

--- . .qQ1

START

DECLARE num1, num2, sum AS INTEGER

DISPLAY "Enter first number: "

READ num1

DISPLAY "Enter second number: "

READ num2

sum \leftarrow num1 + num2

DISPLAY "Sum = ", sum

END

C:\Users\hp\Desktop\New folder\day 1 q1.cpp - Embarcadero Dev-C++ 6.3

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 9.2.0 64-bit Release

(globals)

Project C day 1 q1.cpp

```
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2, sum;
5
6     // Input two numbers
7     printf("Enter first number: ");
8     scanf("%d", &num1);
9
10    printf("Enter second number: ");
11    scanf("%d", &num2);
12
13    // Calculate sum
14    sum = num1 + num2;
15
16    // Display result
17    printf("Sum = %d\n", sum);
18
19    return 0;
20}
```

Compiler Resources Compile Log Debug Find Results Console

Line: 1 Col: 1 Sel: 0 Lines: 21 Length: 342 Insert Done parsing in 0.031 seconds

USD/INR +0.29% 9:19 PM 10/30/2025

C:\Users\hp\Desktop\New folder\day 1 q1.exe

```
Enter first number: 20
Enter second number: 30
Sum = 50

-----
Process exited after 8.922 seconds with return value 0
Press any key to continue . . .
```

Step 1: Start

Step 2: Declare three integer variables: num1, num2, and sum

Step 3: Display message "Enter first number:"

Step 4: Read the first number and store it in num1

Step 5: Display message "Enter second number:"

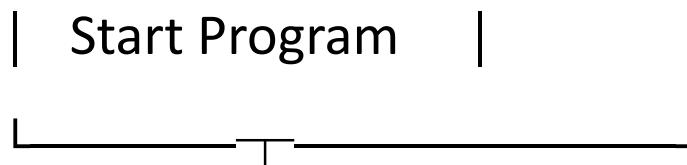
Step 6: Read the second number and store it in num2

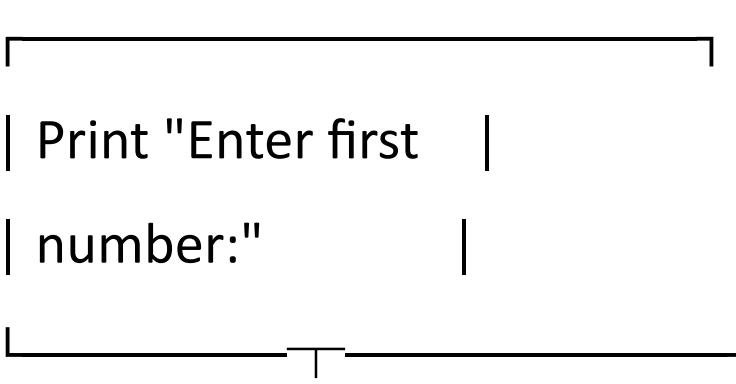
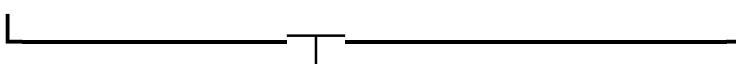
Step 7: Calculate the sum → sum = num1 + num2

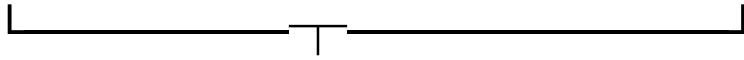
Step 8: Display the result → "Sum = ", followed by the value of sum

