# Клещенко Всеволод

# ЛР #4: [C++ & UNIX]: C++ PROCESSES / THREADS

## Цель

Познакомить студента с принципами параллельных вычислений. Составить несколько программ в простейшими вычислительными действиями, чтобы освоить принципы параллельных вычислений (когда одни алгоритмы зависят / не зависят от других).

## Задача

1. **[С++ SEQUENCE] Последовательные вычисления**

Требуется последовательно выполнить вычисления по формуле 1, вычисления по формуле 2, после чего выполнить вычисления по формуле 3, которые выглядят следующим образом: результат вычислений 1 + результат вычислений 2 – результат вычислений 1

Выполнить последовательно на 10 000 итераций и 100 000 итераций

Формула 1: f(x) = x^2-x^2+x\*4-x\*5+x+x

Формула 2: f(x) = x+x

Вывести длительность выполнения всех 10 000 итераций и 100 000 итераций в сек.

1. **[C++ THREADS] Параллельные вычисления через потоки**

Требуется параллельно (насколько возможно с помощью потоков) выполнить вычисления по формуле 1, вычисления по формуле 2, после чего выполнить вычисления по формуле 3, которые выглядят следующим образом: результат вычислений 1 + результат вычислений 2 – результат вычислений 1

Выполнить последовательно на 10 000 итераций и 100 000 итераций

Формула 1: f(x) = x^2-x^2+x\*4-x\*5+x+x

Формула 2: f(x) = x+x

Вывести длительность выполнения всех 10 000 итераций и 100 000 итераций в сек. в разбивке по шагам вычислений 1, 2 и 3

1. **[C++ PROCESS] Параллельные вычисления через процессы**

Требуется параллельно (насколько возможно с помощью процессов) выполнить вычисления по формуле 1, вычисления по формуле 2, после чего выполнить вычисления по формуле 3, которые выглядят следующим образом: результат вычислений 1 + результат вычислений 2 – результат вычислений 1

Выполнить последовательно на 10 000 итераций и 100 000 итераций

Формула 1: f(x) = x^2-x^2+x\*4-x\*5+x+x

Формула 2: f(x) = x+x

Вывести длительность выполнения всех 10 000 итераций и 100 000 итераций в сек. в разбивке по шагам вычислений 1, 2 и 3

1. **[LOG] Результат всех вышеперечисленных шагов сохранить в репозиторий (+ отчет по данной ЛР в папку doc)**

Фиксацию ревизий производить строго через ветку dev. С помощью скриптов накатить ревизии на stg и на prd. Скрипты разместить в корне репозитория. Также создать скрипты по возврату к виду текущей ревизии (даже если в папке имеются несохраненные изменения + новые файлы).

## Решение

1. После написания функций вычисления заданного количества итераций они вызываются последовательно с подсчетом времени выполнения:

Изображение выглядит как текст, монитор, снимок экрана, электроника

Автоматически созданное описание

1. При запуске параллельных поток std::threads функции выполняются параллельно в одной процессе, созданным программой:

