소수 찾기

Level 2

numberOfPrime 메소드는 정수 n을 매개변수로 입력받습니다.

1부터 입력받은 숫자 n 사이에 있는 소수의 개수를 반환하도록 numberOfPrime 메소드를 만들어 보세요.

소수는 1과 자기 자신으로만 나누어지는 수를 의미합니다.  
(1은 소수가 아닙니다.)

10을 입력받았다면, 1부터 10 사이의 소수는 [2,3,5,7] 4개가 존재하므로 4를 반환  
5를 입력받았다면, 1부터 5 사이의 소수는 [2,3,5] 3개가 존재하므로 3를 반환

#include<iostream>

using namespace std;

int numOfPrime(int n)

{

int answer = 0;

int count = 0;

for (int i = 2; i <= n; i++)

{

for (int j = 1; j <= i; j++)

{

if (i % j == 0) count++;

}

if (count == 2)

{

answer++;

}

count = 0;

}

return answer;

}

int main()

{

int testCase = 10;

int testAnswer = numOfPrime(testCase);

cout<<testAnswer;

}

/\*

과정 설명

소수는 무조건 2개의 값 1과 자기자신이므로 5가 소수인지 검사하기위해서 1 ~ 5까지 나누면서 나머지가 0인 경우를 찾아서 count를 누적하고 끝난 count의 수가 2면 소수로 판별한다.

\*/