Step 9: Stop

flowchart

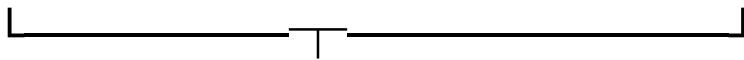




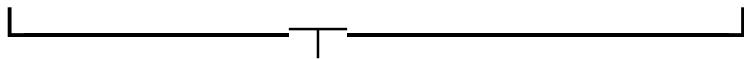


| Print "Enter second |

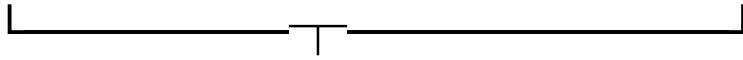
| number:" |



| Read num2 (scanf) |

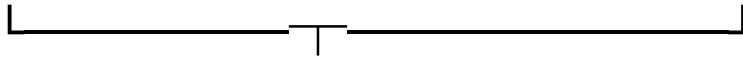


| sum = num1 + num2 |



| Print "Sum = " and |

| display sum |



| End Program |



Q2

```
C:\Users\hp\Desktop\New folder\day 1 q2.cpp - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
Project C day 1 q2.cpp
1 #include <stdio.h>
2
3 int main() {
4     float num1, num2;
5     float sum, difference, product, quotient;
6
7     printf("Enter first number: ");
8     scanf("%f", &num1);
9
10    printf("Enter second number: ");
11    scanf("%f", &num2);
12
13    sum = num1 + num2;
14    difference = num1 - num2;
15    product = num1 * num2;
16
17    if (num2 != 0) {
18        quotient = num1 / num2;
19        printf("\nSum = %.2f", sum);
20        printf("\nDifference = %.2f", difference);
21        printf("\nProduct = %.2f", product);
22        printf("\nQuotient = %.2f\n", quotient);
23    } else {
24        printf("\nSum = %.2f", sum);
25        printf("\nDifference = %.2f", difference);
26        printf("\nProduct = %.2f", product);
27        printf("\nDivision by zero is not allowed.\n");
28    }
29
30    return 0;
31
32
33
34 }
```

Compiler (2) Resources Compile Log Debug Find Results Console

Line: 33 Col: 14 Sel: 0 Lines: 35 Length: 817 Insert Done parsing in 0.016 seconds

18°C Clear Search

11:09 PM 10/30/2025

Algorithm

Step 1: Start

Step 2: Declare variables num1, num2, sum, difference, product, quotient as float.

Step 3: Prompt the user to enter the first number and store it in num1.

Step 4: Prompt the user to enter the second number and store it in num2.

Step 5: Compute

$$\text{sum} = \text{num1} + \text{num2}$$

`difference = num1 - num2`

`product = num1 * num2`

Step 6: Check if `num2 != 0`:

If true:

- Compute `quotient = num1 / num2`

- Display sum, difference, product, and quotient

Else:

