Федеральное государственное автономное образовательное учреждение

высшего образования

«Национальный исследовательский университет «Высшая школа экономики»

**Московский институт электроники и математики им. А.Н. Тихонова**

Департамент прикладной математики

Специалитет

**О Т Ч Е Т**

**по работе**

*Программная реализация алгоритма проверки на простоту чисел Прота*

Выполнили студент гр.СКБ181

Андросов Павел Станиславович

(ФИО)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*(подпись)*

студент гр.СКБ181

Баглаев Артем Валерьевич

(ФИО)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*(подпись)*

студент гр. СКБ181

**МОСКВА 2022**

Оглавление

[Введение 3](#_Toc106133358)

[Теоретическая часть 3](#_Toc106133359)

[Число Прота 3](#_Toc106133360)

[Теорема Прота 3](#_Toc106133361)

[Алгоритм детерминированного доказательства простоты 3](#_Toc106133362)

[Алгоритм 1(вычисление ) 4](#_Toc106133363)

[Алгоритм 2(взятие квадратного корня по модулю N) 4](#_Toc106133364)

[Групповые операции 5](#_Toc106133365)

[Что такое [a] 5](#_Toc106133366)

[Реализация 6](#_Toc106133367)

[Алгоритм 1 6](#_Toc106133368)

[Алгоритм 2 7](#_Toc106133369)

[Групповая операция 8](#_Toc106133370)

[Общий алгоритм 9](#_Toc106133371)

[Время работы программы 10](#_Toc106133372)

[Проверка результатов (Wolfram) 12](#_Toc106133373)

[Список литературы 14](#_Toc106133374)

# Введение

Тема работы – реализация на языке Python алгоритма нахождения простых чисел Прота. Данный алгоритм реализует проверку простоты чисел, используя теорему Прота.

Теорема Прота – это детерминированный тест на простоту с полиномиальным временем. Иначе говоря, он даёт гарантированно точный ответ, простое ли исследуемое число или нет, и время его работы ограничено сверху многочленом от размера входа алгоритма. Несмотря на это, рассматриваемый тест не получил широкого практического применения вследствие сложности алгоритма его реализации и ввиду того, что он работает только с определённой группой чисел – числами Прота.

Ожидаемое время работы и время работы алгоритма в наихудшем случае равны  и  битовых операций соответственно.

## Теоретическая часть

### Число Прота

Число Прота — это положительное целое число вида



Где t – некоторое нечетное число и 

### Теорема Прота

Пусть N — число Прота

Если



Выполняется для некоторого a, тогда N – простое.

### Алгоритм детерминированного доказательства простоты

На вход подается число Прота N>3; Этот алгоритм возвращает PRIME, если N простое число. В противном случае возвращает COMPOSITE.

1. Пробуем найти  по Алгоритму 1. Если Алгоритм 1 возвращает, что N – составное, то остановить алгоритм и вернуть COMPOSITE.
2. Для каждого целого  пробуем посчитать  по Алгоритму 2. Если Алгоритм 2 вернул, что N – составное, то остановить алгоритм и вернуть COMPOSITE
3. Вернуть PRIME.

### Алгоритм 1(вычисление )

На вход подается , N не обязательно число Прота, для некоторых целых е>1 и нечетного t. Если N простое число, то алгоритм возвращает b такое что . В других случаях алгоритм возвращает b такое что  или выводит, что N составное.

1. Считаем  для 
2. Если  для всех, останавливаем алгоритм и пишем, что N составное.
3. Предположим, чтодля некоторых Посчитаем  для
4. Если  для всех  останавливаем алгоритм и пишем, что N составное.
5. Предположим, что  для некоторых . Возвращаем 

## Алгоритм 2(взятие квадратного корня по модулю N)

На вход подаются целые числа N, β и b такие, что  и , где  для некоторых целых е>1 и нечетного t. Если N простое число, то входящее β гарантировано является квадратичным вычетом по модулю N и этот алгоритм возвращает α такое, что  . Если N составное число, то алгоритм возвращает α такое, что  или останавливает алгоритм и пишет, что N составное.

