * 0.3;

}

gross = basic + hra + da;

printf("GROSS SALARY OF EMPLOYEE = %.2f", gross);

return 0;

}

```

The screenshot shows a web browser window with the 'Programiz C Online Compiler' interface. The code editor on the left contains a C program that calculates the gross salary of an employee based on their basic salary. The program uses conditional logic to apply different Dearness Allowance (da) and House Rent Allowance (hra) rates based on the basic salary. The output window on the right shows the program's execution, where the user has entered a basic salary of 15000, and the program has calculated a gross salary of 32250.00.

```

main.c
1 #include <stdio.h>
2 int main()
3 {
4     float basic, gross, da, hra;
5     printf("SHARDUL RATHI");
6     printf("\nEnter basic salary of an employee: ");
7     scanf("%f", &basic);
8     if(basic <= 10000)
9     {
10        da = basic * 0.8;
11        hra = basic * 0.2;
12    }
13    else if(basic <= 20000)
14    {
15        da = basic * 0.9;
16        hra = basic * 0.25;
17    }
18    else
19    {
20        da = basic * 0.95;
21        hra = basic * 0.3;
22    }
23    gross = basic + hra + da;
24    printf("GROSS SALARY OF EMPLOYEE = %.2f", gross);
25    return 0;
26 }
27
Output
/rtp/kndlg19f12:0
SHARDUL RATHI
Enter basic salary of an employee: 15000
GROSS SALARY OF EMPLOYEE = 32250.00

```

PROGRAM 86. TO INPUT ELECTRICITY UNIT CHARGES AND CALCULATE TOTAL ELECTRICITY BILL ACCORDING TO THE GIVEN CONDITION:

FOR FIRST 50 UNITS RS. 0.50/UNIT

FOR NEXT 100 UNITS RS. 0.75/UNIT

FOR NEXT 100 UNITS RS. 1.20/UNIT

FOR UNIT ABOVE 250 RS. 1.50/UNIT

**AN ADDITIONAL SURCHARGE OF 20% IS
ADDED TO THE BILL**

```
#include <stdio.h>

int main()
{
    int unit;
    float amt, total_amt, sur_charge;
    printf("SHARDUL RATURI");
    printf("\nEnter total units consumed: ");
    scanf("%d", &unit);
    if(unit <= 50)
    {
        amt = unit * 0.50;
    }
    else if(unit <= 150)
    {
        amt = 25 + ((unit-50) * 0.75);
    }
    else if(unit <= 250)
```

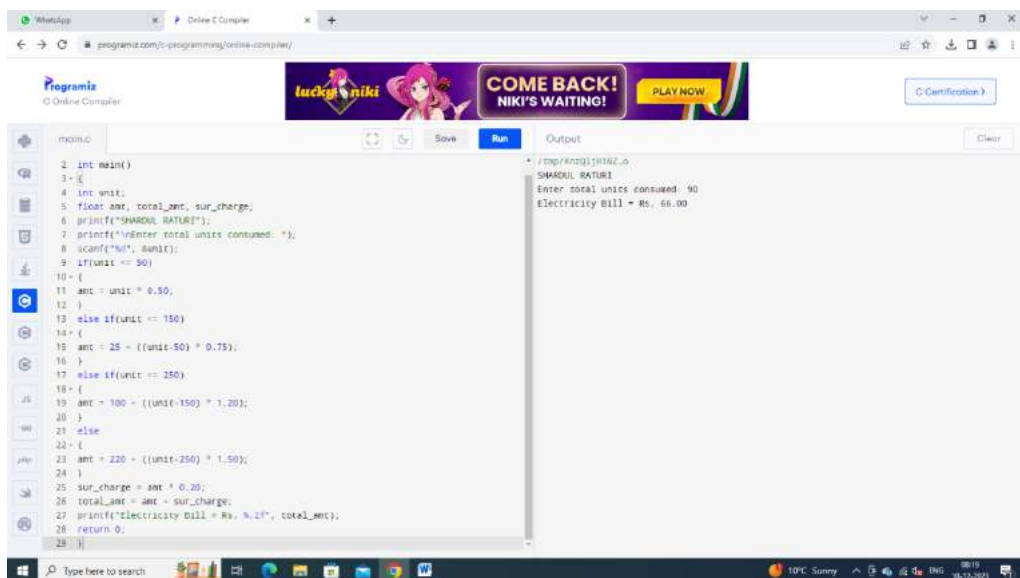
```

{
amt = 100 + ((unit-150) * 1.20);
}

else
{
amt = 220 + ((unit-250) * 1.50);
}

sur_charge = amt * 0.20;
total_amt = amt + sur_charge;
printf("Electricity Bill = Rs. %.2f", total_amt);
return 0;
}

```



The screenshot shows a web browser window with an online C compiler. The code from the previous block is pasted into the editor. The output window shows the following text:

```

/ tmp/kingj1htaz.o
SHARUL RATHI
Enter total units consumed: 90
Electricity Bill = Rs. 66.00

```

PROGRAM 87.TO CONVERT SPECIFIED DAYS INTO YEARS, WEEKS AND DAYS.

```
#include <stdio.h>
```

```

int main()
{
    int days, years, weeks;

    printf("SHARDUL RATURI");

    days = 1329;

    years = days/365;

    weeks = (days % 365)/7;

    days = days- ((years*365) + (weeks*7));

    printf("\nYears: %d\n", years);

    printf("Weeks: %d\n", weeks);

    printf("Days: %d \n", days);

    return 0;
}

```

The screenshot shows a web browser window with an online C compiler. The code is pasted into the editor and has been executed. The output window displays the results of the program's execution.

```

#include <stdio.h>
int main()
{
    int days, years, weeks;
    printf("SHARDUL RATURI");
    days = 1329;
    years = days/365;
    weeks = (days % 365)/7;
    days = days- ((years*365) + (weeks*7));
    printf("\nYears: %d\n", years);
    printf("Weeks: %d\n", weeks);
    printf("Days: %d \n", days);
    return 0;
}

```

Output:

```

/esp/heap01/sh192.0
SHARDUL RATURI
Years: 3
Weeks: 33
Days: 3

```

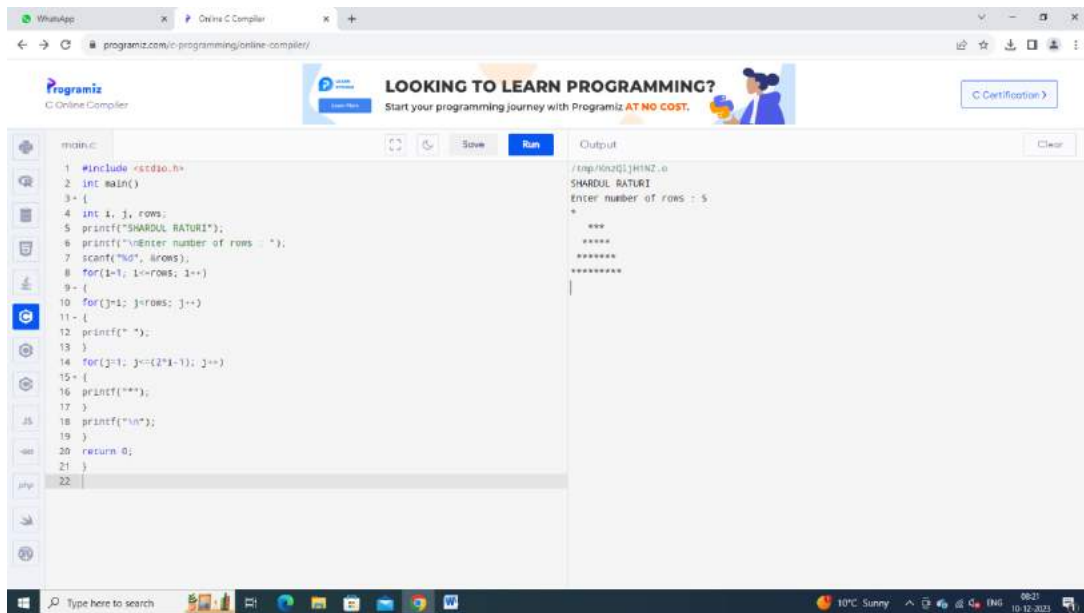
Pattern Exercises

STAR PATTERN PROGRAMS

PROGRAM 88. PYRAMID STAR PATTERN

```
#include <stdio.h>

int main()
{
    int i, j, rows;
    printf("SHARDUL RATURI");
    printf("\nEnter number of rows : ");
    scanf("%d", &rows);
    for(i=1; i<=rows; i++)
    {
        for(j=i; j<=rows; j++)
        {
            printf(" ");
        }
        for(j=1; j<=(2*i-1); j++)
        {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

