**ОТЧЁТ**

**ЛАБОРАТОРНАЯ РАБОТА №8**

**СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ  
«**Хэллоу, Америка с другого берега!**»**

*Выполнил студент 3 курса МОиАИС*

*Сагитов Александр Александрович*

В качестве результата выполнения данной лабораторной работы необходимо предоставить отчёт по следующим пунктам:   
  
*1) Как выбирался шаг по оси ? Как проверяется консистентность данных?*

*2) Сколько версий успел создать пишущий поток?*

*3) Сколько версий смог прочесть каждый из потоков?*

Также необходимо дать определение «*консистентности*».

Со следующей страницы будет дан подробный ответ на каждый пункт.

P.S.: исходный код программы в репозитории ([ссылка](https://github.com/sagitovich/LabWorkFifthSem/tree/main/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D1%8B%20%D1%80%D0%B5%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B3%D0%BE%20%D0%B2%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%B8/lab8))

--------------

**Пункт №1**

*Шаг выбран* ***0.1***, такой выбор обусловлен следующими причинами:

- шаг должен быть достаточно малым, чтобы обеспечить плотное заполнение последовательности;

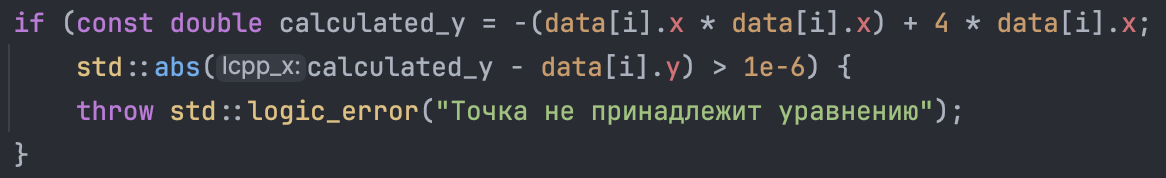
- при малом шаге увеличивается количество данных в стеке, что даёт возможность протестировать нагрузку на память и синхронизацию потоков;

Что такое консистентность? (ответ из презентации «Лекция 8. Lock-Free #2»)

**Консистентность данных** – это согласованность данных друг с другом, целостность данных, а также внутренняя непротиворечивость. Множество всех условий, налагаемых на данные, определяется моделью (структурой) данных.

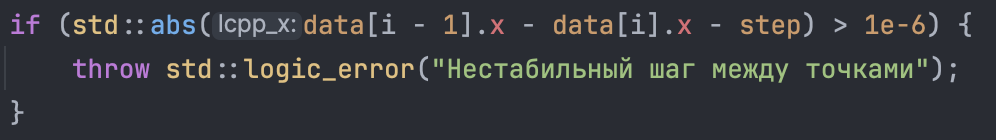
*Консистентность проверяется по двум условиям*:

- для каждой точки (x, y) проверяется, что значение ***y***, вычисленное по уравнению , совпадает с записанным ***y*** в пределах точности ;



файл: LockFreeStack.cpp

- проверяется, что каждая точка по оси отличается от предыдущей на шаг ***0.1****;*

