

Санкт-Петербургский национальный
исследовательский институт информационных
технологий, механики и оптики

Физический факультет



Рабочий протокол и отчет по
лабораторной работе №1

[C++ & UNIX]: ОТЧЁТ ПО ЛАБОРАТОРНОЙ
РАБОТЕ №2

Группа: Z33431

Студент: Пуговкин Владимир

Преподаватель: Маслов И.Д.

1 Цель работы

Ознакомить студента с первыми программами с++, этапами при компиляции, типом данных в этом языке, а также его быстродействием по сравнению со смешанным языком программирования python.

- Содержание файла main.cpp

```
1
2     #include <iostream>
3     #include "header.hpp"
4     #include <chrono>
5     using namespace std;
6
7     int main ()
8     {
9         double x;
10        int n;
11        setlocale(LC_ALL, "ru_RU.UTF-8");
12        cout<<"Введите x:"<<endl;
13        cin>>x;
14        cout<<"Введите n:"<<endl;
15        cin>>n;
16        auto start = chrono::high_resolution_clock::now();
17        for(int i = 0; i < n; ++i)
18            calc(x);
19        auto end = chrono::high_resolution_clock::now();
20
21        auto duration = chrono::duration_cast<chrono::nanoseconds>(end - start);
22        cout << "Программа выполнялась за " << duration.count() << " наносекунд" << endl;
23
24        return 0;
25    }
```

- Содержание заголовочного файла header.hpp, использовался .hpp вместо .h, так как это более современная версия .h, обычно используют такое разрешение заголовка, если в проекте нет языка C, в целом файл .hpp может взаимодействовать с C, но нужно тогда сделать обертку с помощью extern "C"

```

1  #include <cmath>
2  ✓ double calc (double x)
3  {
4      return pow(x,2)-pow(x,2)+ pow(x,4)- pow(x,5)+ x + x;
5  }

```

- Содержимое main.py, здесь находится код, который аналогичен C++ main.cpp+header.hpp

```

1  import time
2
3  def calc(x):
4      return pow(x, 2) - pow(x, 2) + pow(x, 4) - pow(x, 5) + x + x
5
6  print("Введите x:")
7  x = float(input())
8
9  print("Введите n:")
10 n = int(input())
11
12 start = time.time()
13
14 for i in range(0, n):
15     calc(i)
16
17 end = time.time()
18
19 res_time = int((end - start) * 10**9)
20
21 print("Программа выполнялась за", res_time, " наносекунд")
]

```

2 Быстродействие C++ наглядно

C++ быстрее просто напорядок, чем python, давайте просто наглядно запустим код при x=10, n=100:

```

vlapugb@MSI:/usr/local/labitmo/lab_02/src$ python3 main.py
Введите x:
10
Введите n:
100
Программа выполнялась за 252485 наносекунд
vlapugb@MSI:/usr/local/labitmo/lab_02/src$ ./a.out
Введите x:
10
Введите n:
100
Программа выполнялась за 30700 наносекунд
]

```

3 Вывод

C++ быстрее за счет строгой типизации, то есть типизация фиксирована на этапе компиляции, а в питоне нет. Также компилятор C++ генерирует

.obj файлы с машинным кодом, который потом уже линкуются между собой, чтобы образовать полноценную программу. Машинный код напрямую работает с железом. В питоне же идет построчная интерпретация, что добавляет больше задержки. Также хочется отметить, что в C++ можно распоряжаться памятью как хочешь собственными руками, в питоне идет автоматическая оптимизация, которая иногда может привести к неэффективному использованию энергоресурсов.