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

Output

```
/tmp/hs02tj/jH1N2.o
SHARDUL RATURI
Enter number of rows : 5
*
***
*****
*****
*****
```

PROGRAM 89.HOLLOW PYRAMID STAR PATTERN

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int i, j, rows;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter number of rows : ");
```

```
scanf("%d", &rows);
```

```
for(i=1; i<=rows; i++)
```

```
{
```

```
for(j=i; j<=rows; j++)
```

```
{
```

```
printf(" ");
```

```
}
```

```

for(j=1; j<=(2*i-1); j++)
{
if(i==rows || j==1 || j==(2*i-1))
{
printf("*");
}
else
{
printf(" ");
}
}
printf("\n");
}
return 0;
}

```

The screenshot shows the Programiz Online C Compiler interface. The code editor on the left contains the following C program:

```

1 int main()
2 {
3     int i, j, rows;
4     printf("SHARDUL RATURI");
5     printf("\nEnter number of rows : ");
6     scanf("%d", &rows);
7     for(i=1; i<=rows; i++)
8     {
9         for(j=1; j<=rows; j++)
10        {
11            printf(" ");
12        }
13        for(j=1; j<=(2*i-1); j++)
14        {
15            if(i==rows || j==1 || j==(2*i-1))
16            {
17                printf("*");
18            }
19            else
20            {
21                printf(" ");
22            }
23        }
24        printf("\n");
25    }
26    return 0;
27 }
28
29

```

The output window on the right shows the following text:

```

/tmp/xyzQZjH1N2.o
SHARDUL RATURI
Enter number of rows : 5
*
 *
* *
* * *
*****

```

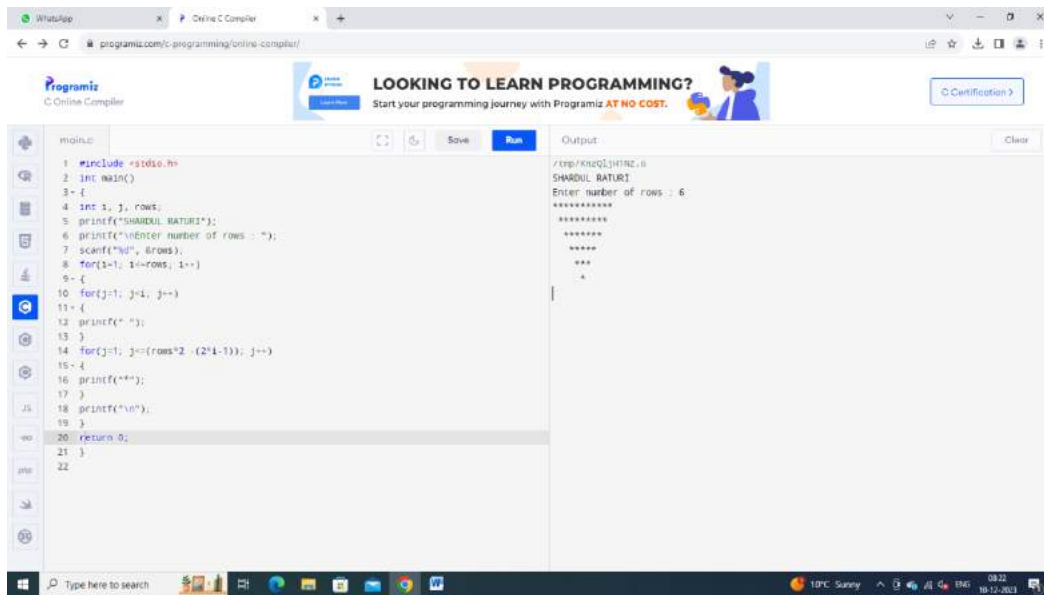
The bottom of the screenshot shows a Windows taskbar with the date and time as 10:10:2023.

PROGRAM 90. INVERTED PYRAMID STAR PATTERN

```
#include <stdio.h>

int main()
{
    int i, j, rows;

    printf("SHARDUL RATURI");
    printf("\nEnter number of rows : ");
    scanf("%d", &rows);
    for(i=1; i<=rows; i++)
    {
        for(j=1; j<i; j++)
        {
            printf(" ");
        }
        for(j=1; j<=(rows*2 -(2*i-1)); j++)
        {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     int i, j, rows;
5     printf("SHARDUL RATURI");
6     printf("\nEnter number of rows: ");
7     scanf("%d", &rows);
8     for(i=1; i<=rows; i++)
9     {
10        for(j=1; j<=i; j++)
11        {
12            printf(" ");
13        }
14        for(j=1; j<=(rows*2 - (2*i-1)); j++)
15        {
16            printf(" ");
17        }
18        printf("\n");
19    }
20    return 0;
21 }
```

Output:

```
/tmp/knz0jHtNZ.o
SHARDUL RATURI
Enter number of rows : 6
*****
*****
*****
***
A
```

PROGRAM 91.HOLLOW INVERTED STAR PATTERN

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int i, j, rows;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter number of rows: ");
```

```
scanf("%d", &rows);
```

```
for(i=1; i<=rows; i++)
```

```
{
```

```
for(j=1; j<i; j++)
```

```
{
```

```
printf(" ");
```

```
}
```

```

for(j=1; j<=(rows*2 - (2*i-1)); j++)
{
if(i==1 || j==1 || j==(rows*2 - (2*i - 1)))
{
printf("*");
}
else
{
printf(" ");
}
}
printf("\n");
}

return 0;
}

```

The screenshot shows a web browser window with the 'Online C Compiler' tab active. The code editor contains the following C program:

```

1 int main()
2 {
3     int i, j, rows;
4     printf("SHARDUL RATURI");
5     printf("\nEnter number of rows: ");
6     scanf("%d", &rows);
7     for(i=1; i<=rows; i++)
8     {
9         for(j=1; j<=(rows*2 - (2*i-1)); j++)
10        {
11            if(i==1 || j==1 || j==(rows*2 - (2*i - 1)))
12            {
13                printf("*");
14            }
15            else
16            {
17                printf(" ");
18            }
19        }
20        printf("\n");
21    }
22    return 0;
23 }

```

The output window shows the following text:

```

SHARDUL RATURI
Enter number of rows: 7
*****
 *   *
* * *
* * *
 *   *
*****

```

The program successfully prints a diamond pattern of asterisks for 7 rows. The output window also shows the file path: /tmp/knzQ1jH1NZ.o.

PROGRAM 92.HALF DIAMOND STAR PATTERN

```
#include<stdio.h>

int main()
{
    int i, j, N, columns;
    printf("SHARDUL RATURI");
    printf("\nEnter number of columns:");
    scanf("%d",&N);
    columns=1;
    for(i=1;i<N*2;i++)
    {
        for(j=1; j<=columns; j++)
        {
            printf("*");
        }
        if(i < N)
        {
            columns++;
        }
        else
        {
            columns--;
        }
    }
}
```

```

}

printf("\n");

}

return 0;

}

```

The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The page title is "Programiz C Online Compiler". The code editor contains the following C program:

```

1 #include<stdio.h>
2 int main()
3 {
4     int i, j, N, columns;
5     printf("SHARDUL RATURI");
6     printf("\nEnter number of columns:");
7     scanf("%d",&N);
8     columns=1;
9     for(i=1;i<=N*2;i++)
10     {
11         for(j=1;j<=columns;j++)
12         {
13             printf(" ");
14         }
15         if(i < N)
16         {
17             columns++;
18         }
19         else
20         {
21             columns--;
22         }
23         printf("\n");
24     }
25     return 0;
26 }
27

```

The output window shows the following text:

```

//http://knzqljht16z.0
SHARDUL RATURI
Enter number of columns:6
*
**
***
****
*****
*****
*****
****
***
**
*

