МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Тверской государственный технический университет»

Кафедра «Программного обеспечения»

**Лабораторная работа №3**

по дисциплине «Параллельные вычисления»

Тема: «Вычисление определенного интеграла»

Выполнил: студент группы

ПИН.РИС.21.06

Пономарев В. С.

Проверил:

Желтов С. А.

Тверь2024

Оглавление

[Цель работы 3](#_Toc184012482)

[Задачи 3](#_Toc184012483)

[Создание приложения 3](#_Toc184012484)

[Последовательный алгоритм 4](#_Toc184012485)

[Параллельный алгоритм 5](#_Toc184012486)

[Сравнение времени выполнения алгоритмов 6](#_Toc184012487)

[Вывод 7](#_Toc184012488)

[Источники 7](#_Toc184012489)

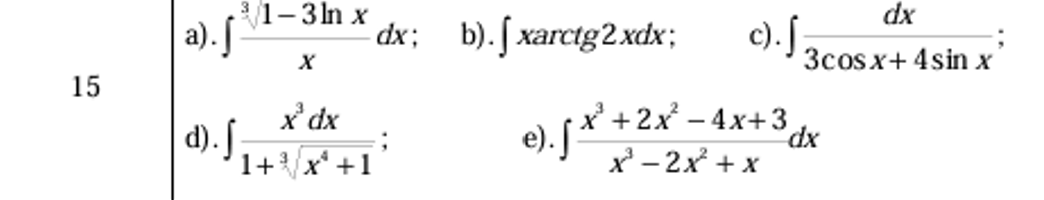
# Цель работы

Целью лабораторной работы является сравнительный анализ различных методов и способов вычисления определенного интеграла.

# Задачи

1. Реализовать последовательный алгоритм вычисления определенного интеграла с определением числа *N*– количества отрезков.
2. Определить время фактического выполнения последовательной программы.
3. Реализовать параллельный алгоритм вычисления определенного интеграла методом разбиения на *р* отрезков и независимого вычисления интеграл на соответствующем отрезке.
4. Определить время фактического выполнения параллельной программы.
5. Провести сравнительный анализ времени выполнения программ, реализующих разные способы вычисления определенного интеграла.