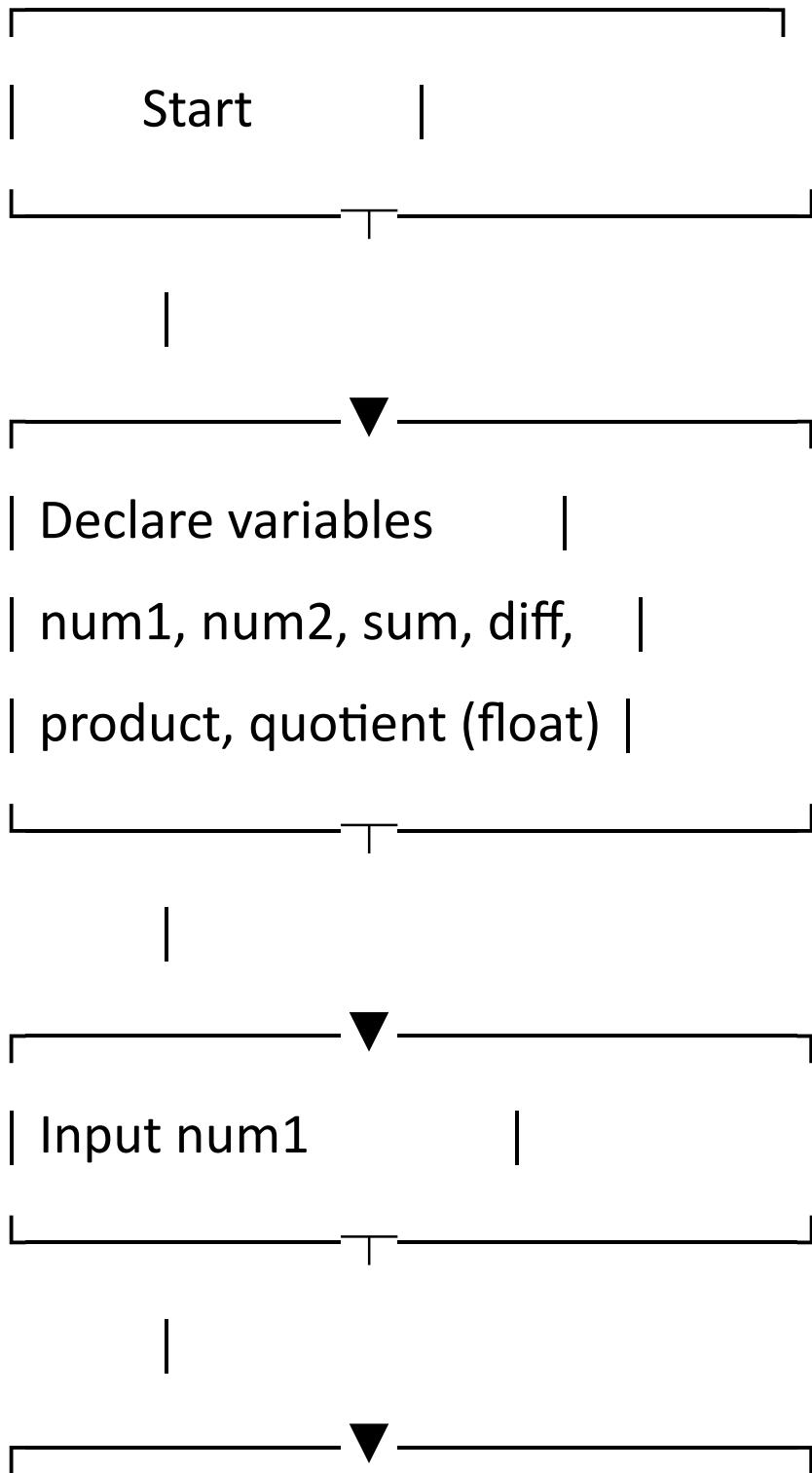
- Display sum, difference, product

- Display “Division by zero is not allowed.”

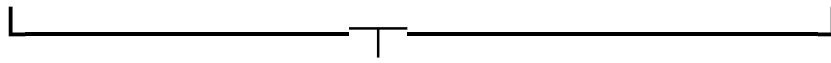
Step 7: Stop

```
C:\Users\hp\Desktop\New folder\day 1 q2.exe
Enter first number: 7
Enter second number: 5
Sum = 12.00
Difference = 2.00
Product = 35.00
Quotient = 1.40
-----
Process exited after 6.518 seconds with return value 0
Press any key to continue . . .
```