```

PROGRAM 93.MIRRORED HALF DIAMOND STAR PATTERN

```

#include <stdio.h>

int main()
{
    int i, j, N;

    int star, spaces;

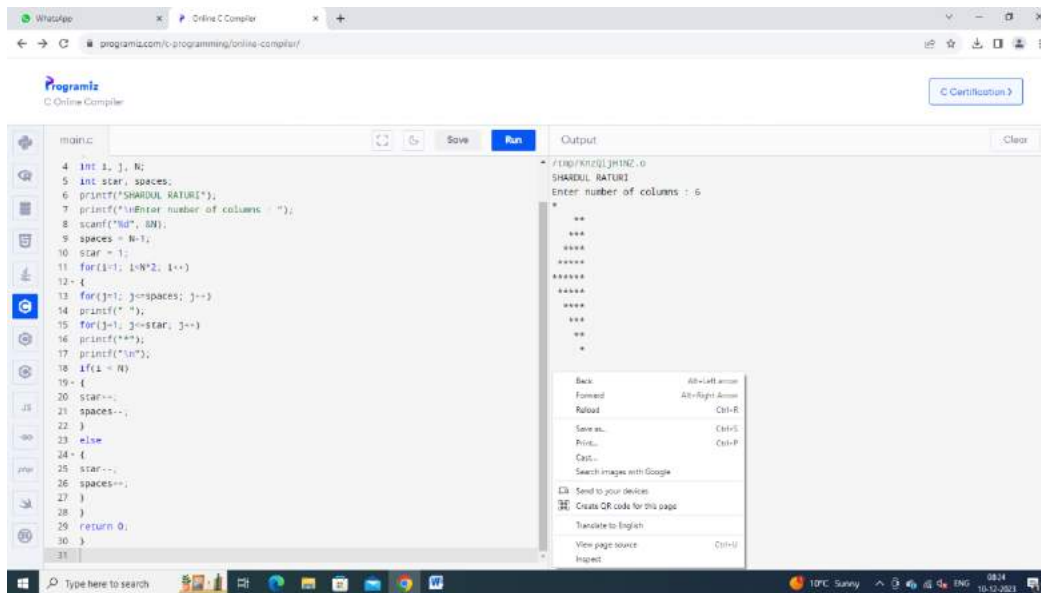
    printf("SHARDUL RATURI");

    printf("\nEnter number of columns : ");

    scanf("%d", &N);

```

```
spaces = N-1;
star = 1;
for(i=1; i<N*2; i++)
{
    for(j=1; j<=spaces; j++)
        printf(" ");
    for(j=1; j<=star; j++)
        printf("*");
    printf("\n");
    if(i < N)
    {
        star++;
        spaces--;
    }
    else
    {
        star--;
        spaces++;
    }
}
return 0;
}
```

```
4 int i, j, R;
5 int star, spaces;
6 printf("SHARDUL RATURI");
7 printf("\nEnter number of columns : ");
8 scanf("%d", &N);
9 spaces = N-1;
10 star = 1;
11 for(i=1; i<N*2; i++)
12 {
13     for(j=1; j<spaces; j++)
14         printf(" ");
15     for(j=1; j<star; j++)
16         printf(++);
17     printf("\n");
18     if(i < N)
19     {
20         star++;
21         spaces--;
22     }
23     else
24     {
25         star--;
26         spaces++;
27     }
28 }
29 return 0;
30 }
```

Output

```
/tmp/KXZ0JH1HZ.o
SHARDUL RATURI
Enter number of columns : 5
*
**
***
****
*****
*****
****
***
**
*
```

NUMBER PATTERN PROGRAMS –

PROGRAM 94. NUMBER PATTERN 1

11111

11111

11111

11111

11111

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int rows, cols, i, j;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter number of rows: ");
```

```

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

for(i=1; i<=rows; i++)
{
for(j=1; j<=cols; j++)
{
printf("1 ");
}

printf("\n");
}

return 0;
}

```

The screenshot shows a web browser window with the URL `programiz.com/online-compiler/`. The page title is "Programiz C Online Compiler". The code editor contains the following C program:

```

1 #include <stdio.h>
2 int main()
3 {
4     int rows, cols, i, j;
5     printf("SHARUL RATURI");
6     printf("\nEnter number of rows: ");
7     scanf("%d", &rows);
8     printf("Enter number of columns: ");
9     scanf("%d", &cols);
10    for(i=1; i<=rows; i++)
11    {
12        for(j=1; j<=cols; j++)
13        {
14            printf("1 ");
15        }
16        printf("\n");
17    }
18    return 0;
19 }

```

The output window shows the following text:

```

/rep/rouqj3h1z7.o
SHARUL RATURI
Enter number of rows: 6
Enter number of columns: 5
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1

```

PROGRAM 95.NUMBER PATTERN 2

11111

00000

11111

00000

11111

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int rows, cols, i, j;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter number of rows: ");
```

```
scanf("%d", &rows);
```

```
printf("\nEnter number of columns: ");
```

```
scanf("%d", &cols);
```

```
for(i=1; i<=rows; i++)
```

```
{
```

```
for(j=1; j<=cols; j++)
```

```
{
```

```
if(i%2 == 1)
```

```
{
```

```
printf("1 ");
```

```
}
```

```
else
```

```

{
printf("0 ");
}
}

printf("\n");
}

return 0;
}

```

```

main.c
1 #include <stdio.h>
2 int main()
3 {
4     int rows, cols, i, j;
5     printf("SHARDUL RATURI");
6     printf("\nEnter number of rows: ");
7     scanf("%d", &rows);
8     printf("\nEnter number of columns: ");
9     scanf("%d", &cols);
10    for(i=1; i<=rows; i++)
11    {
12        for(j=1; j<=cols; j++)
13        {
14            if(i%2 == 1)
15            {
16                printf("1 ");
17            }
18            else
19            {
20                printf("0 ");
21            }
22        }
23        printf("\n");
24    }
25    return 0;
26 }

```

Output

```

/tmp/KxZQjH1NZ.o
SHARDUL RATURI
Enter number of rows: 5
Enter number of columns: 6
1 1 1 1 1 1
0 0 0 0 0 0
1 1 1 1 1 1
0 0 0 0 0 0
1 1 1 1 1 1

```

PROGRAM 96.NUMBER PATTERN 3

01010

01010

01010

01010

01010

```
#include <stdio.h>

int main()
{
    int rows, cols, i, j;
    printf("SHARDUL RATURI");
    printf("\nEnter number of rows: ");
    scanf("%d", &rows);
    printf("\nEnter number of columns: ");
    scanf("%d", &cols);
    for(i=1; i<=rows; i++)
    {
        for(j=1; j<=cols; j++)
        {
            if(j%2 == 1)
            {
                printf("0 ");
            }
            else
            {
                printf("1 ");
            }
        }
    }
}
```

```
printf("\n");

}

return 0;

}
```

The screenshot shows the Programiz C Online Compiler interface. The code editor contains a C program that prints the name 'SHARDUL Raturi', prompts the user for the number of rows and columns, and then prints a grid of 0s and 1s. The output window shows the execution results, including the name and the user input.

```
main.c
1 #include <stdio.h>
2 int main()
3 {
4     int rows, cols, i, j;
5     printf("SHARDUL Raturi");
6     printf("\nEnter number of rows: ");
7     scanf("%d", &rows);
8     printf("\nEnter number of columns: ");
9     scanf("%d", &cols);
10    for(i=1; i<=rows; i++)
11    {
12        for(j=1; j<=cols; j++)
13        {
14            if(j%2 == 1)
15            {
16                printf("0 ");
17            }
18            else
19            {
20                printf("1 ");
21            }
22        }
23        printf("\n");
24    }
25    return 0;
26 }
27
```

Output

```
/tmp/KnzQlJH1NZ.o
SHARDUL Raturi
Enter number of rows: 4
Enter number of columns: 3
0 1 0
0 1 0
0 1 0
0 1 0
```

PROGRAM 97.NUMBER PATTERN 4

11111

10001

10001

10001

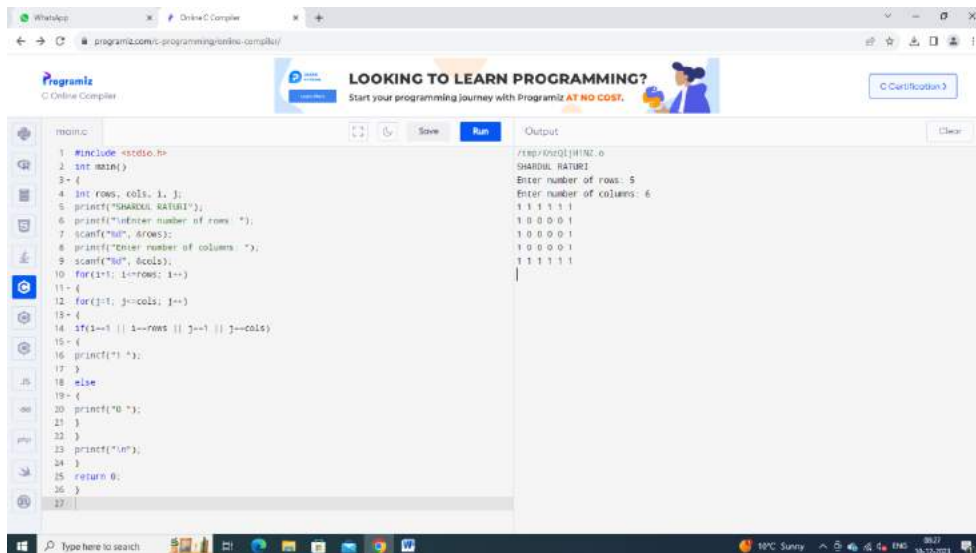
11111

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int rows, cols, i, j;
printf("SHARDUL RATURI");
printf("\nEnter number of rows: ");
scanf("%d", &rows);
printf("Enter number of columns: ");
scanf("%d", &cols);
for(i=1; i<=rows; i++)
{
    for(j=1; j<=cols; j++)
    {
        if(i==1 || i==rows || j==1 || j==cols)
        {
            printf("1 ");
        }
        else
        {
            printf("0 ");
        }
    }
    printf("\n");
}
return 0;
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     int rows, cols, i, j;
5     printf("SHARDUL RATURI\n");
6     printf("\nEnter number of rows: ");
7     scanf("%d", &rows);
8     printf("Enter number of columns: ");
9     scanf("%d", &cols);
10    for(i=1; i<=rows; i++)
11    {
12        for(j=1; j<=cols; j++)
13        {
14            if(i==1 || i==rows || j==1 || j==cols)
15            {
16                printf("1 ");
17            }
18            else
19            {
20                printf("0 ");
21            }
22        }
23        printf("\n");
24    }
25    return 0;
26 }
```

Output

