

C Programming Assignment 1

Nisarg Patel

August 2025

1 QUESTION 1 : ADD TWO NUMBERS

```
#include <stdio.h>

int main()
{
    int a, b, c_sum;
    printf( "  NISARG PATEL \n" );
    printf( "  Enter two numbers: " );
    scanf("%d %d", &a, &b);
    c_sum = a + b;
    printf("Sum = %d\n", c_sum);
}
```

Program Output Screenshot

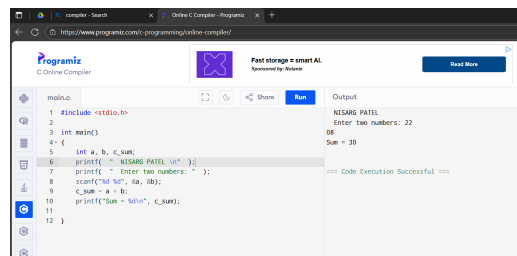


Figure 1: Program 1

Nisarg Patel

August 26, 2025

QUESTION 2:SUBTRACT TWO NUMBERS

```
#include <stdio.h>

int main()
{
    int a, b, c_subtract;
    printf( "    NISARG PATEL \n" );
    printf( "    Enter two numbers: " );
    scanf("%d %d", &a, &b);
    c_subtract = a - b;
    printf("Sub = %d\n", c_subtract);
}
```

Program Output Screenshot

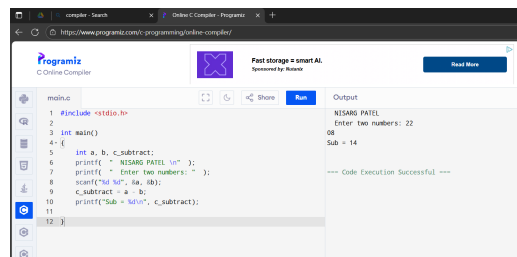


Figure 1: Program 2

Nisarg Patel

August 26, 2025

QUESTION 2: MULTIPLY TWO NUMBERS

```
#include <stdio.h>

int main()
{
    int a, b, c_multiply;
    printf( "    NISARG PATEL \n" );
    printf( "    Enter two numbers: " );
    scanf("%d %d", &a, &b);
    c_multiply = a * b;
    printf("product = %d\n", c_multiply);
}
```

Program Output Screenshot

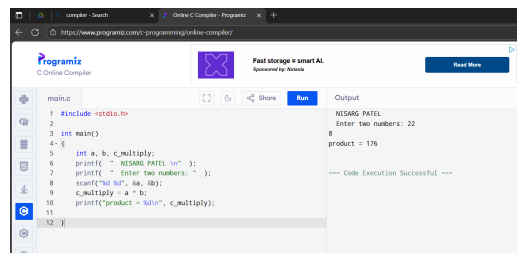


Figure 1: Program 3

Nisarg Patel

August 26, 2025

QUESTION 4: DIVIDE TWO NUMBERS

```
#include <stdio.h>

int main()
{
    float a, b, c_divison;
    printf( "  NISARG PATEL \n" );
    printf( "  Enter two numbers: " );
    scanf("%f %f", &a, &b);
    c_divison = a / b;
    printf("quotient = %f\n", c_divison);
}
```

Program Output Screenshot

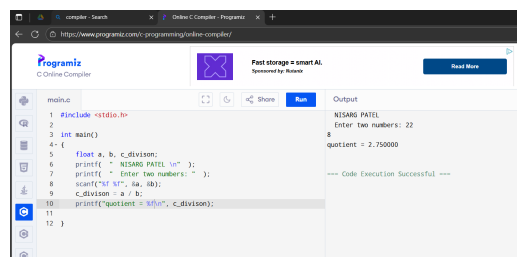


Figure 1: Program 4

QUESTION 5: ALL OPT NUMBERS

```
#include <stdio.h>

int main()
{
    float a, b, c_divison, c_multiply, c_addition, c_subtract;
    printf( "   NISARG PATEL \n" );
    printf( "   Enter two numbers: " );
    scanf("%f %f", &a, &b);
    c_divison = a / b;
    printf("quotient = %f\n", c_divison);
    c_multiply=a*b;
    printf("product = %f\n", c_multiply);
    c_addition=a+b;
    printf("sum = %f\n", c_addition);
    c_subtract=a-b;
    printf("subtract = %f\n", c_subtract);
}
```

Program Output Screenshot



Figure 1: Program 5

QUESTION 6: HOUR TO MIN CONVERSION PGM

```
#include <stdio.h>

int main()
{
    int hour, minutes;
    printf( "  NISARG PATEL \n" );
    printf( "  Enter time inn hour : " );
    scanf("%d", &hour);
    minutes = hour * 60;
    printf ( "minutes = %d\n ", minutes );
}
```

Program Output Screenshot

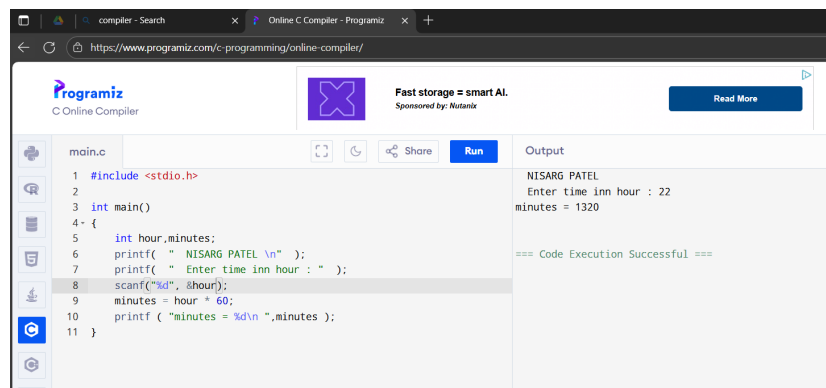


Figure 1: Program 6

QUESTION 7 : MINUTES INTO HOURS

```
#include <stdio.h>

int main()
{
    float hour,minutes;
    int h,m;
    printf( "  NISARG PATEL \n" );
    printf( "  Enter time inn MINUTES : " );
    scanf("%f", &minutes);
    hour = minutes / 60;
    printf ( "hour = %.2f\n ",hour );
    h = (int)(minutes / 60);
    m = (int)minutes % 60;
    printf(": %d hours and %d minutes\n", h, m);
}
```