1. Проверяем тривиальные случаи: если  для  , вернуть j.
2. Найти [a] такое, что  и по следующему алгоритму:
   1. Посчитать  для 
   2. Если для всех останавливаем алгоритм и пишем, что N составное.
   3. Предположим, что  для некоторых . Считаем для 
   4. Если для всех , останавливаем алгоритм и пишем, что N составное.
   5. Предположим, что  для некоторых .Считаем 
3. Считаем α:
   1. Считаем 
   2. Если  останавливаем алгоритм и пишем, что N составное. В противном случае возвращаем α.

## Групповые операции

На вход подаются  и  такие, что  и либо  или  для Если N простое число, то входящее β гарантировано является квадратичным вычетом по модулю N и этот алгоритм возвращает  такое, что  для . В противном случае алгоритм возвращает некоторое  или останавливает алгоритм и пишет, что N составное.

1. Если , вернуть 
2. Если , вернуть 
3. Если , вернуть ∞.
4. Если , останавливаем алгоритм и пишем, что N составное.
5. Считаем 
6. Если , останавливаем алгоритм и пишем, что N составное. В противном случае возвращаем a.

## Что такое [a]

Предположим, что  для некоторых целых α.

Группа  сконструирована так, что . То, какие элементы попадут в  - зависит от параметра α. После, мы берем 4 элемент из . Он должен быть отображен в .

Изображение выглядит как текст

Автоматически созданное описание

Оператор \*:

Изображение выглядит как текст

Автоматически созданное описание

## Реализация

### Алгоритм 1

Изображение выглядит как текст

Автоматически созданное описание

### Алгоритм 2

Изображение выглядит как стол

Автоматически созданное описание

### Групповая операция

Изображение выглядит как текст

Автоматически созданное описание

### Общий алгоритм

Изображение выглядит как текст

Автоматически созданное описание

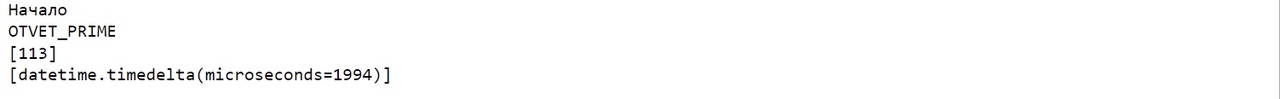
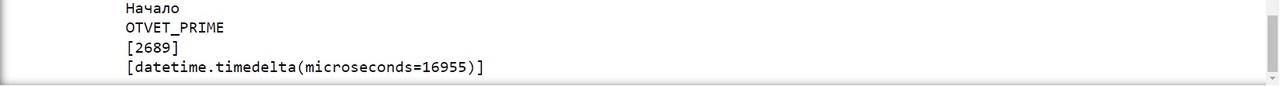
Изображение выглядит как текст

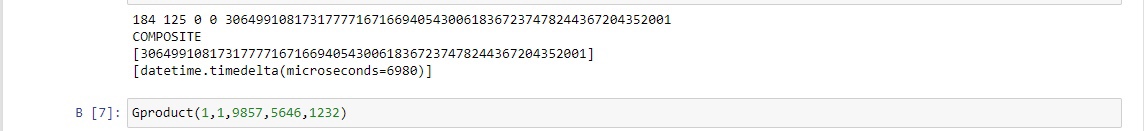
Автоматически созданное описание

### Время работы программы

Изображение выглядит как текст

Автоматически созданное описание





### Проверка результатов (Wolfram)

Изображение выглядит как текст

Автоматически созданное описание

## Список литературы

1. Tsz-Wo, S., 2018. Deterministic Primality Proving on Proth Numbers. Math.nt, szetszwo@cs.umd.edu Date Views 10.06.22 arxiv.org/abs/0812.2596v5.
2. Теорема Прота. Date Views 10.06.22 habr.com/ru/post/545208/.