```
SHARDUL RATURI
Enter number of rows: 5
Enter number of columns: 6
1 1 1 1 1 1
1 0 0 0 0 1
1 0 0 0 0 1
1 0 0 0 0 1
1 1 1 1 1 1
```

PROGRAM 98.NUMBER PATTERN 5

11111

11111

11011

11111

11111

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int rows, cols, i, j;
```

```
int centerRow, centerCol;
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter number of rows: ");
```

```
scanf("%d", &rows);
```



```
printf("Enter number of columns: ");
scanf("%d", &cols);
centerRow = (rows + 1) / 2;
centerCol = (cols + 1) / 2;
for(i=1; i<=rows; i++)
{
    for(j=1; j<=cols; j++)
    {
        if(centerCol == j && centerRow == i)
        {
            printf("0 ");
        }
        else if(cols%2 == 0 && centerCol+1 == j)
        {
            if(centerRow == i || (rows%2 == 0 && centerRow+1 == i))
            printf("0 ");
            else
            printf("1 ");
        }
        else if(rows%2 == 0 && centerRow+1 == i)
        {
            if(centerCol == j || (cols%2 == 0 && centerCol+1 == j))
            printf("0 ");
            else
```

```

printf("1 ");
}

else
{
printf("1 ");
}

}

printf("\n");
}

return 0;
}

```

The screenshot shows a web browser window with an online C compiler. The code in the editor is as follows:

```

17 if(centerCol == 1 && centerRow == 1)
18 {
19 printf("0 ");
20 }
21 else if(cols%2 == 0 && centerCol+1 == 1)
22 {
23 if(centerRow == 1 || (rows%2 == 0 && centerRow-1 == 1))
24 printf("0 ");
25 else
26 printf("1 ");
27 }
28 else if(rows%2 == 0 && centerRow-1 == 1)
29 {
30 if(centerCol == 1 || (cols%2 == 0 && centerCol-1 == 1))
31 printf("0 ");
32 else
33 printf("1 ");
34 }
35 else
36 {
37 printf("1 ");
38 }
39 }
40 printf("\n");
41 }
42 return 0;
43 }
44

```

The output of the program is shown on the right:

```

/tmp/100018100.c
998000: 181000:1
Enter number of rows: 6
Enter number of columns: 7
1 1 1 1 1 1 1
1 1 1 1 1 1 1
1 1 1 0 1 1 1
1 1 1 0 1 1 1
1 1 1 1 1 1 1
1 1 1 1 1 1 1

```

PROGRAM 99.NUMBER PATTERN 6

10101

01010

10101

01010

10101

```
#include <stdio.h>

int main()
{
    int rows, cols, i, j, k;
    printf("SHARDUL RATURI");
    printf("\nEnter number of rows: ");
    scanf("%d", &rows);
    printf("Enter number of columns: ");
    scanf("%d", &cols);
    k = 1;
    for(i=1; i<=rows; i++)
    {
        for(j=1; j<=cols; j++)
        {
            if(k == 1)
            {
                printf("1 ");
            }
            else
            {
                printf("0 ");
            }
        }
    }
}
```

```

}

k *= -1;

}

if(cols % 2 == 0)

{

k *= -1;

}

printf("\n");

}

return 0;

}

```

The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The page title is "Programiz C Online Compiler". The code editor contains the following C code:

```

6 printf("Enter number of rows: ");
7 scanf("%d", &rows);
8 printf("Enter number of columns: ");
9 scanf("%d", &cols);
10 k = 1;
11 for(i=1; i<=rows; i++)
12 {
13     for(j=1; j<=cols; j++)
14     {
15         if(k == 1)
16         {
17             printf("%t ", k);
18         }
19         else
20         {
21             printf("%d ", k);
22         }
23         k *= -1;
24     }
25     if(cols % 2 == 0)
26     {
27         k *= -1;
28     }
29     printf("\n");
30 }
31 return 0;
32 }

```

The output window shows the following text:

```

/c:/tmp/knu01jhtn2.o
SHARDUL RASTURI
Enter number of rows: 6
Enter number of columns: 7
1 0 1 0 1 0 1
0 1 0 1 0 1 0
1 0 1 0 1 0 1
0 1 0 1 0 1 0
1 0 1 0 1 0 1
0 1 0 1 0 1 0

```

PROGRAM 100. TO READ N NUMBER OF VALUES IN AN ARRAY AND DISPLAY THEM IN REVERSE ORDER.

```
#include <stdio.h>
```

```
int main() {
```

```
    int b, c, a[100];
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter the number of elements in the array: ");
```

```
    scanf("%d", &b);
```

```
    printf("Enter %d integers\n", b);
```

```
    for (c = 0; c < b; c++) {
```

```
        scanf("%d", &a[c]);
```

```
    }
```

```
    printf("The elements in reverse order are:\n");
```

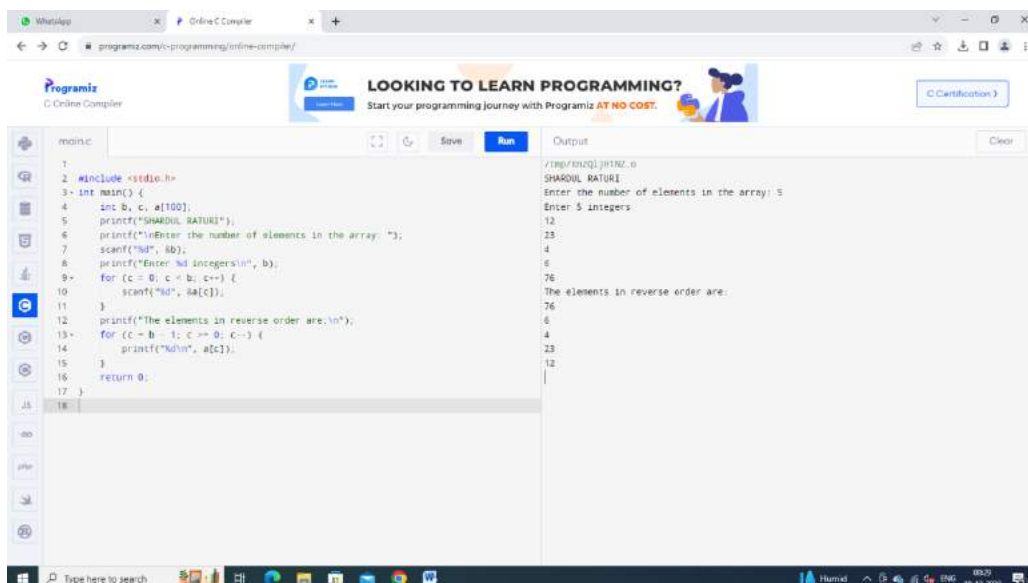
```
    for (c = b - 1; c >= 0; c--) {
```

```
        printf("%d\n", a[c]);
```

```
    }
```

```
    return 0;
```

```
}
```



The screenshot shows the Programiz Online C Compiler interface. The code editor on the left contains the C program, and the output window on the right shows the execution results. The code prints the name 'SHARDUL RATURI', prompts for the number of elements (5), reads 5 integers (12, 23, 4, 6, 76), and then prints them in reverse order (76, 6, 4, 23, 12).

```
main.c
1  #include <stdio.h>
2  int main() {
3      int b, c, a[100];
4      printf("SHARDUL RATURI");
5      printf("\nEnter the number of elements in the array: ");
6      scanf("%d", &b);
7      printf("Enter %d integers\n", b);
8      for (c = 0; c < b; c++) {
9          scanf("%d", &a[c]);
10     }
11     printf("The elements in reverse order are:\n");
12     for (c = b - 1; c >= 0; c--) {
13         printf("%d\n", a[c]);
14     }
15     return 0;
16 }
17 }
18 }
```