Program Output Screenshot

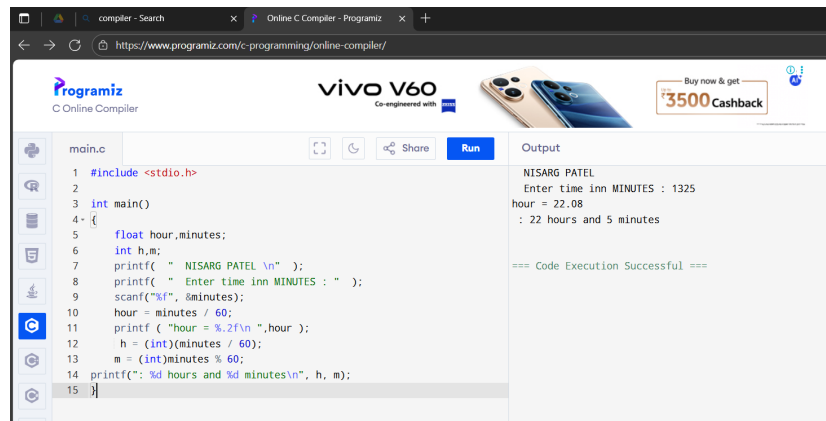


Figure 1: Program 7

QUESTION 8 : CONVERRT DOLLAR INTO RUPEES

```
#include <stdio.h>

int main()
{
    float DOLLAR, RUPEES;
    printf( "  NISARG PATEL \n" );
    printf( "  Enter MONEY inn DOLLAR : " );
    scanf("%f", &DOLLAR);
    RUPEES = DOLLAR *48;
    printf ( "RUPEES = %.2f\n ",RUPEES );
}
```

Program Output Screenshot

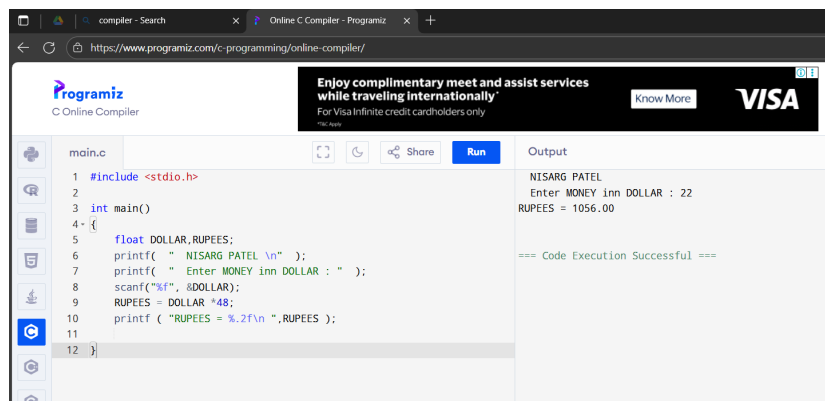


Figure 1: Program 8

QUESTION 9 : CONVERRT RUPEES INTO DOLLAR

```
#include <stdio.h>

int main()
{
    float DOLLAR, RUPEES;
    printf( "   NISARG PATEL \n" );
    printf( "   Enter MONEY inn RUPEES : " );
    scanf("%f", &RUPEES);
    DOLLAR = RUPEES /48;
    printf ( "DOLLAR = %.2f\n ", DOLLAR );
}
```

Program Output Screenshot

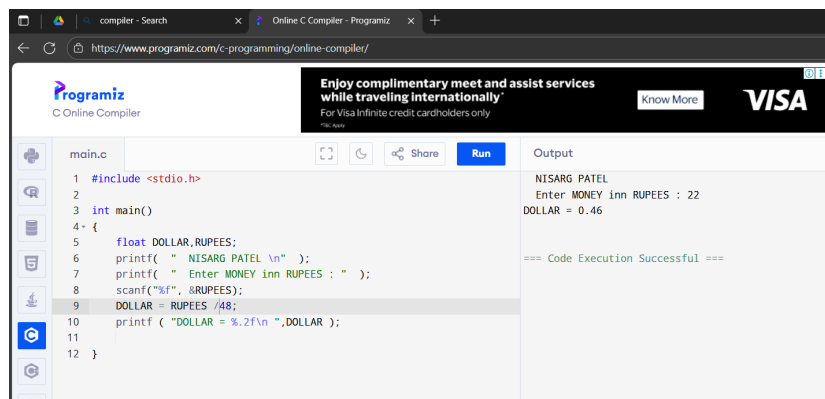


Figure 1: Program 9

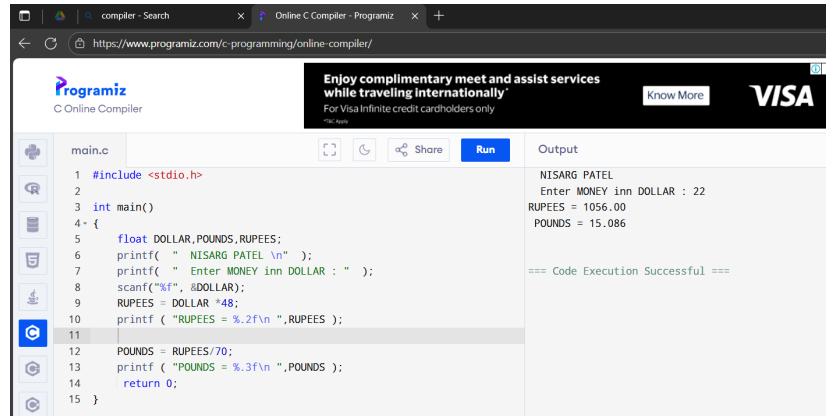
QUESTION 10 : CONVERT DOLLAR INTO POUNDS

```
#include <stdio.h>

int main()
{
    float DOLLAR, POUNDS, RUPEES;
    printf( "    NISARG PATEL \n" );
    printf( "    Enter MONEY inn DOLLAR : " );
    scanf("%f", &DOLLAR);
    RUPEES = DOLLAR *48;
    printf ( "RUPEES = %.2f\n ",RUPEES );

    POUNDS = RUPEES/70;
    printf ( "POUNDS = %.3f\n ",POUNDS );
}
```

Program Output Screenshot



The screenshot displays a web-based C compiler interface. The code editor on the left contains the following C program:

```
1 #include <stdio.h>
2
3 int main()
4 {
5     float DOLLAR, POUNDS, RUPEES;
6     printf( "    NISARG PATEL \n" );
7     printf( "    Enter MONEY inn DOLLAR : " );
8     scanf("%f", &DOLLAR);
9     RUPEES = DOLLAR *48;
10    printf ( "RUPEES = %.2f\n ",RUPEES );
11
12    POUNDS = RUPEES/70;
13    printf ( "POUNDS = %.3f\n ",POUNDS );
14    return 0;
15 }
```