Flowchart



| Input num2 |



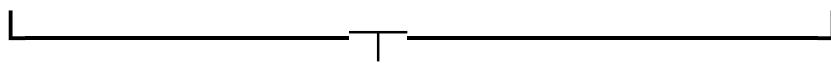
|



| sum = num1 + num2 |

| diff = num1 - num2 |

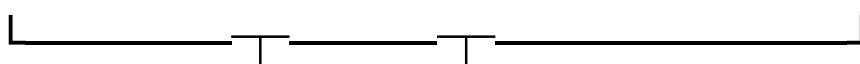
| product = num1 * num2 |



|



| Is num2 ≠ 0 ? |



| Yes | No



| quotient= |



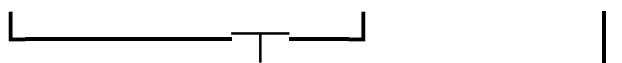
| num1/num2 |

| Display |

| sum, diff|

| product, |

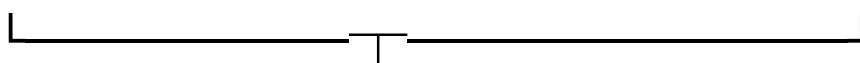
| quotient |



| Display “Division by zero |

| not allowed.” |

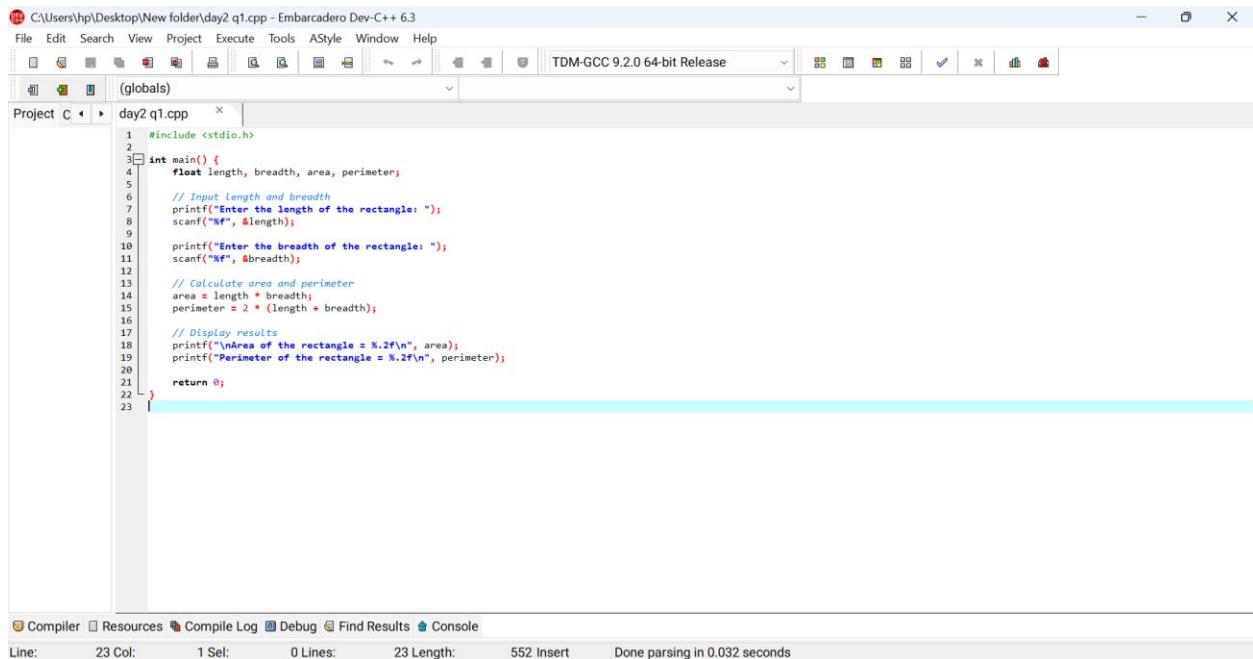
| (and sum, diff, product) |



| Stop |



Q3



```
C:\Users\hp\Desktop\New folder\day2 q1.cpp - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
Project C > day2 q1.cpp (globals)
1 #include <stdio.h>
2
3 int main() {
4     float length, breadth, area, perimeter;
5
6     // Input Length and breadth
7     printf("Enter the length of the rectangle: ");
8     scanf("%f", &length);
9
10    printf("Enter the breadth of the rectangle: ");
11    scanf("%f", &breadth);
12
13    // Calculate area and perimeter
14    area = length * breadth;
15    perimeter = 2 * (length + breadth);
16
17    // Display results
18    printf("\nArea of the rectangle = %.2f\n", area);
19    printf("Perimeter of the rectangle = %.2f\n", perimeter);
20
21    return 0;
22
23 }
```

Compiler Resources Compile Log Debug Find Results Console

Line: 23 Col: 1 Sel: 0 Lines: 23 Length: 552 Insert Done parsing in 0.032 seconds

Step 1: Start

Step 2: Declare variables length, breadth, area, and perimeter as float.

Step 3: Prompt the user to enter the length of the rectangle and store it in length.

Step 4: Prompt the user to enter the breadth of the rectangle and store it in breadth.

Step 5: Calculate the area using the formula:

