Московский авиационный институт

(национально исследовательский университет)

Факультет информационных технологий и прикладной математики

Кафедра прикладной математики и информатики

Лаборатория №8 по курсу «Объектно-ориентируемое программирование»

Студент: Рождественских Аднрей Сергеевич

Преподаватель: А. В. Поповкин

Группа: М80-206Б

Вариант: 15

Дата:

Оценка:

Подпись:

Москва, 2018.

1 Цель работы

• Знакомство с параллельным программированием в C++.

2 Задача

Используя структуры данных, разработанные для лабораторной работы №6 (контейнер 1-ого уровня и классы-фигуры) разработать алгоритм быстрой сортировки для класс-контейнера.

Необходимо разработать два вида алгоритма:

1. Обычный, без параллельных вызовов.

2. С использованием параллельных вызовов. В этом случае, каждый рекурсивный вызов сортировки должен создаваться в отдельном потоке.

Для создания потоков ипользовать механизмы:

• future

• packaged task/async

Для обеспечения потокобезопасности структур использовать механизмы:

• mutex

• lock quard

Программа должна позволять:

• Вводить произвольное количество фигур и добавлять их в контейнер.

• Распечатывать содержимое контейнера.

• Удалять фигуры из контейнера.

• Проводить сортировку контейнера.

Фигуры: Прямоугольник, трапеция, ромб.

3 Описание

Параллельное программирование – это техника программирования, которая использует преимущества многоядерных или многопроцессорных компьютеров и является подмножеством более широкого понятия многопоточности (multithreading).

Параллельное программирование может быть сложным, но его легче понять, если считать его не “трудным“ , а просто “немного иным“. Оно включает в себя все черты более традиционного, последовательного программирования, но в параллельном программировании имеются три дополнительных, четко определенных этапа:

• Определение параллелизма: анализ задачи с целью выделить подзадачи, которые могут выполняться одновременно.

• Выявление параллелизма: изменение структуры задачи таким образом, чтобы можно было эффективно выполнять подзадачи. Для этого часто требуется найти зависимости между подзадачами и организовать исходный код так, чтобы ими можно было эффективно управлять.

• Выражение параллелизма: реализация параллельного алгоритма в исходном коде с помощью системы обозначений параллельного программирования.

4 Изменения

Изменился файл TbinaryTree.h TbinaryTree.cpp

5 Выводы

Благодаря этой лабораторной работе я получила опыт применения потоков в C++, что, несомненно, пригодится в дальнейшем. Распараллеливание вычислений - незаменимая вещь для уменьшения времени работы какого-то алгоритмом, например, сортировки работают в разы быстрее, чем в однопоточной реализации.