The 'Output' pane on the right shows the program's execution results:

```
NISARG PATEL
Enter MONEY inn DOLLAR : 22
RUPEES = 1056.00
POUNDS = 15.086

=== Code Execution Successful ===
```

Figure 1: Program 10

QUESTION 11 : CONVERT GRAMS INTO KGS

```
#include <stdio.h>

int main()
{
    float GRAMS, KILOGRAMS;
    printf( "    NISARG PATEL \n" );
    printf( "    Enter AMOUNT inn GRAMS : " );
    scanf("%f", &GRAMS);
    KILOGRAMS = GRAMS /1000;
    printf ( "KILOGRAMS = %.3f\n ", KILOGRAMS );
}
```

Program Output Screenshot

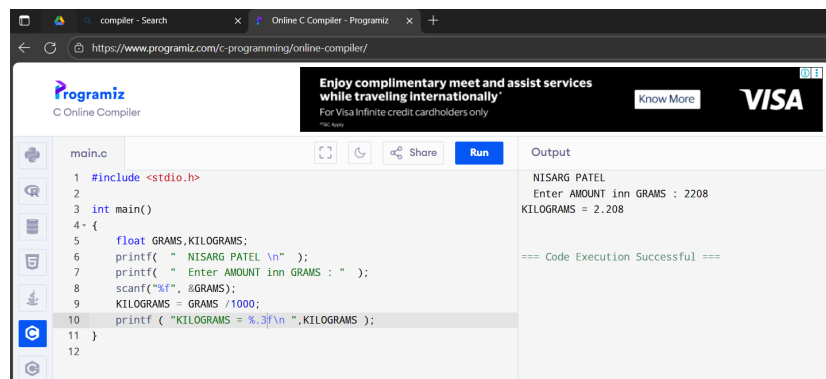


Figure 1: Program 11

QUESTION 12 : CONVERT KILOGRAMS INTO GRAMS

```
#include <stdio.h>

int main()
{
    int GRAMS, KILOGRAMS;
    printf( "    NISARG PATEL \n" );
    printf( "    Enter AMOUNT inn KILOGRAMS : " );
    scanf("%d", &KILOGRAMS);
    GRAMS = KILOGRAMS * 1000;
    printf ( "GRAMS = %d\n ", GRAMS);
}
```

Program Output Screenshot

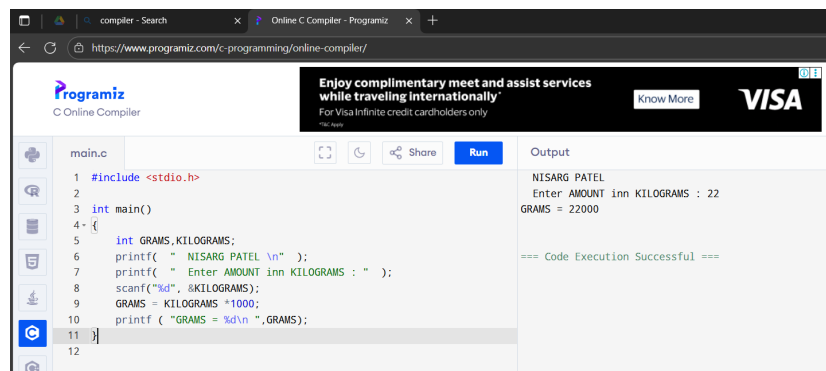


Figure 1: Program 12

QUESTION 13 :

: CONVERT BYTES INTO KB,MB,GB

```
#include <stdio.h>

int main()
{
    int BYTES;
    float KB, MB, GB;

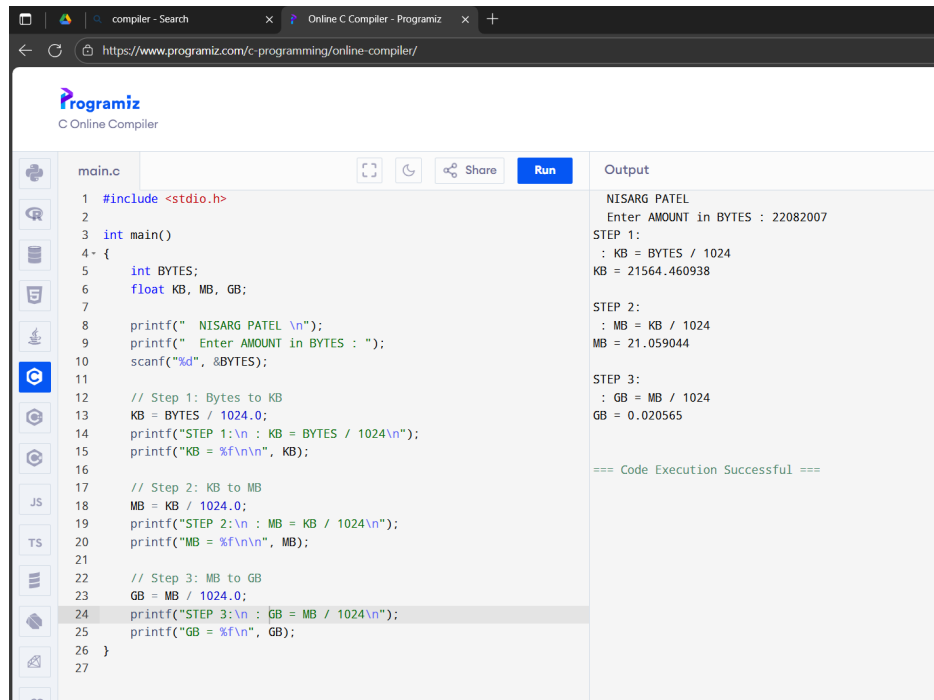
    printf("  NISARG PATEL \n");
    printf("  Enter AMOUNT in BYTES : ");
    scanf("%d", &BYTES);

    // Step 1: Bytes to KB
    KB = BYTES / 1024.0;
    printf("STEP 1:\n : KB = BYTES / 1024\n");
    printf("KB = %f\n\n", KB);

    // Step 2: KB to MB
    MB = KB / 1024.0;
    printf("STEP 2:\n : MB = KB / 1024\n");
    printf("MB = %f\n\n", MB);

    // Step 3: MB to GB
    GB = MB / 1024.0;
    printf("STEP 3:\n : GB = MB / 1024\n");
    printf("GB = %f\n", GB);
}
```