Изображение выглядит как текст, монитор, снимок экрана, внутренний

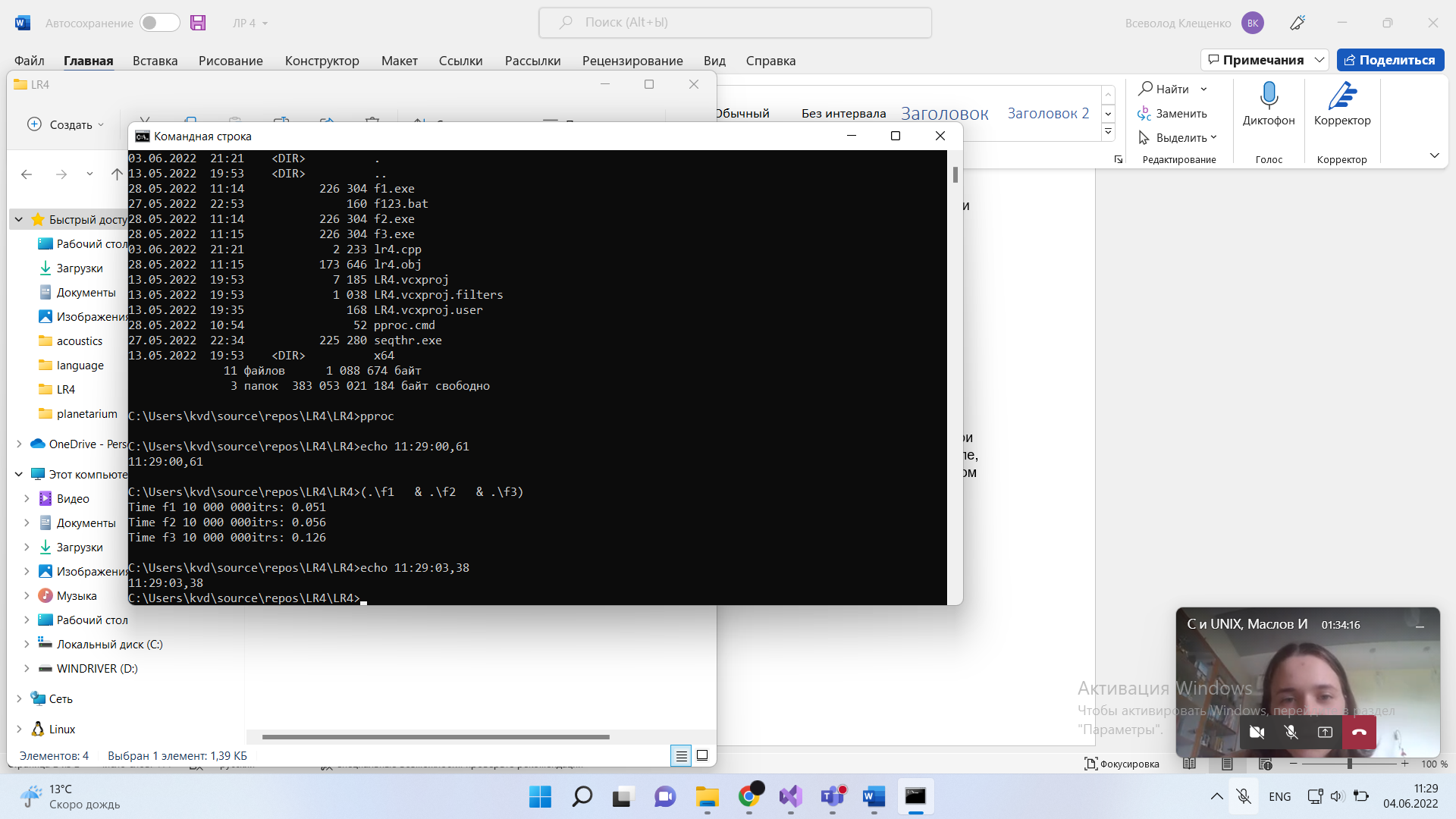
Автоматически созданное описание

Для сравнения время вычисления разного числа итераций последовательно и параллельно в потоках:

Изображение выглядит как текст, электроника, снимок экрана, внутренний

Автоматически созданное описание

1. Для создания параллельных процессов нужно запустить одновременно три программы, каждая из которых производила вычисления по своей формуле, для этого создавались три .exe файла и запускали в .cmd файле с отсчетом времени:



1. Было в прошлой лабораторной:

## Вывод

В результате лабораторной произведено ознакомление с принципами параллельных вычислений, составлено несколько программ с простейшими вычислительными действиями. В результате, можно сделать вывод о том, что при большом объеме вычисление параллельные вычисления сильно ускоряют вычисления, но при небольшом объеме могут сработать медленнее из – за тонкостей реализации.