Output

```
//tmp/rmzqj3H1N2.o
SHARDUL RATURI
Enter the number of elements in the array: 5
Enter 5 integers
12
23
4
6
76
The elements in reverse order are:
76
6
4
23
12
```

PROGRAM 101. TO FIND THE SUM OF ALL ELEMENTS OF THE ARRAY.

```
#include <stdio.h>

int main()
{
    printf("SHARDUL RATURI");

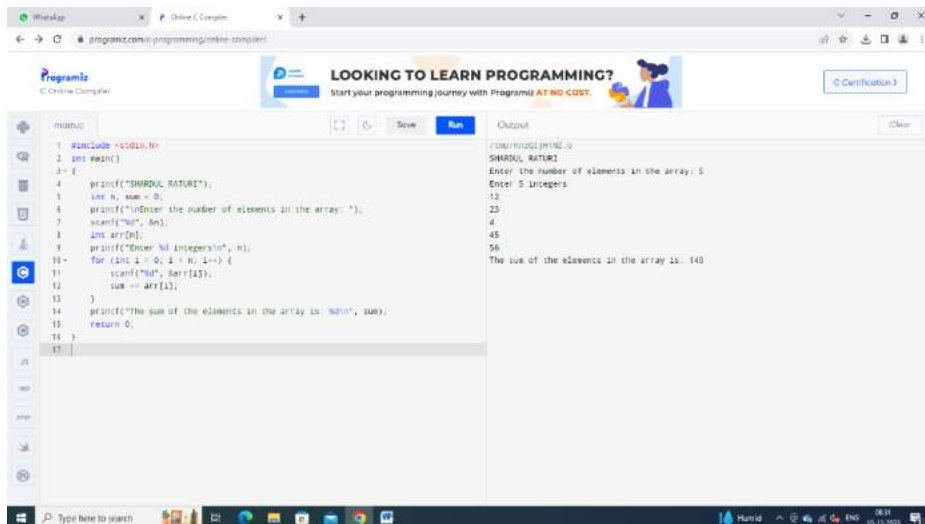
    int n, sum = 0;

    printf("\nEnter the number of elements in the array: ");
    scanf("%d", &n);

    int arr[n];

    printf("Enter %d integers\n", n);
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
        sum += arr[i];
    }

    printf("The sum of the elements in the array is: %d\n", sum);
    return 0;
}
```



PROGRAM 102. TO COPY THE ELEMENTS OF ONE ARRAY INTO ANOTHER ARRAY.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int n, i;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter the number of elements in the array: ");
```

```
    scanf("%d", &n);
```

```
    int array[n];
```

```
    int copiedarray[n];
```

```
    printf("Enter %d integers for the source array\n", n);
```

```
    for (i = 0; i < n; i++) {
```

```
        scanf("%d", &array[i]);
```

```
    }
```

```

for (i = 0; i < n; i++) {
    copiedarray[i] = array[i];
}

printf("Elements copied from source array to destination array
are: \n");

for (i = 0; i < n; i++) {
    printf("%d ", copiedarray[i]);
}

printf("\n");

return 0;
}

```

The screenshot shows a web browser window with an online C compiler. The code in the editor is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     int n, i;
5     printf("SHARDUL RATHI");
6     printf("\nEnter the number of elements in the array: ");
7     scanf("%d", &n);
8     int array[n];
9     int copiedarray[n];
10    printf("Enter %d integers for the source array\n", n);
11    for (i = 0; i < n; i++) {
12        scanf("%d", &array[i]);
13    }
14    for (i = 0; i < n; i++) {
15        copiedarray[i] = array[i];
16    }
17    printf("Elements copied from source array to destination array are: \n");
18    for (i = 0; i < n; i++) {
19        printf("%d ", copiedarray[i]);
20    }
21    printf("\n");
22    return 0;
23 }

```

The output window shows the following text:

```

/rep/0001/print2.c
SHARDUL RATHI
Enter the number of elements in the array:
3
Enter 3 integers for the source array:
4
5
6
Elements copied from source array to destination array are:
4 5 6

```

PROGRAM 103. TO COUNT THE TOTAL NUMBER OF DUPLICATE ELEMENTS IN AN ARRAY.

```
#include <stdio.h>
```



```
int main()
{
    int n, i, j, count = 0;
    printf("SHARDUL RATURI");
    printf("\nEnter the number of elements in the array: ");
    scanf("%d", &n);
    int arr[n];
    int duplicate[n];
    printf("Enter %d integers\n", n);
    for (i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
        duplicate[i] = -1;
    }
    for (i = 0; i < n; i++) {
        int currentElement = arr[i];
        int duplicateCount = 1;
        for (j = i + 1; j < n; j++) {
            if (currentElement == arr[j]) {
                duplicateCount++;
                duplicate[j] = 0;
            }
        }
        if (duplicate[i] != 0) {
            duplicate[i] = duplicateCount;
        }
    }
}
```

```

    }
}

for (i = 0; i < n; i++) {
    if (duplicate[i] > 1) {
        count++;
    }
}

printf("The total number of duplicate elements in the array is:
%d\n", count);

return 0;
}

```

The screenshot shows a web browser window with an online C compiler. The code in the editor is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     int n, i, j, count = 0;
5     printf("PROGRAM: DUPLICATE\n");
6     printf("Enter the number of elements in the array: ");
7     scanf("%d", &n);
8     int arr[n];
9     int duplicate[n];
10    printf("Enter %d integers\n", n);
11    for (i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13        duplicate[i] = 0;
14    }
15    for (i = 0; i < n; i++) {
16        int currentElement = arr[i];
17        int duplicateCount = 0;
18        for (j = i + 1; j < n; j++) {
19            if (currentElement == arr[j]) {
20                duplicateCount++;
21                duplicate[j] = 0;
22            }
23        }
24        if (duplicate[i] != 0) {
25            duplicate[i] = duplicateCount;
26        }
27    }
28    for (i = 0; i < n; i++) {

```

The output window shows the following text:

```

./main.c:1:1: warning: ISO C90 forbids mixed declarations and code [-Wdeclaration-after-statement]
1 #include <stdio.h>
  ^
./main.c:2:1: note: declare 'int main()' before using it
2 int main()
  ^
./main.c:11:14: warning: format '%d' expects argument of type 'int', but argument 2 of type 'int*' [-Wformat]
11     scanf("%d", &n);
    ~~~~~

```

PROGRAM 104. TO FIND THE MAXIMUM AND MINIMUM ELEMENTS IN AN ARRAY.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int n, i;

printf("SHARDUL RATURI");

printf("\nEnter the number of elements in the array: ");

scanf("%d", &n);

int arr[n];

printf("Enter %d integers\n", n);

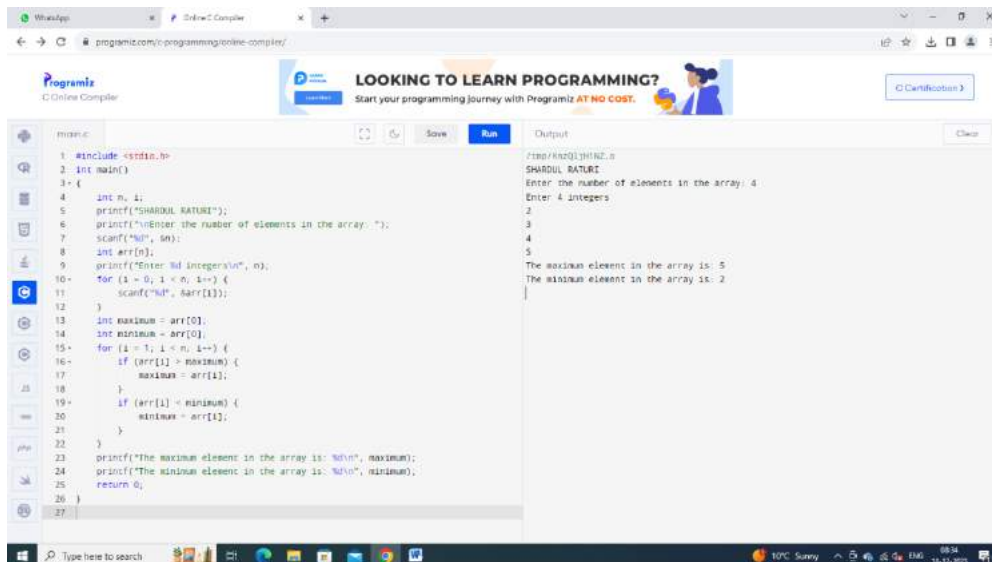
for (i = 0; i < n; i++) {
    scanf("%d", &arr[i]);
}

int maximum = arr[0];
int minimum = arr[0];

for (i = 1; i < n; i++) {
    if (arr[i] > maximum) {
        maximum = arr[i];
    }
    if (arr[i] < minimum) {
        minimum = arr[i];
    }
}

printf("The maximum element in the array is: %d\n", maximum);
printf("The minimum element in the array is: %d\n", minimum);

return 0;
}
```



```
1 #include <stdio.h>
2 int main()
3 {
4     int n, i;
5     printf("SHARDUL RATURI");
6     printf("\nEnter the number of elements in the array: ");
7     scanf("%d", &n);
8     int arr[n];
9     printf("Enter %d integers\n", n);
10    for (i = 0; i < n; i++) {
11        scanf("%d", &arr[i]);
12    }
13    int maximum = arr[0];
14    int minimum = arr[0];
15    for (i = 1; i < n; i++) {
16        if (arr[i] > maximum) {
17            maximum = arr[i];
18        }
19        if (arr[i] < minimum) {
20            minimum = arr[i];
21        }
22    }
23    printf("The maximum element in the array is: %d\n", maximum);
24    printf("The minimum element in the array is: %d\n", minimum);
25    return 0;
26 }
```