Program Output Screenshot



The screenshot displays the Programiz Online C Compiler interface. The left sidebar contains icons for various programming languages: C, C++, Java, JavaScript, Python, PHP, Ruby, Swift, Kotlin, and TypeScript. The main editor area shows a C program in a file named 'main.c'. The program includes `<stdio.h>` and defines a `main` function. It declares variables `BYTES` (int), `KB`, `MB`, and `GB` (float). The program prompts the user to enter an amount in bytes, which is stored in `BYTES`. It then performs three conversion steps: 1. Bytes to KB, 2. KB to MB, and 3. MB to GB. Each step is printed to the console. The output window on the right shows the execution results, including the user's input and the calculated values for KB, MB, and GB. The program concludes with a success message: '=== Code Execution Successful ==='.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int BYTES;
6     float KB, MB, GB;
7
8     printf(" NISARG PATEL \n");
9     printf(" Enter AMOUNT in BYTES : ");
10    scanf("%d", &BYTES);
11
12    // Step 1: Bytes to KB
13    KB = BYTES / 1024.0;
14    printf("STEP 1:\n : KB = BYTES / 1024\n");
15    printf("KB = %f\n", KB);
16
17    // Step 2: KB to MB
18    MB = KB / 1024.0;
19    printf("STEP 2:\n : MB = KB / 1024\n");
20    printf("MB = %f\n", MB);
21
22    // Step 3: MB to GB
23    GB = MB / 1024.0;
24    printf("STEP 3:\n : GB = MB / 1024\n");
25    printf("GB = %f\n", GB);
26 }
27
```

Output

```
NISARG PATEL
Enter AMOUNT in BYTES : 22082007
STEP 1:
: KB = BYTES / 1024
KB = 21564.460938

STEP 2:
: MB = KB / 1024
MB = 21.059044

STEP 3:
: GB = MB / 1024
GB = 0.020565

=== Code Execution Successful ===
```

Figure 1: Program 13

QUESTION 14 :

: CONVERT CELSIUS INTO FARENHEIT

```
#include <stdio.h>

int main()
{

    float CELSIUS, FARENHEIT;

    printf("  NISARG PATEL \n");
    printf("  Enter TEMPERATURE in CELSIUS : ");
    scanf("%f", &CELSIUS);
    FARENHEIT=((9.0/5.0)*CELSIUS)+32;
    printf("FARENHEIT=%f\n", FARENHEIT);
}
```

Program Output Screenshot

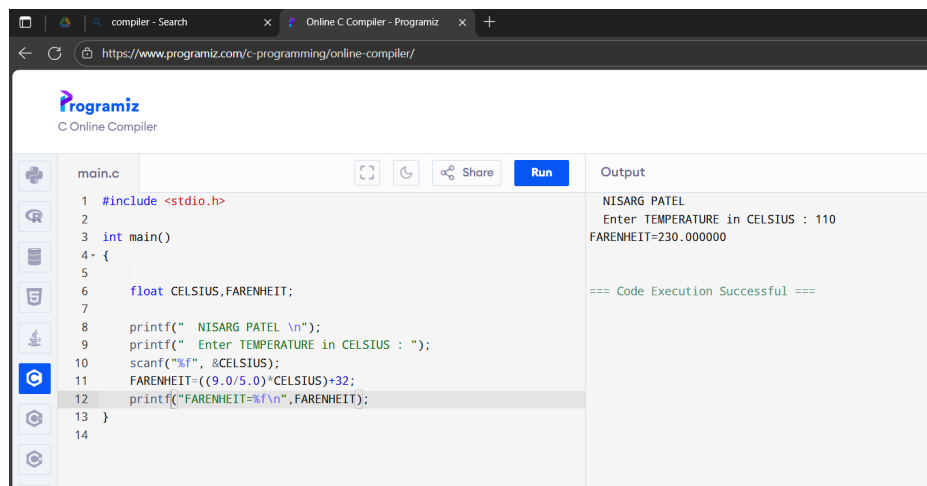


Figure 1: Program 14

QUESTION 15 :

: CONVERT FARENHEIT INTO CELSIUS

```
#include <stdio.h>

int main()
{

    float CELSIUS, FARENHEIT;

    printf("  NISARG PATEL \n");
    printf("  Enter TEMPERATURE in FARENHEIT : ");
    scanf("%f", &FARENHEIT);
    CELSIUS=(5.0/9.0)*(FARENHEIT-32);
    printf("CELSIUS=%f\n", CELSIUS);
}
```

Program Output Screenshot

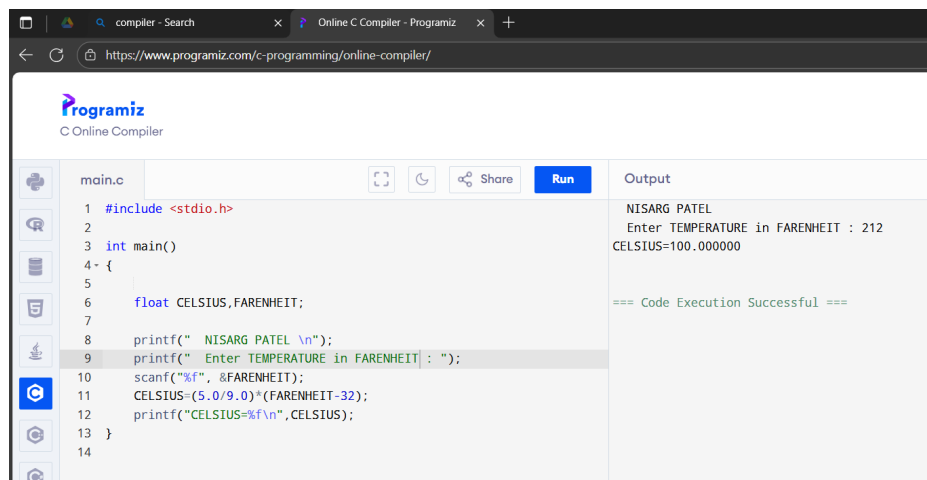


Figure 1: Program 15

QUESTION 16 :

: CALCULATE INTEREST

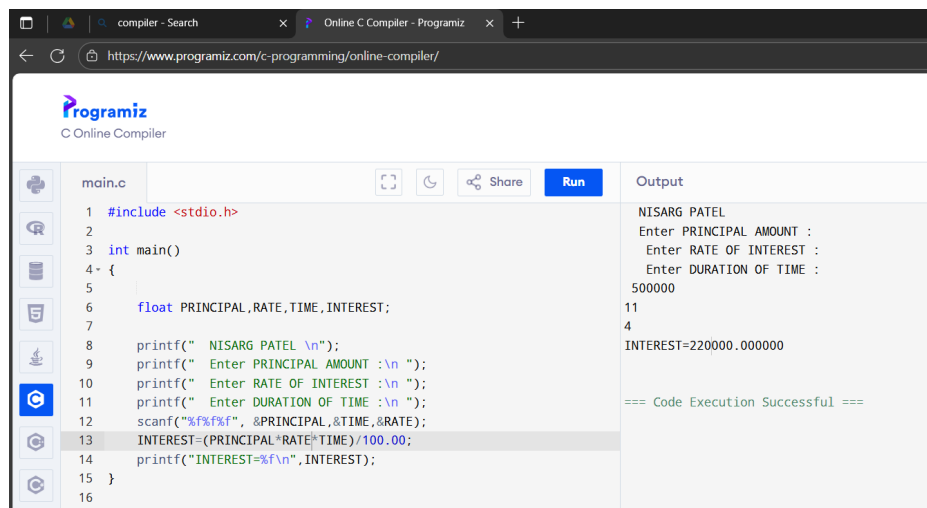
```
#include <stdio.h>

