

# Лабораторная работа №4

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

Выполнил: Макаренко Александр, группа Z33434, 3 курс, 2024 г.

### 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).

### Задачи

#### 1 [C++ 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

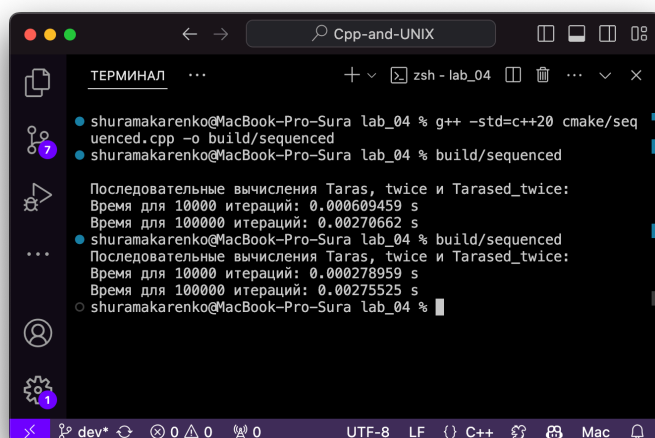
Формула 1:  $f(x) = x^2 - x^2 + x^4 - x^5 + x + x$

Формула 2:  $f(x) = x + x$

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

### Решение

Ожидаемая разница в 10 раз, но иногда бывают сильные скачки



```
shuramakarenko@MacBook-Pro-Sura lab_04 % g++ -std=c++20 cmake/sequenced.cpp -o build/sequenced
shuramakarenko@MacBook-Pro-Sura lab_04 % build/sequenced

Последовательные вычисления Taras, twice и Tarased_twice:
Время для 10000 итераций: 0.000609459 s
Время для 100000 итераций: 0.00270662 s
shuramakarenko@MacBook-Pro-Sura lab_04 % build/sequenced
Последовательные вычисления Taras, twice и Tarased_twice:
Время для 10000 итераций: 0.000278959 s
Время для 100000 итераций: 0.00275525 s
shuramakarenko@MacBook-Pro-Sura lab_04 %
```

## 2. [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

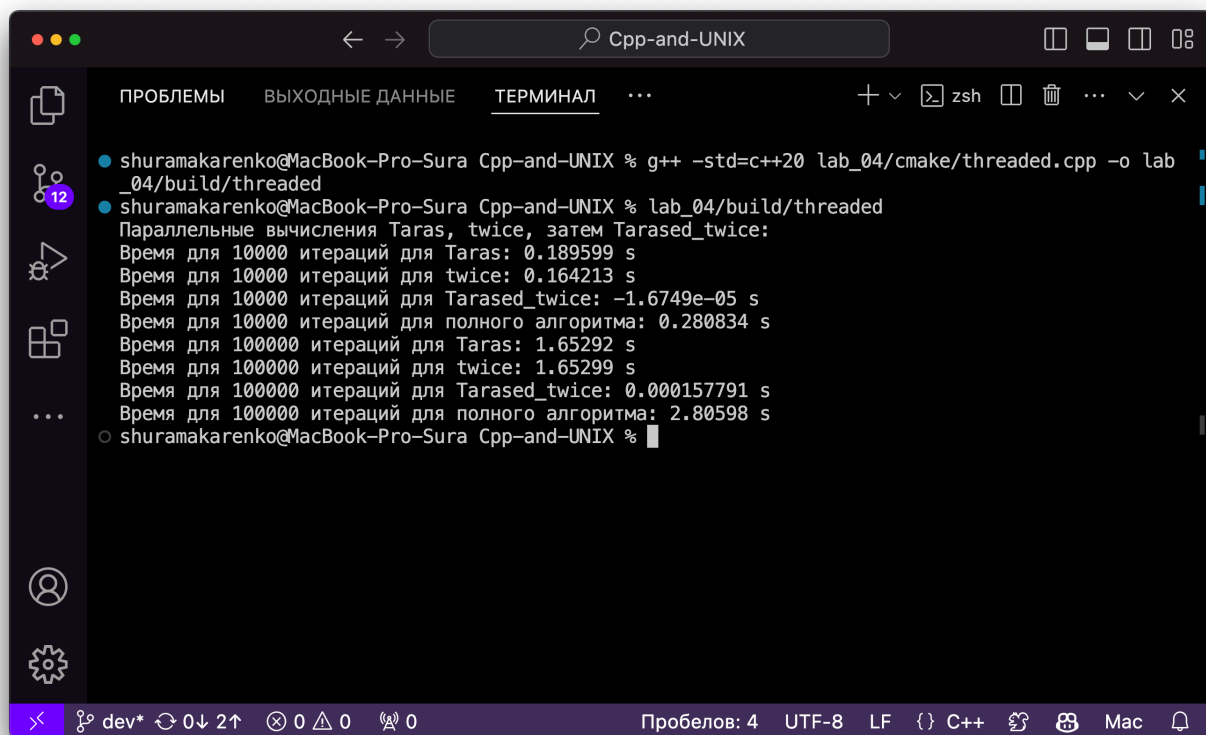
Формула 1:  $f(x) = x^2 - x^2 + x^4 - x^5 + x + x$

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

### Решение

Видно, что суммарное время меньше чем сумма времен



```
shuramakarenko@MacBook-Pro-Sura Cpp-and-UNIX % g++ -std=c++20 lab_04/cmake/threaded.cpp -o lab_04/build/threaded
shuramakarenko@MacBook-Pro-Sura Cpp-and-UNIX % lab_04/build/threaded
Параллельные вычисления Taras, twice, затем Tarased_twice:
Время для 10000 итераций для Taras: 0.189599 s
Время для 10000 итераций для twice: 0.164213 s
Время для 10000 итераций для Tarased_twice: -1.6749e-05 s
Время для 10000 итераций для полного алгоритма: 0.280834 s
Время для 100000 итераций для Taras: 1.65292 s
Время для 100000 итераций для twice: 1.65299 s
Время для 100000 итераций для Tarased_twice: 0.000157791 s
Время для 100000 итераций для полного алгоритма: 2.80598 s
shuramakarenko@MacBook-Pro-Sura Cpp-and-UNIX %
```

### 3. [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 1s

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

#### Решение

```
shuramakarenko@MacBook-Pro-Sura Cpp-and-UNIX % g++ -std=c++20 lab_04/cmake/p
rocessed.cpp -o lab_04/build/processed
shuramakarenko@MacBook-Pro-Sura Cpp-and-UNIX % lab_04/build/processed

Параллельные вычисления Taras, twice, затем Tarased_twice:
Время для 10000 итераций для Taras: 2.69912 s
Время для 10000 итераций для twice: 2.67566 s
Время для 10000 итераций для Tarased_twice: -1.5042e-05 s
Время для 10000 итераций для полного алгоритма: 3.77362 s
Время для 100000 итераций для Taras: 26.9789 s
Время для 100000 итераций для twice: 27.7593 s
Время для 100000 итераций для Tarased_twice: 0.000160875 s
Время для 100000 итераций для полного алгоритма: 38.1251 s
shuramakarenko@MacBook-Pro-Sura Cpp-and-UNIX %
```

**4. [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**

---

## Выводы

1.