Output

```
SHARDUL RATURI
Enter the number of elements in the array: 4
Enter 4 integers
2
3
4
5
The maximum element in the array is: 5
The minimum element in the array is: 2
```

PROGRAM 105.. TO SORT THE ELEMENTS OF AN ARRAY IN DESCENDING ORDER

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int n, i, j, temp;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter the number of elements in the array: ");
```

```
    scanf("%d", &n);
```

```
    int arr[n];
```

```
    printf("Enter %d integers\n", n);
```

```
    for (i = 0; i < n; i++) {
```

```
        scanf("%d", &arr[i]);
```

```
    }
```

```
    for (i = 0; i < n; i++) {
```

```
        for (j = i + 1; j < n; j++) {
```

```

        if (arr[i] < arr[j]) {
            temp = arr[i];
            arr[i] = arr[j];
            arr[j] = temp;
        }
    }
}

printf("The array in descending order is: \n");
for (i = 0; i < n; i++) {
    printf("%d ", arr[i]);
}

printf("\n");

return 0;
}

```

The screenshot shows the Programiz Online C Compiler interface. The code editor on the left contains the following C program:

```

1 int main()
2 {
3     int n, i, j, temp;
4     printf("SHARDUL RATURI");
5     printf("\nEnter the number of elements in the array: ");
6     scanf("%d", &n);
7     int arr[n];
8     printf("Enter %d integers\n", n);
9     for (i = 0; i < n; i++) {
10        scanf("%d", &arr[i]);
11    }
12    for (i = 0; i < n; i++) {
13        for (j = i + 1; j < n; j++) {
14            if (arr[i] < arr[j]) {
15                temp = arr[i];
16                arr[i] = arr[j];
17                arr[j] = temp;
18            }
19        }
20    }
21    printf("The array in descending order is: \n");
22    for (i = 0; i < n; i++) {
23        printf("%d ", arr[i]);
24    }
25    printf("\n");
26    return 0;
27 }

```

The output window on the right shows the following execution results:

```

/tmp/RnzQ1jH1NZ.o
SHARDUL RATURI
Enter the number of elements in the array: 5
Enter 5 integers
7
8
9
10
11
The array in descending order is:
11 10 9 8 7

```

PROGRAM 106. TO SEPARATE ODD AND EVEN INTEGERS INTO SEPARATE ARRAYS.

```
#include <stdio.h>

int main()
{
    int n, i, j = 0, k = 0;
    printf("SHARDUL RATURI");
    printf("\nEnter the number of elements in the array: ");
    scanf("%d", &n);
    int arr[n];
    int evenArray[n], oddArray[n];
    printf("Enter %d integers\n", n);
    for (i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
        if (arr[i] % 2 == 0) {
            evenArray[j] = arr[i];
            j++;
        } else {
            oddArray[k] = arr[i];
            k++;
        }
    }
    printf("The even elements are: \n");
    for (i = 0; i < j; i++) {
        printf("%d ", evenArray[i]);
    }
}
```

```

printf("\n");

printf("The odd elements are: \n");

for (i = 0; i < k; i++) {

    printf("%d ", oddArray[i]);

}

printf("\n");

return 0;

}

```

The screenshot shows a web browser window with the 'Programiz C Online Compiler' interface. The code editor on the left contains a C program that prompts the user to enter the number of elements in an array (n), then enters n integers. It then separates these integers into two arrays: 'evenArray' for even numbers and 'oddArray' for odd numbers. The program prints the even elements first, followed by the odd elements. The output window on the right shows the execution results: the user entered 4 integers (1, 2, 3, 4), the even elements are 2 and 4, and the odd elements are 1 and 3.

PROGRAM 107. WRITE A PROGRAM IN C TO MERGE TWO ARRAYS OF THE SAME SIZE SORTED IN ASCENDING ORDER

```

#include <stdio.h>

void mergeArrays(int arr1[], int arr2[], int n, int result[])
{

    int i = 0, j = 0, k = 0;

    printf("SHARDUL RATURI");

```

```
while (i < n && j < n) {  
    if (arr1[i] <= arr2[j]) {  
        result[k] = arr1[i];  
        i++;  
    } else {  
        result[k] = arr2[j];  
        j++;  
    }  
    k++;  
}
```

```
while (i < n) {  
    result[k] = arr1[i];  
    i++;  
    k++;  
}
```

```
while (j < n) {  
    result[k] = arr2[j];  
    j++;  
    k++;  
}
```

```
}
```

```
int main() {
```

```
    int n;
```

```
    printf("\nEnter the size of the arrays: ");
```



```
scanf("%d", &n);
int arr1[n], arr2[n], result[2 * n];
printf("Enter elements of the first array (ascending order):\n");
for (int i = 0; i < n; i++) {
    scanf("%d", &arr1[i]);
}
printf("Enter elements of the second array (ascending order):\n");
for (int i = 0; i < n; i++) {
    scanf("%d", &arr2[i]);
}
mergeArrays(arr1, arr2, n, result);
printf("\nMerged array in ascending order:\n");
for (int i = 0; i < 2 * n; i++) {
    printf("%d ", result[i]);
}
printf("\n");
return 0;
}
```

```
21 while (j < n) {
22     result[k] = arr2[j];
23     j++;
24     k++;
25 }
26
27 int main() {
28     int n;
29     printf("Enter the size of the arrays: ");
30     scanf("%d", &n);
31     int arr1(n), arr2(n), result(2 * n);
32     printf("Enter elements of the first array (ascending order):\n");
33     for (int i = 0; i < n; i++) {
34         scanf("%d", &arr1[i]);
35     }
36     printf("Enter elements of the second array (ascending order):\n");
37     for (int i = 0; i < n; i++) {
38         scanf("%d", &arr2[i]);
39     }
40     mergeArrays(arr1, arr2, n, result);
41     printf("Merged array in ascending order:\n");
42     for (int i = 0; i < 2 * n; i++) {
43         printf("%d ", result[i]);
44     }
45     printf("\n");
46     return 0;
47 }
```

Output

```
/tmp/king111112.o
Enter the size of the arrays: 4
Enter elements of the first array (ascending order):
1
2
3
4
Enter elements of the second array (ascending order):
3
4
5
6
S4450UL B4TUBI
Merged array in ascending order:
1 2 3 4 5 6
```

PROGRAM 108. TO MERGE TWO ARRAYS OF THE SAME SIZE SORTED IN DESCENDING ORDER.

#include <stdio.h>

```
void mergeDescending(int arr1[], int arr2[], int result[], int size) {
```

```
    int i = 0, j = 0, k = 0;
```

```
    while (i < size && j < size) {
```

```
        if (arr1[i] >= arr2[j]) {
```

```
            result[k] = arr1[i];
```

```
            i++;
```

```
        } else {
```

```
            result[k] = arr2[j];
```

```
            j++;
```

```

    }
    k++;
}
while (i < size) {
    result[k] = arr1[i];
    i++;
    k++;
}
while (j < size) {
    result[k] = arr2[j];
    j++;
    k++;
}
}

int main()
{
    int size;
    printf("SHARDUL RATURI");
    printf("\nEnter the size of the arrays: ");
    scanf("%d", &size);
    int arr1[size], arr2[size], mergedArray[2 * size];
    printf("Enter elements of the first array in descending order:\n");
    for (int i = 0; i < size; i++) {
        scanf("%d", &arr1[i]);
    }

```

```

    }

    printf("Enter elements of the second array in descending
order:\n");

    for (int i = 0; i < size; i++) {

        scanf("%d", &arr2[i]);