int main()
{

    float PRINCIPAL ,RATE ,TIME ,INTEREST;

    printf("  NISARG PATEL \n");
    printf("  Enter PRINCIPAL AMOUNT :\n ");
    printf("  Enter RATE OF INTEREST :\n ");
    printf("  Enter DURATION OF TIME :\n ");
    scanf("%f%f%f", &PRINCIPAL ,&TIME ,&RATE);
    INTEREST=(PRINCIPAL*RATE*TIME)/100.00;
    printf("INTEREST=%f\n",INTEREST);
}
```

Program Output Screenshot



The screenshot displays the Programiz Online C Compiler interface. The code editor on the left shows a C program that prompts the user for Principal Amount, Rate of Interest, and Duration of Time, then calculates the interest using the formula $\text{INTEREST} = (\text{PRINCIPAL} \times \text{RATE} \times \text{TIME}) / 100.00$. The output window on the right shows the program's execution with the following input and output:

```
NISARG PATEL
Enter PRINCIPAL AMOUNT :
Enter RATE OF INTEREST :
Enter DURATION OF TIME :
500000
11
4
INTEREST=220000.000000

=== Code Execution Successful ===
```

Figure 1: Program 16

QUESTION 17 :

: CALCULATE AREA AND PERIMETER OF SQUARE

```
#include <stdio.h>

int main()
{

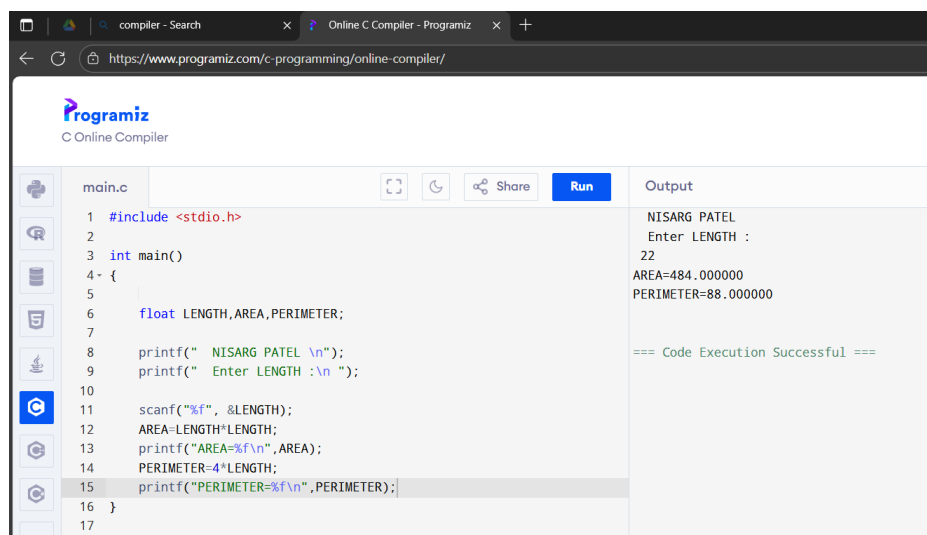
    float LENGTH, AREA, PERIMETER;

    printf("    NISARG PATEL \n");
    printf("    Enter LENGTH :\n ");

    scanf("%f", &LENGTH);
    AREA=LENGTH*LENGTH;
    printf("AREA=%f\n", AREA);
    PERIMETER=4*LENGTH;
    printf("PERIMETER=%f\n", PERIMETER);

}
```

Program Output Screenshot



The screenshot displays the Programiz Online C Compiler interface. The code editor on the left shows a C program that prompts the user for the length of a square, calculates its area and perimeter, and prints the results. The output window on the right shows the program's execution with the input length of 22, resulting in an area of 484.000000 and a perimeter of 88.000000. The execution was successful.

```
main.c
1 #include <stdio.h>
2
3 int main()
4 {
5
6     float LENGTH, AREA, PERIMETER;
7
8     printf("    NISARG PATEL \n");
9     printf("    Enter LENGTH :\n ");
10
11     scanf("%f", &LENGTH);
12     AREA=LENGTH*LENGTH;
13     printf("AREA=%f\n", AREA);
14     PERIMETER=4*LENGTH;
15     printf("PERIMETER=%f\n", PERIMETER);
16 }
17
```

Output

```
NISARG PATEL
Enter LENGTH :
22
AREA=484.000000
PERIMETER=88.000000

=== Code Execution Successful ===
```

Figure 1: Program 17

QUESTION 18 :

: CALCULATE AREA AND PERIMETER OF RECT-ANGLE

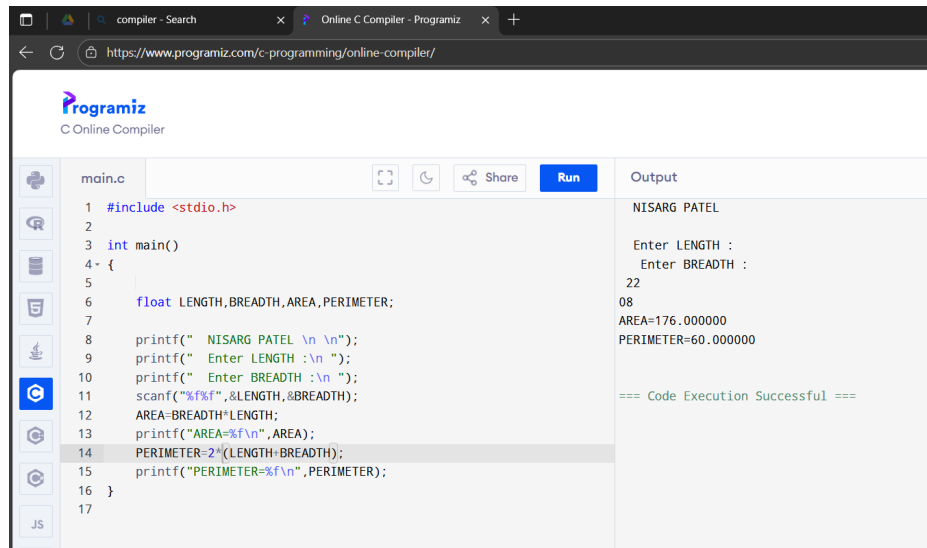
```
#include <stdio.h>