$$\text{area} = \text{length} * \text{breadth}$$

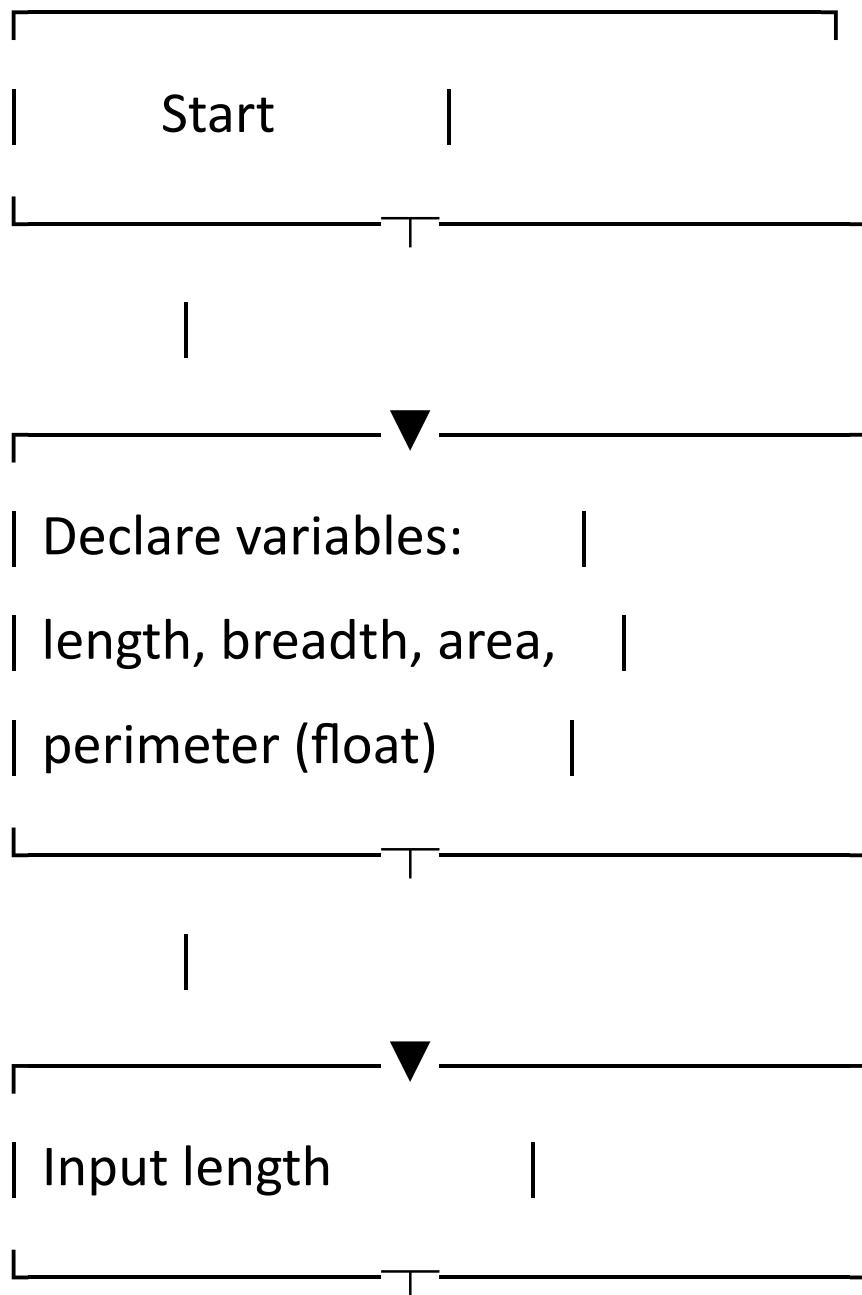
Step 6: Calculate the perimeter using the formula:

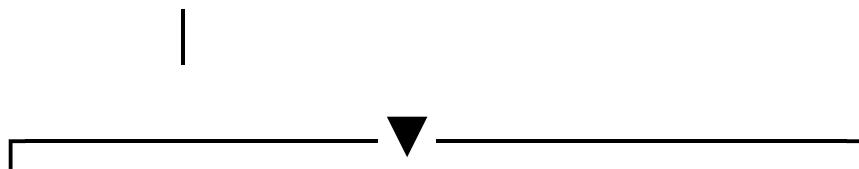
$$\text{perimeter} = 2 * (\text{length} + \text{breadth})$$

Step 7: Display the area and perimeter.

Step 8: Stop

flowchart



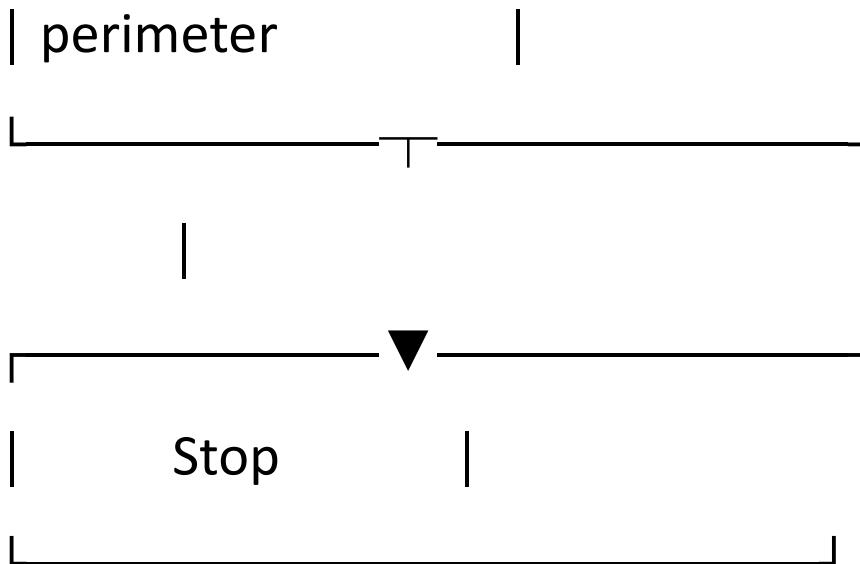


| area = length * breadth |

| perimeter = 2*(length + |

| breadth) |

| Display area and |



```
Enter the length of the rectangle: 7  
Enter the breadth of the rectangle: 8  
  
Area of the rectangle = 56.00  
Perimeter of the rectangle = 30.00  
  
-----  
Process exited after 9.287 seconds with return value 0  
Press any key to continue . . .
```

