МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

“Московский государственный университет геодезии и картографии”

(МИИГАИК)

Факультет геоинформатики и информационной безопасности

Кафедра геоинформационных систем и технологий

**Лабораторная работа №10**

**"Разработка Модуля Python на С++"**

Проверил: Выполнил:

Лебедев Евгений Денисович Студент группы: 2024-ФГИИБ-ПИ-1б

Центнер Валерий Викторович

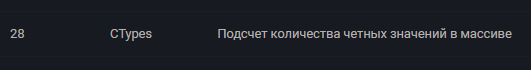
Москва 2024

1. Общая информация о задании:

Формулировка задания:

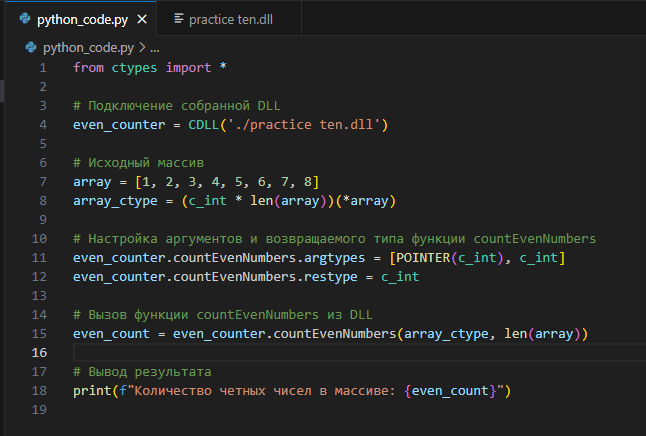
Данная лабораторная работа состоит в том, чтобы я создал модуль на С++ который выполняет такую функцию как “Подсчёт количества чётных значений в массиве”. После чего методом компиляции CTypes преобразовать это в модуль на Python.  
Также я должен провести замеры по скорости работы модулей и организовать это в график, после чего сделать вывод, а также не забыть приложить Листинги к проекту.

Номер моего варианта — это №28:



В задании мне был дан такой метод компиляции как CTypes:

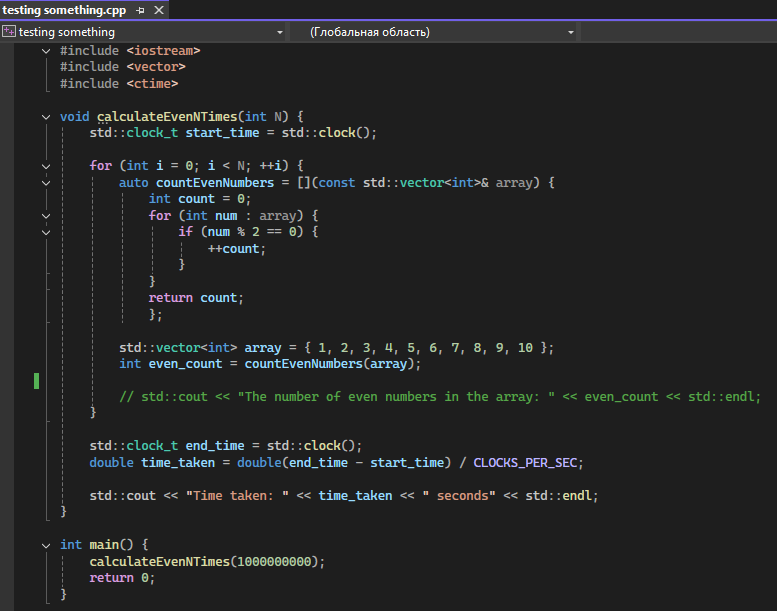
Доказательство использования CTypes в том, что интерфейс модуля CTypes позволяет использовать “питоновские” целые числа и строки при вызове C-функций.



Ссылка на GitHub:  
<https://github.com/guguker/inf_university>

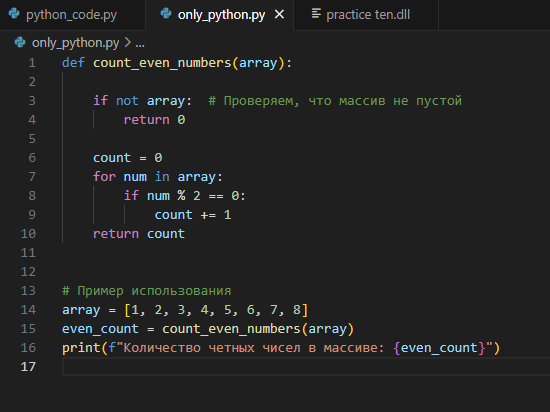
Зайдите пожалуйста в папку “laboratory work №10”, прямая ссылка с кучей лишних символов очень много места здесь занимает))

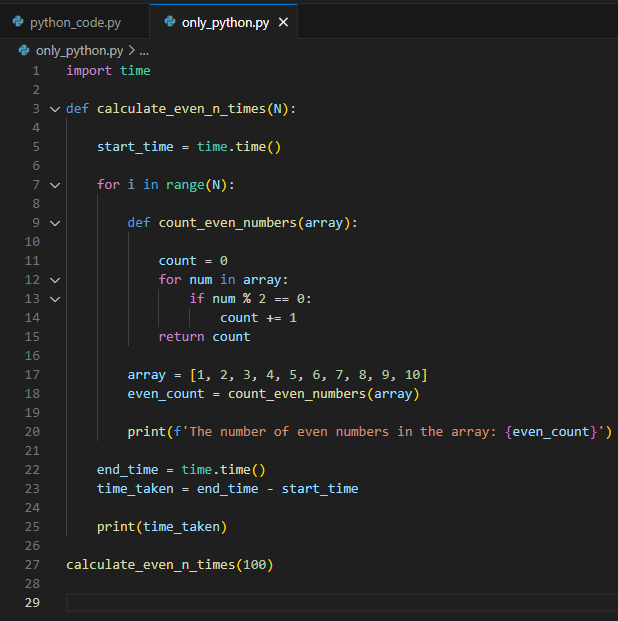
2. Реализация функции на чистом С++ и замер её скорости:



Результаты измерения приведены в пункте №5 в таблице.