int main()
{

    float LENGTH , BREADTH , AREA , PERIMETER;

    printf("    NISARG PATEL \n \n");
    printf("    Enter LENGTH :\n ");
    printf("    Enter BREADTH :\n ");
    scanf("%f%f", &LENGTH, &BREADTH);
    AREA=BREADTH*LENGTH;
    printf("AREA=%f\n", AREA);
    PERIMETER=2*(LENGTH+BREADTH);
    printf("PERIMETER=%f\n", PERIMETER);
}
```

Program Output Screenshot



The screenshot displays the Programiz Online C Compiler interface. The code editor on the left shows a C program that prompts the user for length and breadth, calculates the area and perimeter, and prints the results. The output panel on the right shows the program's execution, including the user input and the calculated values.

```
main.c
1 #include <stdio.h>
2
3 int main()
4 {
5
6     float LENGTH, BREADTH, AREA, PERIMETER;
7
8     printf("    NISARG PATEL \n \n");
9     printf("    Enter LENGTH :\n ");
10    printf("    Enter BREADTH :\n ");
11    scanf("%f%f", &LENGTH, &BREADTH);
12    AREA=BREADTH*LENGTH;
13    printf("AREA=%f\n", AREA);
14    PERIMETER=2*(LENGTH+BREADTH);
15    printf("PERIMETER=%f\n", PERIMETER);
16 }
17
```

Output

```
NISARG PATEL

Enter LENGTH :
Enter BREADTH :
22
08
AREA=176.000000
PERIMETER=60.000000

=== Code Execution Successful ===
```

Figure 1: Program 18

QUESTION 19 :

: CALCULATE AREA OF CIRCLE

```
#include <stdio.h>

int main()
{

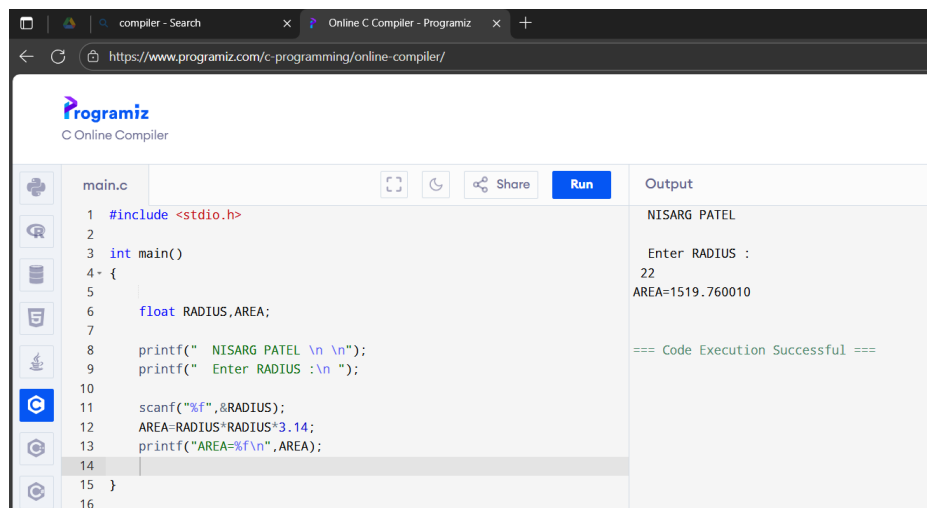
    float RADIUS, AREA;

    printf("  NISARG PATEL \n \n");
    printf("  Enter RADIUS :\n ");

    scanf("%f",&RADIUS);
    AREA=RADIUS*RADIUS*3.14;
    printf("AREA=%f\n", AREA);

}
```

Program Output Screenshot



The screenshot displays the Programiz Online C Compiler interface. The code editor on the left shows a C program named 'main.c' with the following content:

```
1 #include <stdio.h>
2
3 int main()
4 {
5
6     float RADIUS, AREA;
7
8     printf("  NISARG PATEL \n \n");
9     printf("  Enter RADIUS :\n ");
10
11     scanf("%f",&RADIUS);
12     AREA=RADIUS*RADIUS*3.14;
13     printf("AREA=%f\n", AREA);
14
15 }
16
```

The 'Run' button is highlighted in blue. The 'Output' panel on the right shows the program's execution results:

```
NISARG PATEL

Enter RADIUS :
22
AREA=1519.760010

=== Code Execution Successful ===
```

Figure 1: Program 19

QUESTION 20 :

: CALCULATE AREA OF TRIANGLE

```
#include <stdio.h>

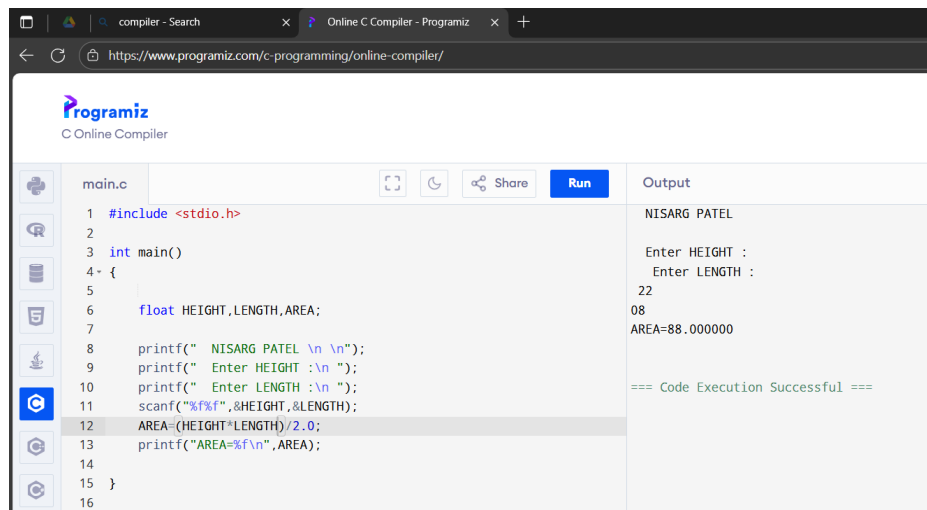
int main()
{

    float HEIGHT,LENGTH,AREA;

    printf("  NISARG PATEL \n \n");
    printf("  Enter HEIGHT :\n ");
    printf("  Enter LENGTH :\n ");
    scanf("%f%f",&HEIGHT,&LENGTH);
    AREA=(HEIGHT*LENGTH)/2.0;
    printf("AREA=%f\n",AREA);

}
```

Program Output Screenshot



The screenshot displays the Programiz Online C Compiler interface. The code editor on the left shows a C program that prints the user's name, prompts for height and length, reads the input, calculates the area, and prints the result. The output window on the right shows the program's execution with the input values 22 and 08, resulting in an area of 88.000000. The status bar at the bottom indicates 'Code Execution Successful'.