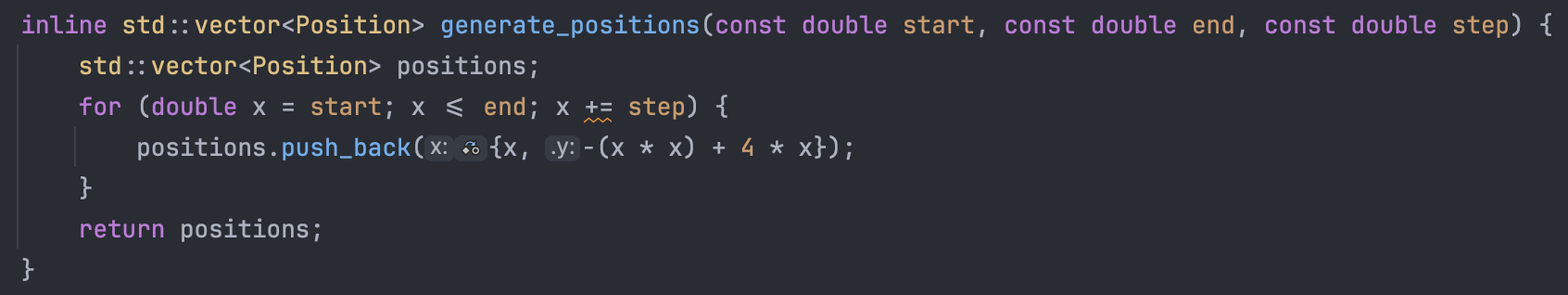


файл: LockFreeStack.cpp

--------------

**Пункт №2**

Пишущий поток добавлял данные в стек из последовательности точек, сгенерированной функцией:



файл: lockfree\_stack.hpp

***Количество созданных версий***:

Шаг = 0.1 , диапазон = [0, 4] => общее количество точек:

Таким образом, пишущий поток создал 41 версию.

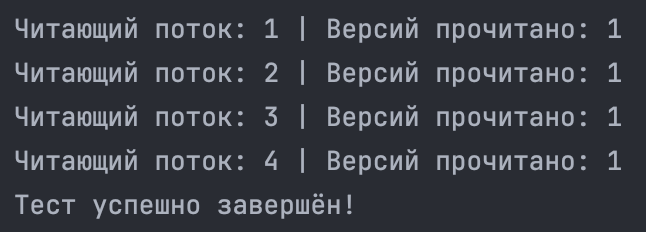
--------------

**Пункт №3**

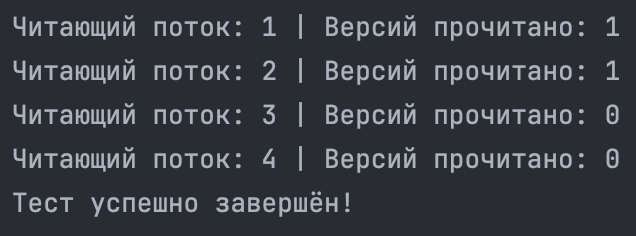
В эксперименте было задействовано 4 читающих потока.

Каждый поток считывал данные из Lock-Free стека параллельно.

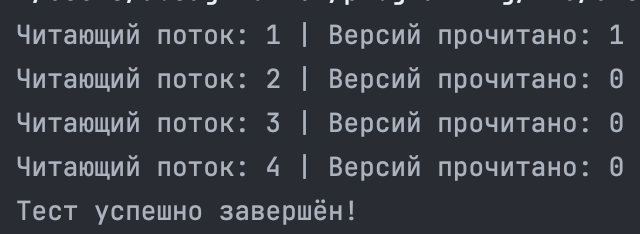
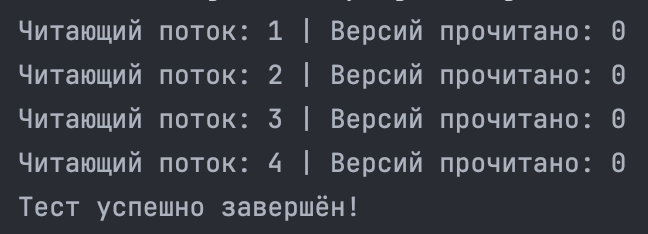
Каждому потоку необходимо было подписаться на стек, считать данные и проверить их на консистентность. Итоги по каждому потоку получились неоднозначными:



Запуск № 2



Запуск №



Запуск № 3

Запуск № 4

Каждый новый запуск потоки считывают разное количество версий. Связано это с тем, что потоки либо не успели подписаться на данные, либо стек завершил свою работу раньше, чем они смогли обработать. Это показывает, что синхронизации между потоками не стабильна.

Также не исключён факт того, что сама система / машина расходует разное количество ресурсов в разное время, что тоже влияет на результат выполнения программы.