

1. Write a program that has a float function `triangle()` that computes the area of a triangle and rectangle using its two formal parameters `h` and `w`, where `h` is the height and `w` is the length of the bases of the triangle.

```
#include <iostream>

float triangle(float h, float w)
{
    return h * w / 2;
}

int main()
{
    std::cout << "Please enter the height and the length of bases of triangle: ";
    float h, w;
    std::cin >> h >> w;
    if(!std::cin || h < 0 || w < 0)
    {
        std::cout << "Bad input\n";
        return 1;
    }
    std::cout << "The area of triangle is " << triangle(h, w) << "\n";
    return 0;
}
```

2. Write a program that accepts a positive integer from the user and displays the factorial of the given number. You should use a recursive function called `factorial()` to calculate the factorial of the number.

```
#include<iostream>

using namespace std;
```

```

int factorial(int n);

int main() {
    int n;
    cout << "Enter a positive integer: ";
    cin >> n;
    cout << "Factorial of " << n << " = " << factorial(n);
    return 0;
}

int factorial(int n) {
    if(n > 1)
        return n * factorial(n - 1);
    else
        return 1;
}

```

3. Write a function called isPrime() that accepts a number and determine whether the number is prime or

```

#include <iostream>
using namespace std;
bool check_prime(int);

int main() {
    int n;
    cout << "Enter a positive integer: ";
    cin >> n;

    if (check_prime(n))

```

```
    cout << n << " is a prime number.";
else
    cout << n << " is not a prime number.";
return 0;
}

bool check_prime(int n) {
    bool is_prime = true;
    // 0 and 1 are not prime numbers
    if (n == 0 || n == 1) {
        is_prime = false;
    }
    for (int i = 2; i <= n / 2; ++i) {
        if (n % i == 0) {
            is_prime = false;
            break;
        }
    }
    return is_prime;
}
```