Автономное образовательное учреждение высшего профессионального образования

«Нижегородский государственный университет

им. Н.И. Лобачевского»

Институт Информационных технологий, математики и механики

Отчёт по лабораторной работе

Вычисление интеграла методом трапеций

Выполнила:

студентка группы 381506-2

Лемяскина Е.

Проверил:

ассистент каф. МОСТ

Козинов Е.А.

Нижний Новгород

2018 г.

**Содержание**

[Содержание 2](#_Toc515062100)

[Введение 3](#_Toc515062101)

[Постановка задачи 4](#_Toc515062102)

[Метод решения 5](#_Toc515062103)

[Подтверждение корректности 5](#_Toc515062104)

[Результаты экспериментов 6](#_Toc515062105)

[Заключение 7](#_Toc515062106)

[Литература 8](#_Toc515062107)

# Введение

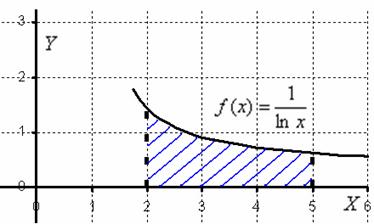
С увеличением вычислительной мощи компьютерных систем становится возможным производить довольно трудоемкие ранее невозможные расчеты. Появление такого подхода как параллельное программирование, позволяет использовать вычислительную мощь новейших компьютерных систем в полной мере. Становится возможным говорить о быстроте решения прикладных задач.

В математике интегралы используются повсеместно. Многие математические алгоритмы подразумевают в себе взятие интеграла. Однако, компьютер не может взять интеграл точно, потому требуются приближённые вычисления. Чем больше необходимо приближение, тем больше итераций потребуется – это довольно трудоемкая операция даже для современных компьютерных систем, поэтому проблема времени вычислений актуальна.

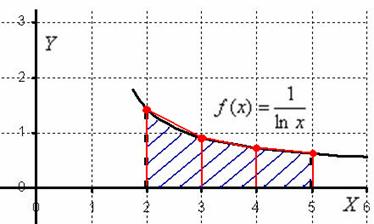
В данной лабораторной работе разработан алгоритм вычисления интеграла методом трапеций.

# Постановка задачи

Интеграл функции представляют, как площадь криволинейной трапеции, образованной границами интегрирования, осью Ox и самой функцией. При этом площадь криволинейной трапеции сосчитать можно лишь приблизительно, особенно для сложных функций.



Чтобы наиболее точно посчитать площадь, можно разбить отрезок интегрирования на несколько равных отрезков, график подынтегральной функции таким образом приближается ломаной кривой.



Формулу для вычисления площади трапеции можно представить, как:

Для успешного достижения цели необходимо решить следующие задачи:

1. Написать последовательную версию программы и параллельные версии с помощью OpenMP и TBB.
2. Оценить эффективность параллельных программ.

# Метод решения

Generator.exe создаёт 10 файлов с задачами, где задана сама функция, её левая и правая границы интегрирования. Также программа создаёт 10 файлов с ответами на эти задачи.

Solver.exe читает файл с задачей. Далее в последовательной версии сразу используется функция Integral. В качестве параметров ей передаются строка с функцией, левая и правая границы, а также количество итераций. В самой функции реализована формула метода трапеций.

В OpenMP версии распараллеливание происходит по количеству итераций. Оно делится на все потоки, и каждый поток выполняет функцию Integral только для своего участка отрезка и с меньшим количеством итераций.

В TBB версии распараллеливание происходит по участкам отрезка, таким же образом, как и в OpenMP версии.

В конце работы программы она выводит время работы, количество процессов и ответ в файл с номером задачи для проверки корректности.

# Подтверждение корректности

Checker.exe находит файлы ответа, заданного пользователем, и ответа, полученного программой. Если ответы отличаются не более, чем на допустимую погрешность, результат считается верным.

# Результаты экспериментов

В таблице ниже приведена зависимость времени работы от количества потоков на разных количествах итераций для OpenMP.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Количество элементов: | Последовательная версия | 2 | 4 | 8 |
| 10000 | 8.13336 | 6.38351 | 5.06699 | 4.9284 |
| 50000 | 42.3188 | 27.1374 | 20.6544 | 18.9163 |
| 100000 | 83.5698 | 58.3457 | 44.4135 | 39.9853 |

В таблице ниже приведена зависимость времени работы от количества потоков на разных количествах итераций для TBB.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Количество элементов: | Последовательная версия | 2 | 4 | 8 |
| 10000 | 8.13336 | 8.40357 | 4.10768 | 3.181952 |
| 50000 | 42.3188 | 25.442 | 20.0385 | 19.2752 |
| 100000 | 83.5698 | 52.3529 | 44.1603 | 38.3788 |

# Заключение

Если использовать для вычислений параллельную реализацию данного алгоритма, можно ускорить время выполнения программы в несколько раз. Насколько можно видеть по результатам эксперимента, время выполнения программы на разном количестве процессов сокращается примерно в столько раз, во сколько отличается количество процессов. Однако для столь значительного ускорения необходимо, чтобы количество узлов было больше или равно числу процессов. Также можно заметить, что время работы OpenMP-версии и TBB-версии практически не отличается.

# Литература

* 1. http://www.hpcc.unn.ru/?dir=910
  2. http://ssd.sscc.ru/sites/default/files/content/attach/343/mult\_matrix2d\_v1.pdf