```
main.c
1 #include <stdio.h>
2
3 int main()
4 {
5
6     float HEIGHT,LENGTH,AREA;
7
8     printf("  NISARG PATEL \n \n");
9     printf("  Enter HEIGHT :\n ");
10    printf("  Enter LENGTH :\n ");
11    scanf("%f%f",&HEIGHT,&LENGTH);
12    AREA=(HEIGHT*LENGTH)/2.0;
13    printf("AREA=%f\n",AREA);
14
15 }
16
```

Output

```
NISARG PATEL
Enter HEIGHT :
Enter LENGTH :
22
08
AREA=88.000000

=== Code Execution Successful ===
```

Figure 1: Program 20

QUESTION 21 :

: CALCULATE NET SALARY (ALLOWANCE 10 PER)(DEDUCTION 3 PER)

```
#include <stdio.h>

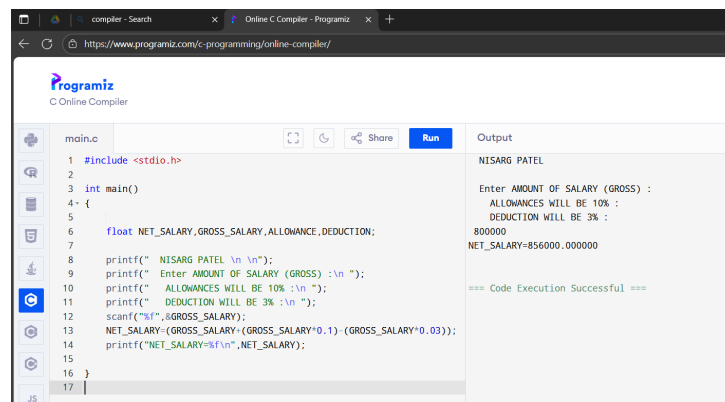
int main()
{

    float NET_SALARY , GROSS_SALARY , ALLOWANCE , DEDUCTION ;

    printf("    NISARG PATEL \n \n");
    printf("    Enter AMOUNT OF SALARY (GROSS) :\n ");
    printf("    ALLOWANCES WILL BE 10% :\n ");
    printf("    DEDUCTION WILL BE 3% :\n ");
    scanf("%f",&GROSS_SALARY);
    NET_SALARY=(GROSS_SALARY+(GROSS_SALARY*0.1)-(GROSS_SALARY*0.03));
    printf("NET_SALARY=%f\n",NET_SALARY);

}
```

Program Output Screenshot



The screenshot shows the Programiz online compiler interface. On the left, the code editor displays the C program. On the right, the output window shows the program's execution results. The output includes the name 'NISARG PATEL', prompts for salary and deductions, the input value '800000', and the calculated net salary '856000.000000'. A success message '=== Code Execution Successful ===' is also present.

```
main.c
1 #include <stdio.h>
2
3 int main()
4 {
5
6     float NET_SALARY , GROSS_SALARY , ALLOWANCE , DEDUCTION ;
7
8     printf("    NISARG PATEL \n \n");
9     printf("    Enter AMOUNT OF SALARY (GROSS) :\n ");
10    printf("    ALLOWANCES WILL BE 10% :\n ");
11    printf("    DEDUCTION WILL BE 3% :\n ");
12    scanf("%f",&GROSS_SALARY);
13    NET_SALARY=(GROSS_SALARY+(GROSS_SALARY*0.1)-(GROSS_SALARY*0.03));
14    printf("NET_SALARY=%f\n",NET_SALARY);
15
16 }
17
```

Output

```
NISARG PATEL
Enter AMOUNT OF SALARY (GROSS) :
ALLOWANCES WILL BE 10% :
DEDUCTION WILL BE 3% :
800000
NET_SALARY=856000.000000

=== Code Execution Successful ===
```

Figure 1: Program 21

QUESTION 22 :

: CALCULATE NET SALES (DISCOUNT 10 PER)

```
#include <stdio.h>

int main()
{

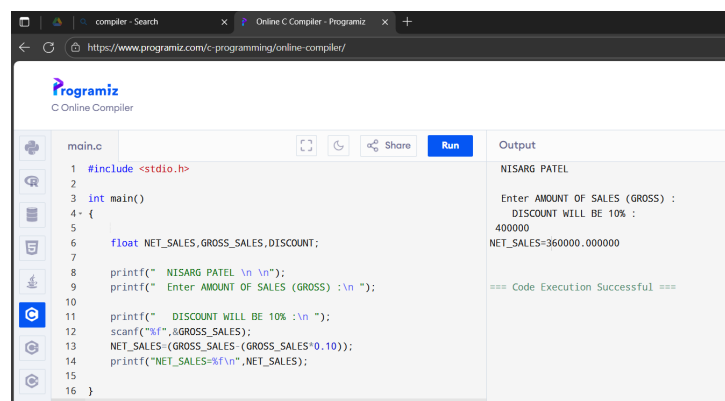
    float NET_SALES ,GROSS_SALES ,DISCOUNT;

    printf("  NISARG PATEL \n \n");
    printf("  Enter AMOUNT OF SALES (GROSS) :\n ");

    printf("  DISCOUNT WILL BE 10% :\n ");
    scanf("%f",&GROSS_SALES);
    NET_SALES=(GROSS_SALES-(GROSS_SALES*0.10));
    printf("NET_SALES=%f\n",NET_SALES);

}
```

Program Output Screenshot



The screenshot shows the Programiz Online C Compiler interface. The code editor on the left contains the C program for calculating net sales with a 10% discount. The output window on the right shows the program's execution results, including the name 'NISARG PATEL', the prompt for gross sales, the discount rate, the entered gross sales value of 400000, and the calculated net sales of 360000.000000. A success message 'Code Execution Successful' is also displayed.