Q3 end

Q4

START

```
DECLARE radius, area, circumference AS float
```

```
SET PI = 3.14159
```

```
PRINT "Enter the radius of the circle: "
```

```
READ radius
```

```
CALCULATE area = PI * radius * radius  
CALCULATE circumference = 2 * PI * radius  
  
PRINT "Area of the circle = ", area (formatted to 2 decimal places)  
PRINT "Circumference of the circle = ", circumference (formatted to 2 decimal places)
```

END

```
1 #include <stdio.h>  
2  
3 int main() {  
4     float radius, area, circumference;  
5     const float PI = 3.14159;  
6  
7     // Input radius  
8     printf("Enter the radius of the circle: ");  
9     scanf("%f", &radius);  
0  
1     // Calculate area and circumference  
2     area = PI * radius * radius;  
3     circumference = 2 * PI * radius;  
4  
5     // Display results  
6     printf("\nArea of the circle = %.2f\n", area);  
7     printf("Circumference of the circle = %.2f\n", circumference);  
8  
9     return 0;  
0 }
```

```
Enter the radius of the circle: 7
```

```
Area of the circle = 153.94  
Circumference of the circle = 43.98
```

```
-----  
Process exited after 2.817 seconds with return value 0  
Press any key to continue . . .
```

Algorithm

Step 1: Start

Step 2: Declare variables radius, area, circumference, and constant PI = 3.14159.

Step 3: Prompt the user to enter the radius of the circle.

Step 4: Read the radius value from user input.

Step 5: Calculate the **area** using the formula

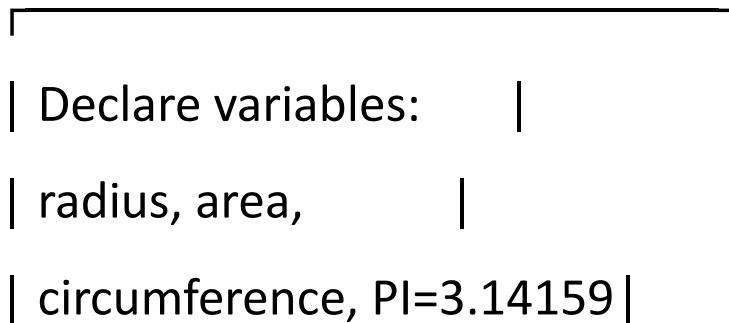
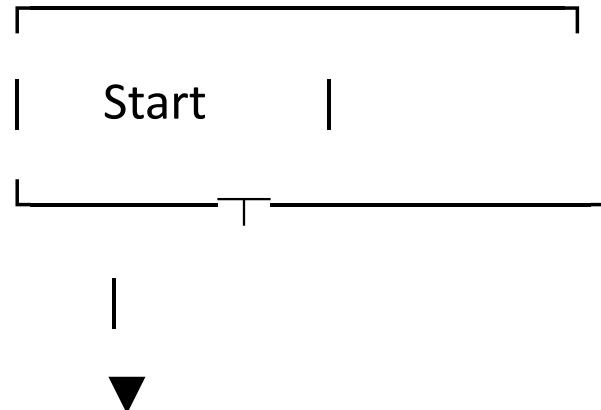
$$\text{area} = \text{PI} * \text{radius} * \text{radius}$$

Step 6: Calculate the **circumference** using the formula

$$\text{circumference} = 2 * \text{PI} * \text{radius}$$

Step 7: Display the area and circumference.

Step 8: Stop



```
|
```

```
|
```



```
|
```

```
| Input radius from user |
```

```
|
```

```
|
```



```
|
```

```
| area = PI * radius * radius |
```

```
|
```

```
|
```



```
|
```

```
| circumference = 2 * PI * |
```

```
| radius |
```

```
|
```

```
|
```



```
|-----|
```

| Display area and |
| circumference |

```
|-----|
```



```
|-----|
```

| Stop |

```
|-----|
```

Q5

```
1 #include <stdio.h>
2
3 int main() {
4     float celsius, fahrenheit;
5
6     // Input temperature in Celsius
7     printf("Enter temperature in Celsius: ");
8     scanf("%f", &celsius);
9
10    // Convert Celsius to Fahrenheit
11    fahrenheit = (celsius * 9 / 5) + 32;
12
13    // Display the result
14    printf("Temperature in Fahrenheit: %.2f\n", fahrenheit);
15
16    return 0;
17 }
```