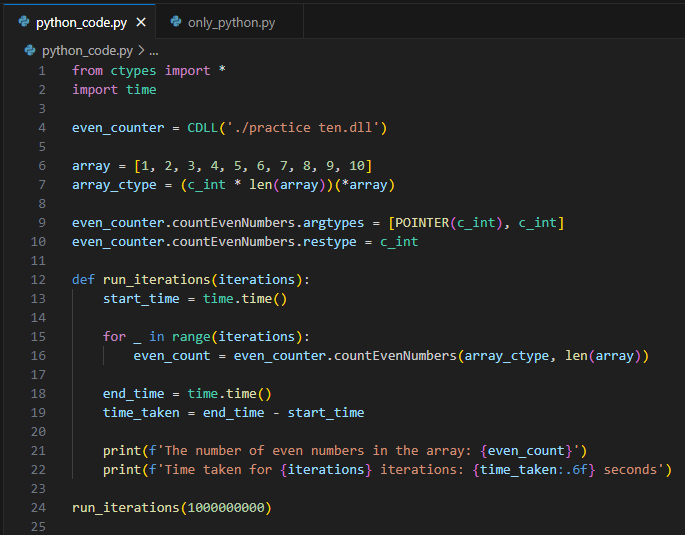
3. Реализация функции на чистом Python и замер её скорости:





Результаты измерения приведены в пункте №5 в таблице.

4. Реализация функции на модуле С++ в Python проекте и замер её скорости:

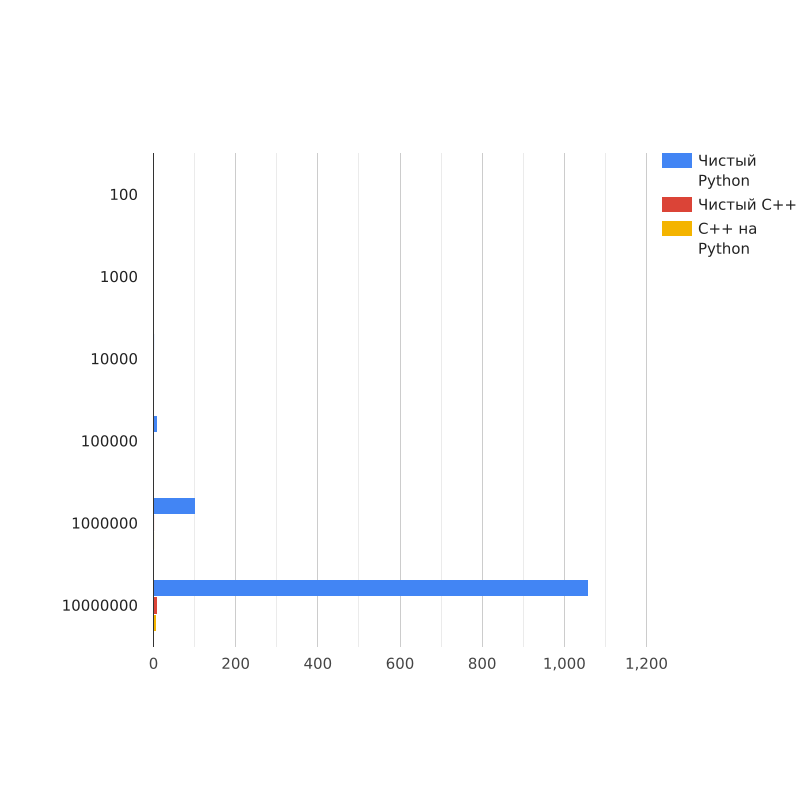


Результаты измерения приведены в пункте №5 в таблице.

5. Таблица замеров скорости работы:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Тест | Количество итераций | Чистый Python | Чистый С++ | C++ на Python |
| №1 | 100 | 0.01 секунды | 0.001 секунды | 0.000078 секунды |
| №2 | 1000 | 0.10 секунды | 0.003 секунды | 0.000709 секунды |
| №3 | 10 000 | 1.02 секунды | 0.01 секунды | 0.006 секунды |
| №4 | 100 000 | 10.15 секунды | 0.09 секунды | 0.07 секунды |
| №5 | 1 000 000 | 101.73 секунды  ~1.7 минуты | 0.95 секунды | 0.69 секунды |
| №6 | 10 000 000 | 1059.74 секунды  ~17.7 минуты | 8.54 секунды | 6.7 секунды |

6. График по полученным данным:



\*\*произошло небольшое унижение питона\*\*

7. Выводы о работе:

Исходя из данных таблицы и графика, можно точно сделать вывод что реализация функции через модуль С++ в Python проекте однозначно быстрее всех.

В цифрах:

(При 100 тысячах итераций):  
 На ~99.4% быстрее чем Чистый Python.  
 На ~23% быстрее чем Чистый C++.

(При 1 миллионе итераций):  
 На ~99.4% быстрее чем Чистый Python.  
 На ~28% быстрее чем Чистый C++.

(При 10 миллионах итераций):  
 На ~99.4% быстрее чем Чистый Python.  
 На ~22% быстрее чем Чистый C++.