```
main.c
1 #include <stdio.h>
2
3 int main()
4 {
5
6     float NET_SALES,GROSS_SALES,DISCOUNT;
7
8     printf("  NISARG PATEL \n \n");
9     printf("  Enter AMOUNT OF SALES (GROSS) :\n ");
10
11    printf("  DISCOUNT WILL BE 10% :\n ");
12    scanf("%f",&GROSS_SALES);
13    NET_SALES=(GROSS_SALES-(GROSS_SALES*0.10));
14    printf("NET_SALES=%f\n",NET_SALES);
15
16 }
17
```

Output

```
NISARG PATEL
Enter AMOUNT OF SALES (GROSS) :
DISCOUNT WILL BE 10% :
400000
NET_SALES=360000.000000

=== Code Execution Successful ===
```

Figure 1: Program 22

QUESTION 23 :

: CALCULATE AVERAGE AND TOTAL FOR 3 SUB

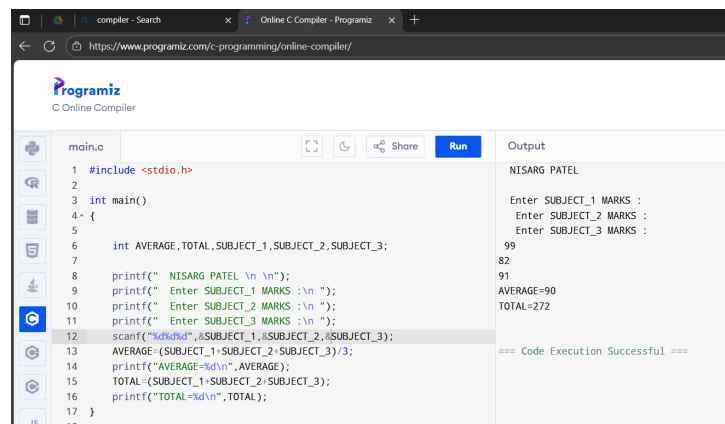
```
#include <stdio.h>

int main()
{

    int AVERAGE , TOTAL , SUBJECT_1 , SUBJECT_2 , SUBJECT_3;

    printf("  NISARG PATEL \n \n");
    printf("  Enter SUBJECT_1 MARKS : \n ");
    printf("  Enter SUBJECT_2 MARKS : \n ");
    printf("  Enter SUBJECT_3 MARKS : \n ");
    scanf("%d%d%d",&SUBJECT_1,&SUBJECT_2,&SUBJECT_3);
    AVERAGE=(SUBJECT_1+SUBJECT_2+SUBJECT_3)/3;
    printf("AVERAGE=%d\n",AVERAGE);
    TOTAL=(SUBJECT_1+SUBJECT_2+SUBJECT_3);
    printf("TOTAL=%d\n",TOTAL);
}
```

Program Output Screenshot



The screenshot shows the Programiz Online C Compiler interface. The code editor on the left contains the C program for calculating the average and total of three subjects. The output window on the right shows the execution results, including the name 'NISARG PATEL', prompts for subject marks, the input values 99, 82, and 91, the calculated average of 90, and the total of 272. The status bar at the bottom indicates 'Code Execution Successful'.

```
main.c
1 #include <stdio.h>
2
3 int main()
4 {
5
6     int AVERAGE , TOTAL , SUBJECT_1 , SUBJECT_2 , SUBJECT_3;
7
8     printf("  NISARG PATEL \n \n");
9     printf("  Enter SUBJECT_1 MARKS : \n ");
10    printf("  Enter SUBJECT_2 MARKS : \n ");
11    printf("  Enter SUBJECT_3 MARKS : \n ");
12    scanf("%d%d%d",&SUBJECT_1,&SUBJECT_2,&SUBJECT_3);
13    AVERAGE=(SUBJECT_1+SUBJECT_2+SUBJECT_3)/3;
14    printf("AVERAGE=%d\n",AVERAGE);
15    TOTAL=(SUBJECT_1+SUBJECT_2+SUBJECT_3);
16    printf("TOTAL=%d\n",TOTAL);
17 }
```

Output

```
NISARG PATEL
Enter SUBJECT_1 MARKS :
Enter SUBJECT_2 MARKS :
Enter SUBJECT_3 MARKS :
99
82
91
AVERAGE=90
TOTAL=272
=== Code Execution Successful ===
```

Figure 1: Program 23

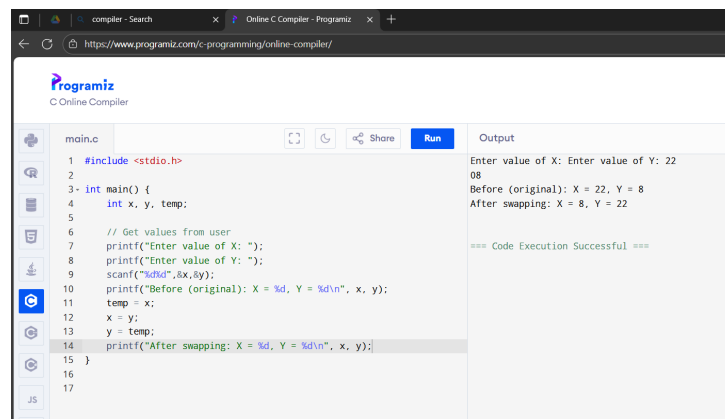
QUESTION 24 :

: SWAPPING OF TWO VALUES

```
#include <stdio.h>

int main() {
    int x, y, temp;
    printf("Enter value of X: ");
    printf("Enter value of Y: ");
    scanf("%d%d",&x,&y);
    printf("Before (original): X = %d, Y = %d\n", x, y);
    temp = x;
    x = y;
    y = temp;
    printf("After SWAPPING : X = %d, Y = %d\n", x, y);
}
```

Program Output Screenshot



The screenshot displays the Programiz Online Compiler interface. On the left, the code editor shows the C program for swapping two integers. The code includes a header file, declares variables, prompts for input, prints the original values, performs the swap using a temporary variable, and prints the swapped values. On the right, the output panel shows the program's execution with user input of 22 and 8, resulting in the swapped values 8 and 22. A success message "Code Execution Successful" is also visible.

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int x, y, temp;
5
6     // Get values from user
7     printf("Enter value of X: ");
8     printf("Enter value of Y: ");
9     scanf("%d%d",&x,&y);
10    printf("Before (original): X = %d, Y = %d\n", x, y);
11    temp = x;
12    x = y;
13    y = temp;
14    printf("After swapping: X = %d, Y = %d\n", x, y);
15 }
16
17
```

Output

```
Enter value of X: Enter value of Y: 22
08
Before (original): X = 22, Y = 8
After swapping: X = 8, Y = 22

=== Code Execution Successful ===
```

Figure 1: Program 24