

```

// CPP program for simpson's 1/3 rule
#include <iostream>
#include <math.h>
using namespace std;

// Function to calculate f(x)
float func(float x)
{
    return log(x);
}

// Function for approximate integral
float simpsons_(float ll, float ul, int n)
{
    // Calculating the value of h
    float h = (ul - ll) / n;

    // Array for storing value of x and f(x)
    float x[10], fx[10];

    // Calculating values of x and f(x)
    for (int i = 0; i <= n; i++) {
        x[i] = ll + i * h;
        fx[i] = func(x[i]);
    }

    // Calculating result
    float res = 0;
    for (int i = 0; i <= n; i++) {
        if (i == 0 || i == n)
            res += fx[i];
        else if (i % 2 != 0)
            res += 4 * fx[i];
        else
            res += 2 * fx[i];
    }
    res = res * (h / 3);
    return res;
}

// Driver program
int main()
{
    float lower_limit = 4; // Lower limit
    float upper_limit = 5.2; // Upper limit
    int n = 6; // Number of interval
    cout << simpsons_(lower_limit, upper_limit, n);
    return 0;
}

```

Output:

1.827847