Step 1: Start

Step 2: Declare two float variables — celsius and fahrenheit

Step 3: Prompt the user to enter the temperature in Celsius

Step 4: Read the input value and store it in the variable celsius

Step 5: Calculate Fahrenheit using the formula:

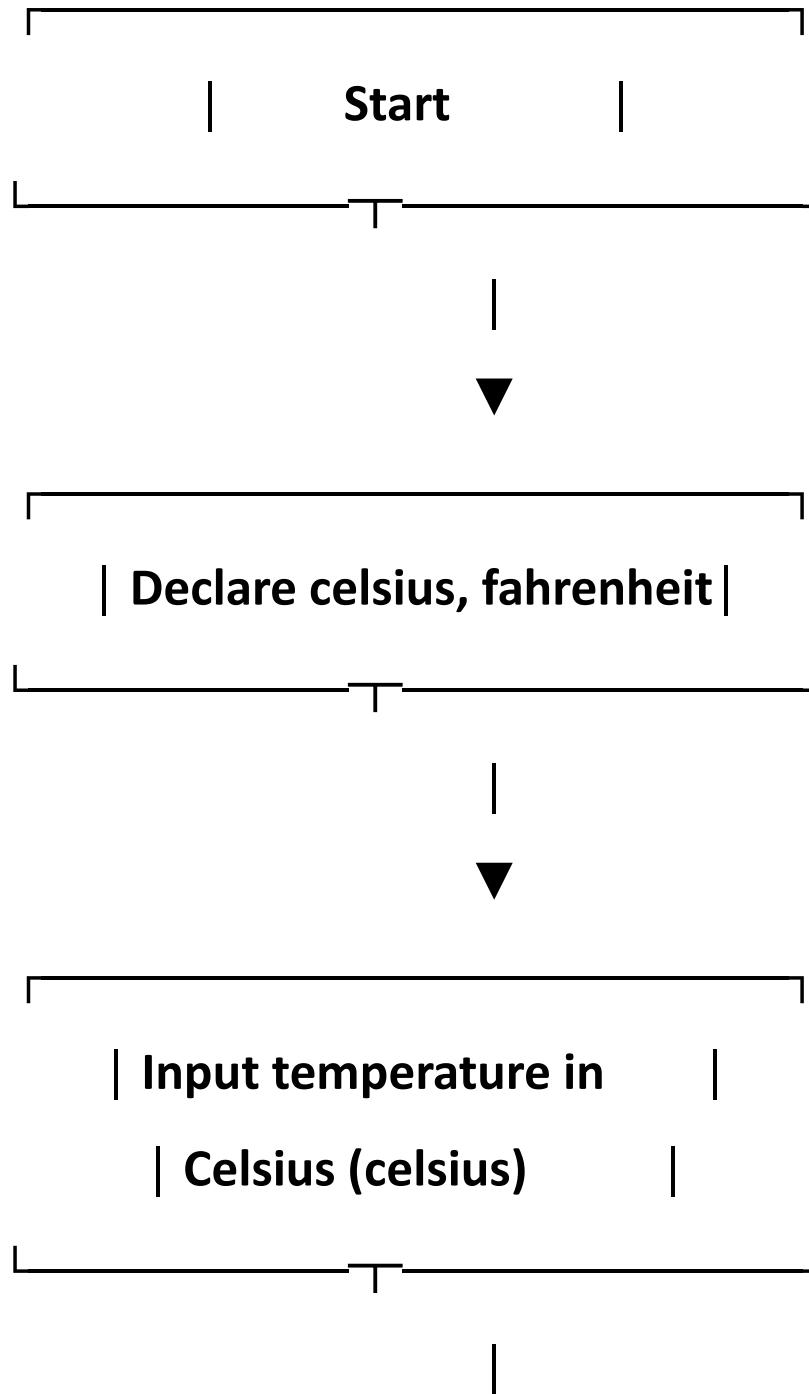
$$\text{fahrenheit} = (\text{celsius} * 9 / 5) + 32$$

