Day 74/180 4 problems in One shot

1: Sum of cubes of N natural numbers using Recursion.

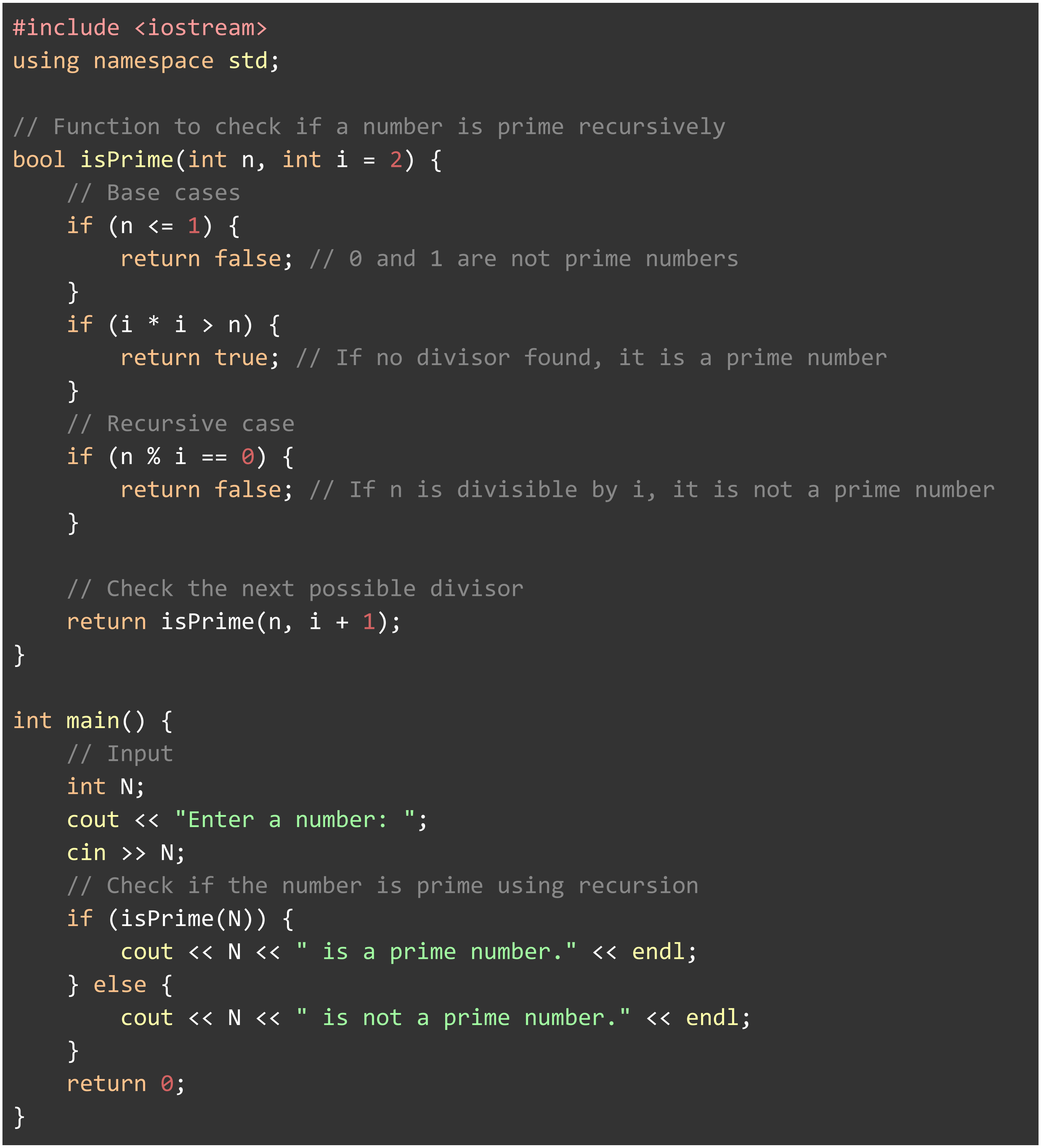
```
#include <iostream>
using namespace std;

// Function to calculate the sum of cubes of N natural numbers recursively
int sumOfCubes(int n) {
    if (n == 0) {
        return 0; // Base case: stop recursion when n becomes 0
    } else {
        return n * n * n + sumOfCubes(n - 1); // Recursive call with
decremented n
    }
}

int main() {
    int N;
    cout << "Enter the value of N: ";
    cin >> N;

    // Calculate and print the sum of cubes using recursion
    int result = sumOfCubes(N);
    cout << "Sum of cubes of first " << N << " natural numbers: " << result
<< endl;

    return 0;
}
```

2: Power of 4

Solution:

```cpp
#include <iostream>
using namespace std;

// Function to check if a number is a power of four using recursion
bool isPowerOfFour(int n) {
    // Base case 1: If n is 1, it is a power of four
    if (n == 1) {
        return true;
    }
    // Base case 2: If n is non-positive or not divisible by 4, it is not a power of four
    if (n <= 0 || n % 4 != 0) {
        return false;
    }
    // Recursive case: Check if the result of dividing n by 4 is a power of four
    return isPowerOfFour(n / 4);
}

int main() {
    // Example usage
    int num;
    cout << "Enter a number: ";
    cin >> num;

    // Check if the entered number is a power of four and print the result
    if (isPowerOfFour(num)) {
        cout << num << " is a power of four." << endl;
    } else {
        cout << num << " is not a power of four." << endl;
    }

    return 0;
}
```

3: Given a Number N, check whether it is prime or not using Recursion.



```cpp
#include <iostream>
using namespace std;

// Function to check if a number is prime recursively
bool isPrime(int n, int i = 2) {
    // Base cases
    if (n <= 1) {
        return false; // 0 and 1 are not prime numbers
    }
    if (i * i > n) {
        return true; // If no divisor found, it is a prime number
    }
    // Recursive case
    if (n % i == 0) {
        return false; // If n is divisible by i, it is not a prime number
    }

    // Check the next possible divisor
    return isPrime(n, i + 1);
}

int main() {
    // Input
    int N;
    cout << "Enter a number: ";
    cin >> N;
    // Check if the number is prime using recursion
    if (isPrime(N)) {
        cout << N << " is a prime number." << endl;
    } else {
        cout << N << " is not a prime number." << endl;
    }
    return 0;
}
```