Министерство образования и науки Российской Федерации Федеральное агентство по образованию Федеральное государственное бюджетное образовательное учреждение высшего образования «Вятский государственный университет»

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Лабораторная работа №9

по курсу «Разработка программных систем»

Выполнил студент группы ИВТ-31\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Птахова А.М/

Проверил доцент кафедры ЭВМ \_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Чистяков Г.А./

Киров 2023

1. Задание

В ходе выполнения лабораторной работы необходимо решить следующие задачи:

- По результатам лабораторной работы №1 выбрать один из наиболее требовательных методов;

- Реализовать данный метод на С/С++;

- Оформить реализацию в виде подключаемого модуля;

- Провести сравнения реализаций метода на Python и на С/С++ для набора подготовленных данных.

2. Листинг программы

**Header.h**

extern "C" \_\_declspec(dllexport) int pollard(int);

**Source.cpp**

#include <iostream>

int gcd(int a, int b)

{

if (a == 0)

{

return b;

}

else

{

return gcd(b % a, a);

}

}

int mulmod(long a, long b, long m)

{

return int((a \* b)) % m;

}

long powmod(long a, long n, long m)

{

long res = 1;

for (int i = 0; i < n; ++i) {

res = (res \* a) % m;

}

return res;

}

extern "C" \_\_declspec(dllexport)

int pollard(int n)

{

int b = 13;

int q[] = { 2, 3, 5, 7, 11, 13 };

int a = 5 % n;

for (int j = 0; j < 10; j++)

{

while (gcd(a, n) != 1)

{

a = mulmod(a, a, n);

a += 3;

a %= n;

}

for (int i = 0; i < sizeof(q); i++)

{

int qq = q[i];

int e = (int)floor(log((double)b) / log((double)qq));

int aa = (int)powmod(a, powmod(qq, e, n), n);

if (aa == 0) {

continue;

}

int g = gcd(aa - 1, n);

if (1 < g && g < n) {

return g;

}

}

}

return 1;

}

**Primerandom.py**

from typing import List

import ctypes

\_LAST\_PRIME\_NUMBER = 999999937

def is\_prime(number: int) -> bool:

lib = ctypes.CDLL('./pollard.dll')

return number == 2 or lib.pollard(number)==1

def get\_next(number: int) -> int:

lrec = lambda n: -1 if n == 10 \*\* 9 else n if is\_prime(n) else lrec(n + 1)

return lrec(number + 1)

def get\_random\_number() -> int:

from random import randint

n = randint(2, \_LAST\_PRIME\_NUMBER)

return n if is\_prime(n) else get\_next(n)

def get\_random\_array(length: int) -> List[int]:

return sorted(get\_random\_number() for \_ in range(length))

3. Сравнение

|  |  |  |
| --- | --- | --- |
| Размерность | С | Python |
| 200 | 0.01802349090576172 | 0.04303574562072754 |
| 500 | 0.03601503372192383 | 0.0980370044708252 |
| 1000 | 0.07503771781921387 | 0.20107197761535645 |
| 2000 | 0.14203619956970215 | 0.3950822353363037 |
| 5000 | 0.35108113288879395 | 0.9742448329925537 |

4. Вывод

В ходе выполнения лабораторной работы были изучены основы подключения модуля С/С++ к Python. Разработана библиотека dll, в которую вынесен самый требовательный метод – метод Полларда P-1 для проверки числа на простоту. Подключение к Python было выполнено при помощи библиотеки ctypes, с созданием экземпляра и вызовом метода через обращение.