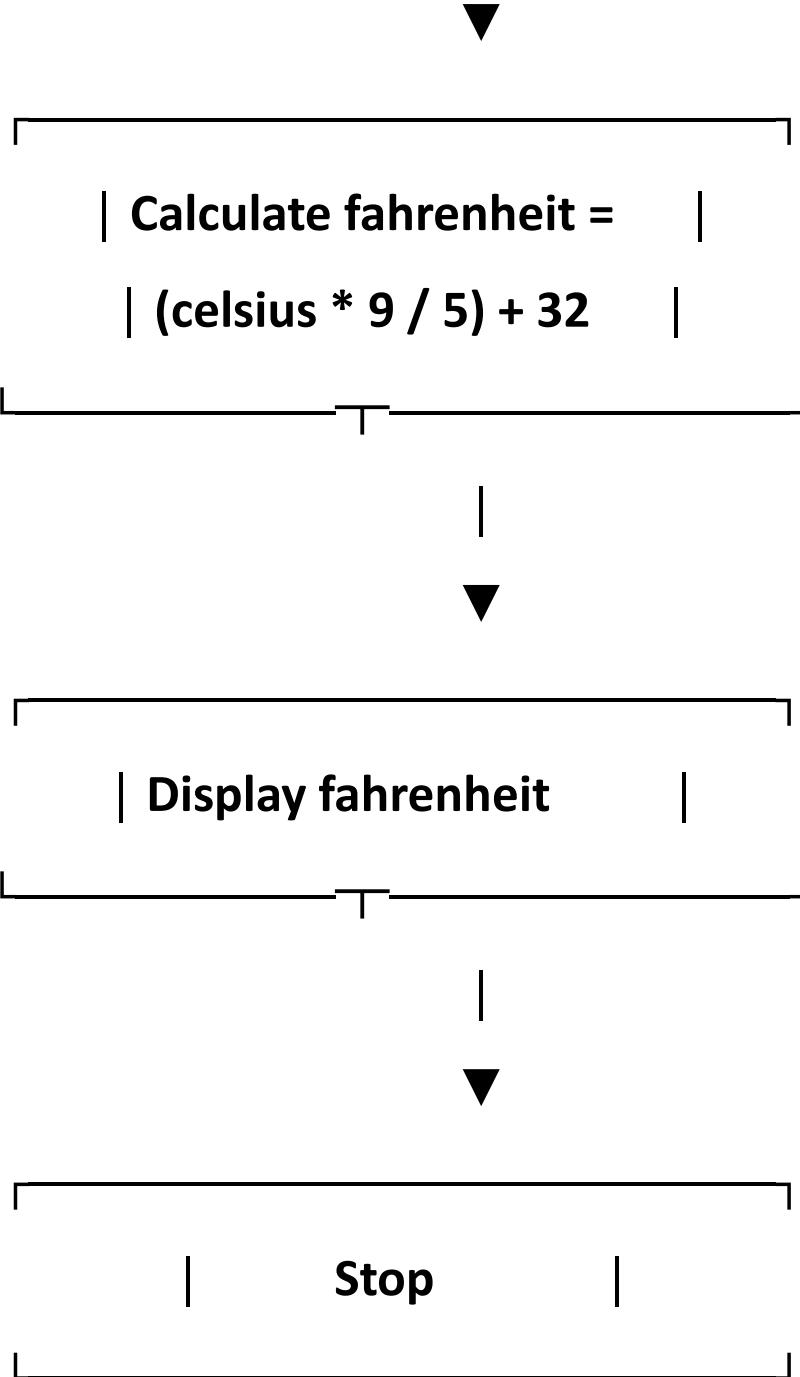
Step 6: Display the result (Fahrenheit value) on the screen

Step 7: Stop

```
Enter temperature in Celsius: 27
Temperature in Fahrenheit: 80.60
```

```
-----
Process exited after 7.787 seconds with return value 0
Press any key to continue . . .
```





Q6

```
1 #include <stdio.h>
2
3 int main() {
4     int a, b, temp;
5
6     // Input two numbers
7     printf("Enter first number: ");
8     scanf("%d", &a);
9
10    printf("Enter second number: ");
11    scanf("%d", &b);
12
13    // Display numbers before swapping
14    printf("\nBefore swapping: a = %d, b = %d\n", a, b);
15
16    // Swap using a third variable
17    temp = a;
18    a = b;
19    b = temp;
20
21    // Display numbers after swapping
22    printf("After swapping: a = %d, b = %d\n", a, b);
23
24    return 0;
25 }
26
```

```
Enter first number: 20
Enter second number: 31

Before swapping: a = 20, b = 31
After swapping: a = 31, b = 20

-----
Process exited after 11.65 seconds with return value 0
Press any key to continue . . .
```