Вариант 15



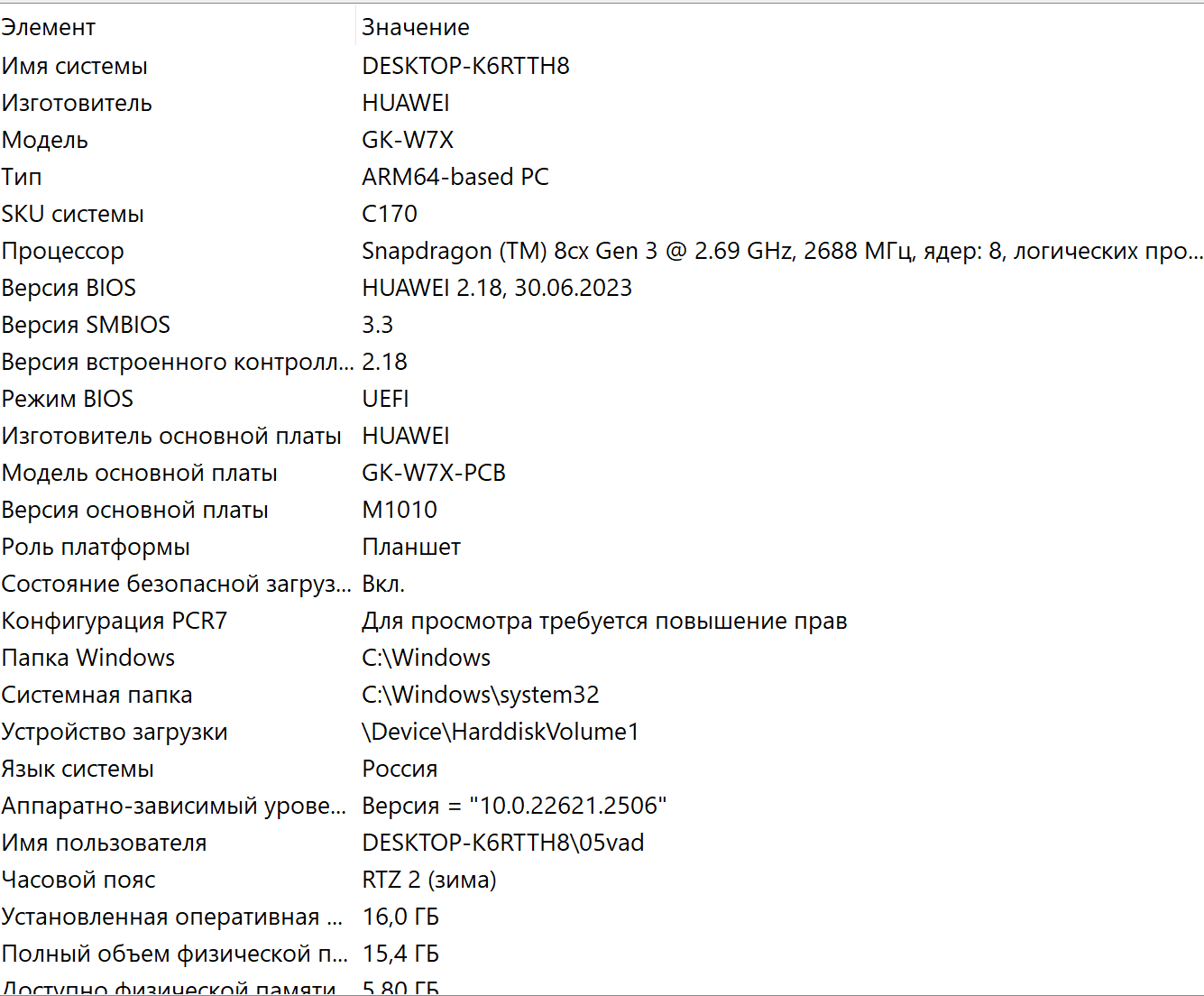
# Создание приложения

Для выполнения данной работы было разработано консольное приложение на языке программирования c++ в IDE Visual Studio Code. Основные характеристики системы на которой выполнялся код:

OC: Windows 11

CPU: Snapdragon 8cx Gen 3

RAM: 16Gb



# Последовательный алгоритм

Данный алгоритм заключается в последовательном сложении N площадей трапеций на интервале a, b для нахождения значения интеграла с заданной точностью precision.

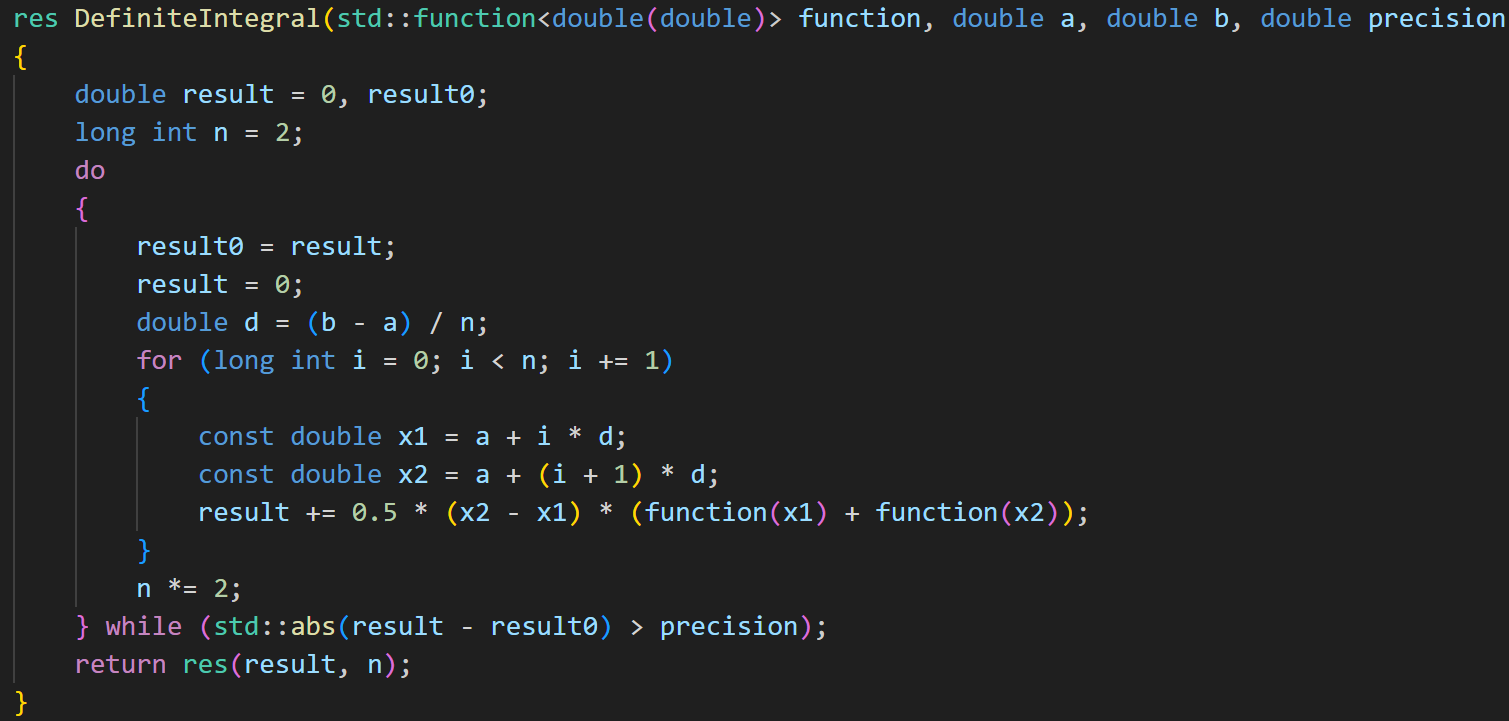


Рисунок 1 – код функции последовательного вычисления интеграла

# Параллельный алгоритм

Параллельный алгоритм предварительно делит интервал a, b на p интервалов и вычисляет параллельно значение каждого подотрезка и складывает их.