    }

    mergeDescending(arr1, arr2, mergedArray, size);

    printf("Merged array in descending order: ");

    for (int i = 0; i < 2 * size; i++) {

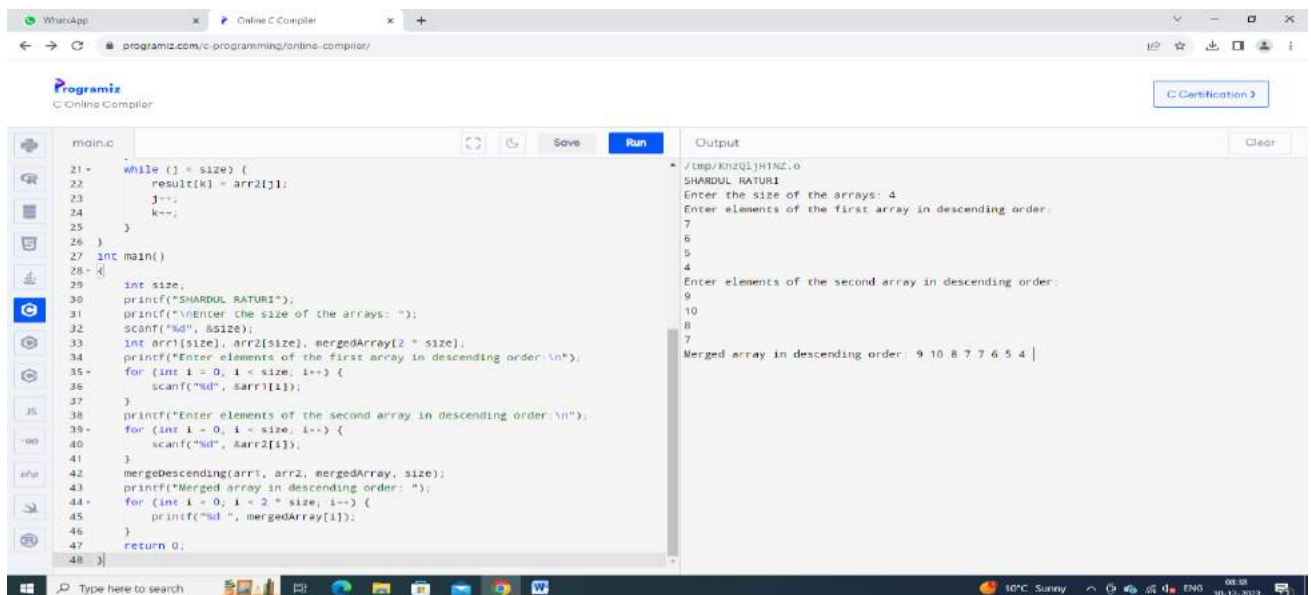
        printf("%d ", mergedArray[i]);

    }

    return 0;

}

```



The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The page displays the C Online Compiler interface. On the left, there is a file explorer showing `main.c`. The main editor contains the following C code:

```

21+ while (j < size) {
22+     result[k] = arr2[j];
23+     j++;
24+     k++;
25+ }
26+ }
27+ int main()
28+ {
29+     int size;
30+     printf("SHARDUL RATURI");
31+     printf("\nEnter the size of the arrays: ");
32+     scanf("%d", &size);
33+     int arr1[size], arr2[size], mergedArray[2 * size];
34+     printf("Enter elements of the first array in descending order:\n");
35+     for (int i = 0; i < size; i++) {
36+         scanf("%d", &arr1[i]);
37+     }
38+     printf("Enter elements of the second array in descending order:\n");
39+     for (int i = 0; i < size; i++) {
40+         scanf("%d", &arr2[i]);
41+     }
42+     mergeDescending(arr1, arr2, mergedArray, size);
43+     printf("Merged array in descending order: ");
44+     for (int i = 0; i < 2 * size; i++) {
45+         printf("%d ", mergedArray[i]);
46+     }
47+     return 0;
48+ }

```

On the right, the 'Output' window shows the following text:

```

/tmp/knzqljH1NZ.o
SHARDUL RATURI
Enter the size of the arrays: 4
Enter elements of the first array in descending order:
7
6
5
4
Enter elements of the second array in descending order:
9
10
8
7
Merged array in descending order: 9 10 8 7 7 6 5 4 |

```

PROGRAM . CONSIDER TWO MATRICES OF THE SIZE M AND N. IMPLEMENT

MATRIX OPERATION AND DISPLAY. SHOW THESE THINGS IN PROGRAM

109) READ MATRIX ELEMENTS AND DISPLAY

```
#include <stdio.h>

int main()
{
    int i, j, m, n;
    int matrix[10][20];
    printf("SHARDUL RATURI");
    printf("\nEnter number of rows : ");
    scanf("%d", &m);
    printf("Enter number of columns : ");
    scanf("%d", &n);
    /* Input data in matrix */
    for (i = 0; i < m; i++)
    {
        for (j = 0; j < n; j++)
        {
            printf("Enter data in [%d][%d]: ", i, j);
            scanf("%d", &matrix[i][j]);
        }
    }
}
```

```

for (i = 0; i < m; i++)
{
    for (j = 0; j < n; j++)
    {
        printf("%d\t", matrix[i][j]);
    }
    printf("\n");
}

return 0;
}

```

The screenshot shows a web browser window with the 'Programiz Online C Compiler'. The code editor contains a C program named 'main.c' that takes user input for the number of rows (3) and columns (5), then enters data into a 3x5 matrix. The output window shows the entered matrix and two search queries that failed to find specific elements.

```

main.c
3- {
4-     int i, j, m, n;
5-     int matrix[10][20];
6-     printf("SHARUOL RATURU");
7-     printf("\nEnter number of rows : ");
8-     scanf("%d", &m);
9-     printf("Enter number of columns : ");
10-    scanf("%d", &n);
11-    /* Input data in matrix */
12-    for (i = 0; i < m; i++)
13-    {
14-        for (j = 0; j < n; j++)
15-        {
16-            printf("Enter data in [%d][%d]: ", i, j);
17-            scanf("%d", &matrix[i][j]);
18-        }
19-    }
20-    for (i = 0; i < m; i++)
21-    {
22-        for (j = 0; j < n; j++)
23-        {
24-            printf("%d\t", matrix[i][j]);
25-        }
26-        printf("\n");
27-    }
28-    return 0;
29- }
30-
Output
/cmp/K0201jH1N2.o
SHARUOL RATURU
Enter number of rows : 3
Enter number of columns : 5
Enter data in [0][0]: 2 4
Enter data in [0][1]: Enter data in [0][2]: 5 7
Enter data in [0][3]: Enter data in [0][4]: 3 6
Enter data in [1][0]: Enter data in [1][1]: 4
Enter data in [1][2]: 7
Enter data in [1][3]: 9
Enter data in [1][4]: 11
Enter data in [2][0]: 45
Enter data in [2][1]: 2
Enter data in [2][2]: 45
Enter data in [2][3]: 45
Enter data in [2][4]: 67
72 4 5 7 3
6 4 7 9 11
45 2 45 45 67

8
dash: 3: 8: not found
7
dash: 4: 7: not found

```

110) MATRIX MULTIPLICATION AND DISPLAY

```

#include<stdio.h>

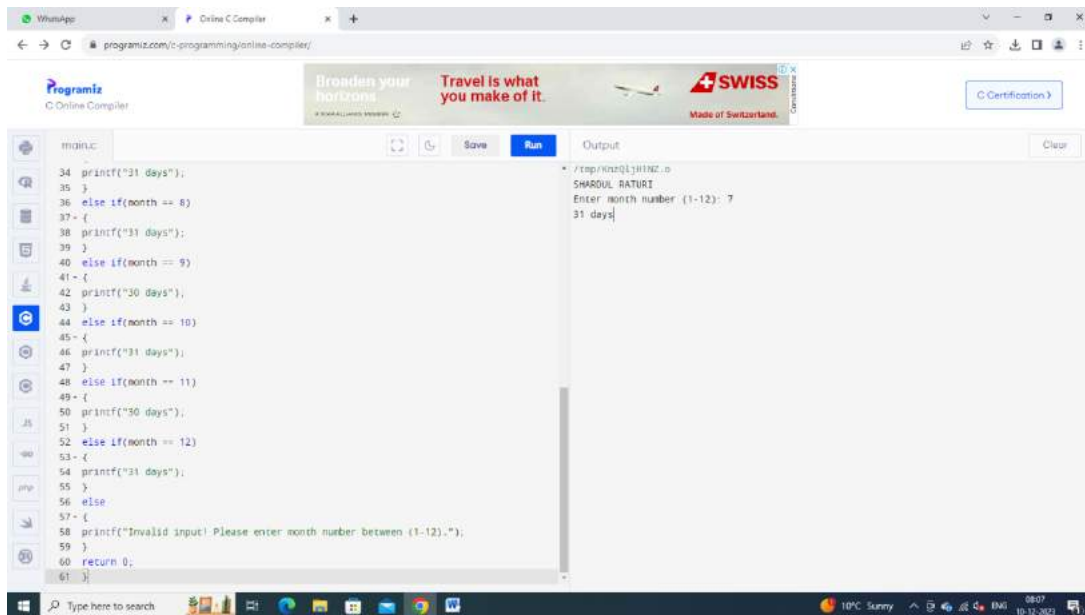
#include<stdlib.h>

int main()

```

```
{
    int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;
    printf("SHARDUL RATURI");
    printf("\nenter the number of row=");
    scanf("%d",&r);
    printf("enter the number of column=");
    scanf("%d",&c);
    printf("enter the first matrix element=\n");
    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }
    printf("enter these second matrix element=\n");
    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
        {
            scanf("%d",&b[i][j]);
        }
    }
    printf("multiply of the matrix=\n");
```

