



main.cpp



Run

```
1 #include <iostream>
2
3 int factorial(int n) {
4     if (n == 0 || n == 1)
5         return 1;
6     else
7         return n * factorial(n - 1);
8 }
9
10 int main() {
11     int num;
12     std::cout << "Enter a number: ";
13     std::cin >> num;
14     std::cout << "Factorial of " << num << " is " << factorial(num) << std
        ::endl;
15     return 0;
16 }
```

Output

Clear

```
/tmp/Bqow4G9rAc.o
Enter a number: 5
Factorial of 5 is 120
```

main.cpp

Run

Output

Clear

```
1 //prime or not
2
3 #include <iostream>
4 #include <cmath>
5
6 bool isPrime(int n) {
7     if (n <= 1)
8         return false;
9     for (int i = 2; i <= std::sqrt(n); i++)
10         if (n % i == 0)
11             return false;
12     return true;
13 }
14
15 int main() {
16     int num;
17     std::cout << "Enter a number: ";
18     std::cin >> num;
19     if (isPrime(num))
20         std::cout << num << " is a prime number." << std::endl;
21     else
22         std::cout << num << " is not a prime number." << std::endl;
23     return 0;
24 }
```

```
/tmp/Bqow4G9rAc.o
Enter a number: 3
3 is a prime number
```

main.cpp




























Run

Output



Clear

```
1 //reverse of a string
2
3 #include <iostream>
4 #include <string>
5
6 std::string reverseString(std::string str) {
7     std::string reversed = "";
8     for (int i = str.length() - 1; i >= 0; i--)
9         reversed += str[i];
10    return reversed;
11 }
12
13 int main() {
14     std::string input;
15     std::cout << "Enter a string: ";
16     std::getline(std::cin, input);
17     std::cout << "Reversed string: " << reverseString(input) << std::endl;
18     return 0;
19 }
```

/tmp/Bqow4G9rAc.o
Enter a string: good
Reversed string: doog



main.cpp



Run

```
1 //MAX and MIN
2
3 #include <iostream>
4
5 void minMax(int arr[], int size, int& min, int& max) {
6     min = arr[0];
7     max = arr[0];
8     for (int i = 1; i < size; i++) {
9         if (arr[i] < min)
10             min = arr[i];
11         if (arr[i] > max)
12             max = arr[i];
13     }
14 }
15
16 int main() {
17     int arr[] = {3, 7, 12, 9, 5};
18     int size = sizeof(arr) / sizeof(arr[0]);
19     int min, max;
20     minMax(arr, size, min, max);
21     std::cout << "Minimum element: " << min << ", Maximum element: " << max
22         << std::endl;
23     return 0;
24 }
```

Output

Clear

```
/tmp/Bqow4G9rAc.o
Minimum element: 3, Maximum element: 12
```

The screenshot displays a C++ development environment with two main panels: a code editor on the left and an output console on the right.

Code Editor:

- File Name:** main.cpp
- Language:** C++ (indicated by the icon)
- Code Content:**




























```
1 //GCD
2
3 #include <iostream>
4
5 int gcd(int a, int b) {
6     if (b == 0)
7         return a;
8     return gcd(b, a % b);
9 }
10
11 int main() {
12     int num1, num2;
13     std::cout << "Enter two numbers: ";
14     std::cin >> num1 >> num2;
15     std::cout << "GCD of " << num1 << " and " << num2 << " is " << gcd(num1
        , num2) << std::endl;
16     return 0;
17 }
```

Output Console:



- Action:** Run (button)
- Path:** /tmp/Bqow4G9rAc.o
- Execution Output:**

```
Enter two numbers: 2 4
GCD of 2 and 4 is 2
```
- Action:** Clear (button)

A vertical toolbar on the far left contains icons for various file operations and language-specific features like JS, php, and Go.



main.cpp

















Run

```
1 //Number of words in a string
2
3 #include <iostream>
4 #include <string>
5
6 int countElements(const std::string& str) {
7     int count = 0;
8     for (char c : str)
9         if (c != ' ')
10             count++;
11     return count;
12 }
13
14 int main() {
15     std::string input;
16     std::cout << "Enter a string: ";
17     std::getline(std::cin, input);
18     std::cout << "Number of elements in the string: " << countElements
19         (input) << std::endl;
20     return 0;
21 }
```



Output

Clear

```
/tmp/K963E8HJX1.o
Enter a string: programming language
Number of elements in the string:2
```



main.cpp



Run

```
1 //Convert Celsius and Fahrenheit
2
3 #include <iostream>
4
5 double celsiusToFahrenheit(double celsius) {
6     return (celsius * 9 / 5) + 32;
7 }
8
9 double fahrenheitToCelsius(double fahrenheit) {
10    return (fahrenheit - 32) * 5 / 9;
11 }
12
13 int main() {
14     double celsius, fahrenheit;
15     std::cout << "Enter temperature in Celsius: ";
16     std::cin >> celsius;
17     std::cout << "Temperature in Fahrenheit: " << celsiusToFahrenheit
        (celsius) << std::endl;
18
19     std::cout << "\nEnter temperature in Fahrenheit: ";
20     std::cin >> fahrenheit;
21     std::cout << "Temperature in Celsius: " << fahrenheitToCelsius
        (fahrenheit) << std::endl;
22     return 0;
23 }
```

Output

Clear

```
/tmp/I7LDqQ6QVf.o
Enter temperature in Celsius: 34
Temperature in Fahrenheit:93.2
```

main.cpp

Run

```
1 //area of a circle
2
3 #include <iostream>
4 #include <cmath>
5
6 double circleArea(double radius) {
7     return M_PI * radius * radius;
8 }
9
10 int main() {
11     double radius;
12     std::cout << "Enter the radius of the circle: ";
13     std::cin >> radius;
14     std::cout << "Area of the circle: " << circleArea(radius) << std::endl;
15     return 0;
16 }
17
```

Output

Clear

```
/tmp/I7LDqQ6QVf.o
Enter the radius of the circle: 2
Area of the circle:12.56
```


main.cpp

Output

```
1 //string is palindrome or not
2
3 #include <iostream>
4 #include <string>
5
6 bool isPalindrome(const std::string& str) {
7     int start = 0;
8     int end = str.length() - 1;
9     while (start < end) {
10         if (str[start] != str[end])
11             return false;
12         start++;
13         end--;
14     }
15     return true;
16 }
17
18 int main() {
19     std::string input;
20     std::cout << "Enter a string: ";
21     std::getline(std::cin, input);
22     if (isPalindrome(input))
23         std::cout << "The string is a palindrome." << std::endl;
24     else
25         std::cout << "The string is not a palindrome." << std::endl;
26     return 0;
27 }
28
```

```
/tmp/I7LDqQ6QVf.o
Enter a string: abba
The string is a palindrome
```