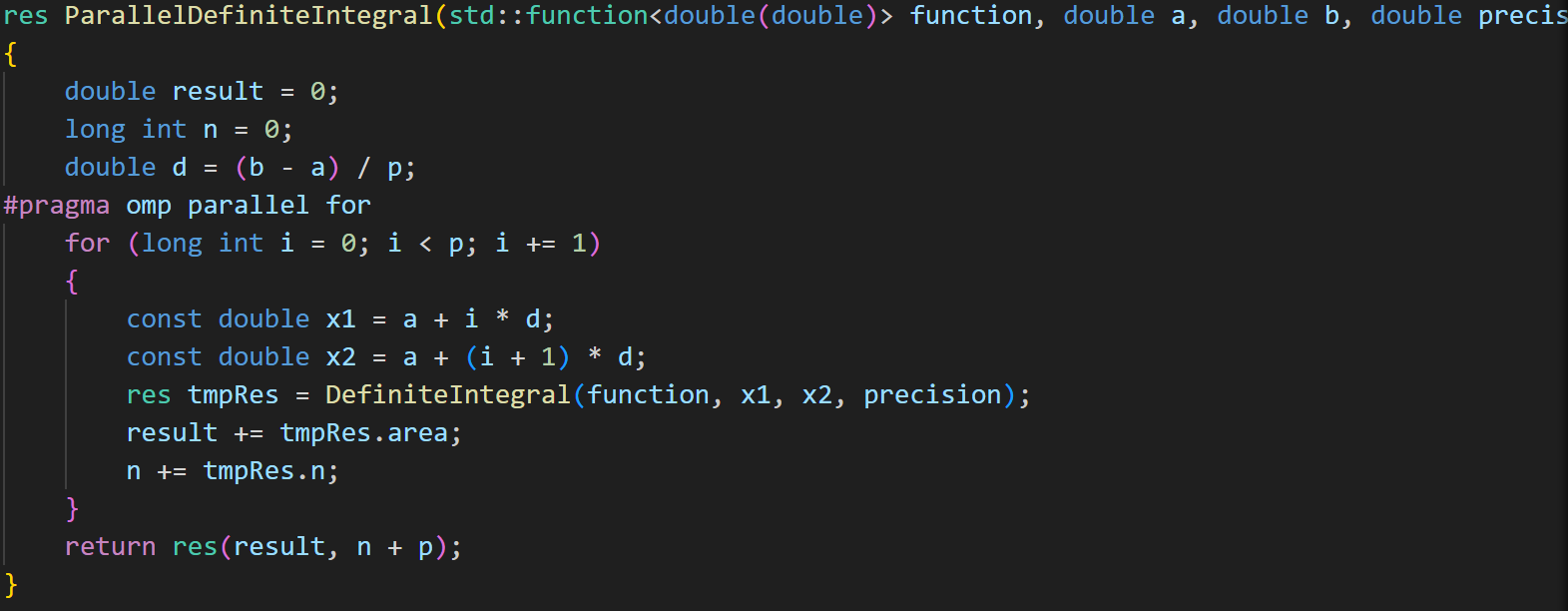


Рисунок 2 – код функции параллельного вычисления интеграла

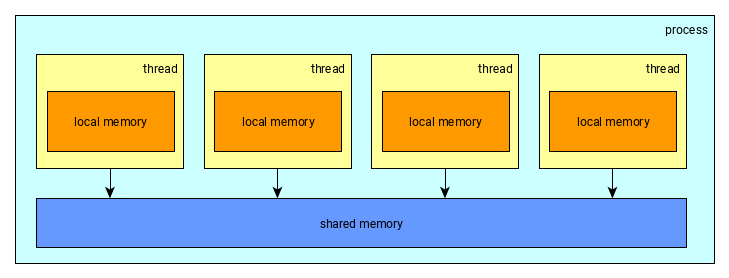
Распараллеливание происходит с помощью библиотеки omp.h, принцип распределения памяти между потоками указан на рисунке [3] ниже. 

Рисунок 3 – схема распределения памяти между потоками в библиотеке omp.h

# Сравнение времени выполнения алгоритмов

Рисунок 4 – График зависимости времени выполнения функции 1 от степени точности

Рисунок 5 – График зависимости времени выполнения функции 2 от степени точности

Рисунок 6 – График зависимости времени выполнения функции 3 от степени точности

Рисунок 7 – График зависимости времени выполнения функции 4 от степени точности

Рисунок 8 – График зависимости времени выполнения функции 5 от степени точности

# Вывод

Мною были реализованы алгоритмы нахождения значения интеграла – последовательный и параллельный. Выявлена зависимость, что чем больше отрезок и больше степень заданной точности, тем эффективнее работает параллельный алгоритм.

# Источники

1. Код программы на GitHub

URL: <https://github.com/vadimyt/Multi-processing>

1. Академия Microsoft: Параллельные вычисления и многопоточное программирование

URL: <https://intuit.ru/studies/courses/10554/1092/info>