```
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
mul[i][j]=0;
for(k=0;k<c;k++)
{
mul[i][j]+=a[i][k]*b[k][j];
}
}
}
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
printf("%d\t",mul[i][j]);
}
printf("\n");
}
return 0;
}
```

111) ADDITION OF MATRIX AND DISPLAY

```
#include <stdio.h>
```

```
int main() {
```

```
    int r, c, a[100][100], b[100][100], sum[100][100], i, j;
```

```
    printf("SHARDUL RATURI");
```

```
    printf("\nEnter the number of rows : ");
```

```
    scanf("%d", &r);
```

```
    printf("Enter the number of columns: ");
```

```
    scanf("%d", &c);
```

```
    printf("\nEnter elements of 1st matrix:\n");
```

```
    for (i = 0; i < r; ++i)
```

```
        for (j = 0; j < c; ++j) {
```

```
            printf("Enter element a%d%d: ", i + 1, j + 1);
```

```
            scanf("%d", &a[i][j]);
```

```

    }
    printf("Enter elements of 2nd matrix:\n");
    for (i = 0; i < r; ++i)
        for (j = 0; j < c; ++j) {
            printf("Enter element b%d%d: ", i + 1, j + 1);
            scanf("%d", &b[i][j]);
        }
    for (i = 0; i < r; ++i)
        for (j = 0; j < c; ++j) {
            sum[i][j] = a[i][j] + b[i][j];
        }
    printf("\nSum of two matrices: \n");
    for (i = 0; i < r; ++i)
        for (j = 0; j < c; ++j) {
            printf("%d  ", sum[i][j]);
            if (j == c - 1) {
                printf("\n\n");
            }
        }
    return 0;
}

```

The screenshot shows a web browser window with the 'Programiz C Online Compiler'. The code in the editor is as follows:

```
main.c
8 scanf("%d", &c);
9 printf("\nEnter elements of 1st matrix:\n");
10 for (i = 0; i < r1; ++i)
11     for (j = 0; j < c; ++j) {
12         printf("Enter element a%d%d: ", i + 1, j + 1);
13         scanf("%d", &a[i][j]);
14     }
15 printf("Enter elements of 2nd matrix:\n");
16 for (i = 0; i < r2; ++i)
17     for (j = 0; j < c; ++j) {
18         printf("Enter element b%d%d: ", i + 1, j + 1);
19         scanf("%d", &b[i][j]);
20     }
21 for (i = 0; i < r; ++i)
22     for (j = 0; j < c; ++j) {
23         sum[i][j] = a[i][j] - b[i][j];
24     }
25 printf("\nSum of two matrices: \n");
26 for (i = 0; i < r; ++i)
27     for (j = 0; j < c; ++j) {
28         printf("%d ", sum[i][j]);
29         if (j == c - 1) {
30             printf("\n");
31         }
32     }
33 return 0;
34 }
```

The output window shows the following execution:

```
./omp/oa2xout003.o
SHARDUL RATURI
Enter the number of rows : 2
Enter the number of columns: 3
Enter elements of 1st matrix:
Enter element a11: 1
Enter element a12: 2
Enter element a13: 3
Enter element a21: 5
Enter element a22: 7
Enter element a23: 5
Enter elements of 2nd matrix:
Enter element b11: 4
Enter element b12: 4
Enter element b13: 0
Enter element b21: 5
Enter element b22: 5
Enter element b23: 2
Sum of two matrices:
5 6 11
10 12 7
```

112)SUBTRACTION OF MATRIX AND DISPLAY

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int m,n,c,d,first[10][10],second[10][10],difference[10][10];
```

```
printf("SHARDUL RATURI");
```

```
printf("\nEnter the number of rows and columns of matrix\n");
```

```
scanf("%d%d",&m,&n);
```

```
printf("Enter the elements of first matrix\n");
```

```
for (c=0;c<m;c++)
```

```
for(d=0;d<n;d++)scanf("%d",&first[c][d]);
```

```
printf("Enter the elements of second matrix\n");
```

```
for(c=0;c<m;c++)
```

```
for(d= 0;d< n;d++) scanf("%d", &second[c][d]);
```

```

printf("Difference of entered matrices:\n");
for (c= 0;c< m;c++)
{
for(d=0;d< n;d++)
{
difference[c][d] =first[c][d]-second[c][d];
printf("%d\t",difference[c][d]);
}
printf("\n");
}
return 0;
}

```

The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The page title is "Programiz C Online Compiler". The code editor on the left contains the following C code:

```

1 #include <stdio.h>
2 int main()
3 {
4     int m,n,c,d,first[10][10],second[10][10],difference[10][10];
5     printf("SHARDUL RATURI");
6     printf("\nEnter the number of rows and columns of matrix\n");
7     scanf("%d%d",&m,&n);
8     printf("Enter the elements of first matrix\n");
9     for (c=0;c<m;c++)
10         for (d=0;d<n;d++)scanf("%d",&first[c][d]);
11     printf("Enter the elements of second matrix\n");
12     for (c=0;c<m;c++)
13         for (d=0;d<n;d++) scanf("%d",&second[c][d]);
14     printf("Difference of entered matrices:\n");
15     for (c=0;c<m;c++)
16     {
17         for (d=0;d<n;d++)
18         {
19             difference[c][d] =first[c][d]-second[c][d];
20             printf("%d\t",difference[c][d]);
21         }
22         printf("\n");
23     }
24     return 0;
25 }

```

The output window on the right shows the following text:

```

/csp/ea280v80b3.0
SHARDUL RATURI
Enter the number of rows and columns of matrix
2
3
Enter the elements of first matrix
1
2
3
4
5
6
Enter the elements of second matrix
7
8
9
10
11
12
Difference of entered matrices:
-6 -6 -6
-6 -6 -6

```

113)TRANPOSE OF MATRIX AND DISPLAY

```
#include <stdio.h>
```

```
int main()
{
    int rows, cols;
    printf("SHARDUL RATURI");
    printf("\nEnter the number of rows: ");
    scanf("%d", &rows);
    printf("Enter the number of columns: ");
    scanf("%d", &cols);
    int matrix[rows][cols];
    int transpose[cols][rows];
    printf("Enter the elements of the matrix:\n");
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            transpose[j][i] = matrix[i][j];
        }
    }
    printf("Transpose of the matrix:\n");
    for (int i = 0; i < cols; i++) {
        for (int j = 0; j < rows; j++) {
```

```

        printf("%d ", transpose[i][j]);

    }

    printf("\n");

}

return 0;

}

```

The screenshot shows a web browser window with the URL `programiz.com/c-programming/online-compiler/`. The page features the Programiz logo and a "C Certification" button. The main area displays a C program for matrix multiplication. The code includes a `main` function that prompts the user for the number of rows and columns, then for each element of the first matrix. It then prompts for the second matrix elements and finally prints the resulting matrix.

```

main.c
22  for(j=0;j<c;j++)
23  {
24      scanf("%d",&b[i][j]);
25  }
26  }
27  printf("multiply of the matrix=\n");
28  for(i=0;i<r;i++)
29  {
30      for(j=0;j<c;j++)
31      {
32          mul[i][j]=0;
33          for(k=0;k<c;k++)
34          {
35              mul[i][j]=a[i][k]*b[k][j];
36          }
37      }
38  }
39  for(i=0;i<r;i++)
40  {
41      for(j=0;j<c;j++)
42      {
43          printf("%d\t",mul[i][j]);
44      }
45      printf("\n");
46  }
47  return 0;
48  }
49

```

The output window shows the following text:

```

/rxp/ea2K0VMe03.o
SHARDUL RATUR1
enter the number of row=2
enter the number of column=
2
enter the first matrix element=
1
2
3
4
enter these second matrix element=
2
4
6
8
multiply of the matrix=
14 20
30 44

```