Отчёт по лабораторной работе 6

Супонина Анастасия Павловна

Содержание

[Цель работы 1](#_Toc183294954)

[Задание 1](#_Toc183294955)

[Теоретическое введение 1](#_Toc183294956)

[Выполнение лабораторной работы 3](#_Toc183294957)

[Выводы 5](#_Toc183294958)

[Список литературы 5](#_Toc183294959)

Список иллюстраций

[Расширенный алгоритм Евклида 4](#_Toc183294960)

[Функция f 4](#_Toc183294961)

[Входные данные 4](#_Toc183294962)

[р-алгоритм Полланда 5](#_Toc183294963)

[Запуск функции и вывод результата 5](#_Toc183294964)

[Результат в консоли 6](#_Toc183294965)

Список таблиц

**Элементы списка иллюстраций не найдены.**

# Цель работы

Изучить $ p $-метод Полларда и научиться его программно реализовывать.

# Задание

***Программно реализовать на языке Julia $ p $-метод Полларда***

1. Реализовать алгоритм программно.
2. Разложить на множители данное преподавателем число.

# Теоретическое введение

#### Задача разложения составного числа на множители

Формулируется следующим образом: для данного положительного целого числа $ n $ найти его каноническое разложение:

где $ p\_i $ — попарно различные простые числа, $ a\_i $.

На практике не обязательно находить каноническое разложение числа $ n $ n = p q, p q < n. $$  
Далее будем понимать задачу разложения именно в этом смысле.

#### $ p $-Метод Полларда

Пусть $ n $ — нечетное составное число, $ S = {0, 1, …, n - 1} $, и $ f : S S $ — случайное отображение, обладающее сжимающими свойствами, например:

Основная идея метода состоит в следующем:

1. Выбираем случайный элемент $ x\_0 S $ и строим последовательность $ x\_0, x\_1, x\_2, $ x\_{i+1} = f(x\_i), $$ где $ i $, до тех пор, пока не найдем такие числа $ i, j $, что $ i < j $ и $ x\_i = x\_j $.
2. Поскольку множество $ S $ конечно, такие индексы $ i, j $ существуют (последовательность “зацикливается”). Последовательность $ {x\_i} $ будет состоять из “хвоста” $ x\_0, x\_1, …, x\_{l-1} $ длины $ O() $ и цикла $ x\_j, x\_{j+1}, …, x\_{j-1} $ той же длины.

#### Алгоритм, реализующий $ p $-метод Полларда:

Вход:  
Число $ n $, начальное значение $ c $, функция $ f $, обладающая сжимающими свойствами.

Выход:  
Нетривиальный делитель числа $ n $.

1. Положить $ a c $, $ b c $.
2. Вычислить:



1. Найти:



1. Если $ 1 < d < n $, то положить $ p d $ и вернуть $ p $.  
   Если $ d = n $, результат: “Делитель не найден”; при $ d = 1 $ вернуться на шаг 2.

#### Пример:

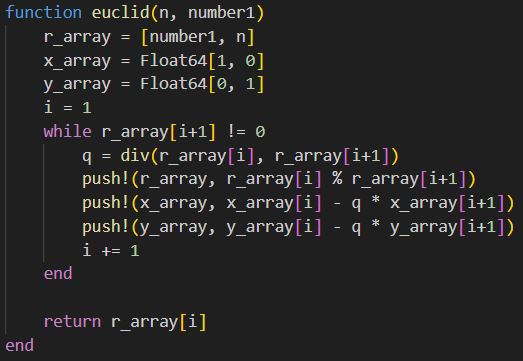
Найти $ p $-методом Полларда нетривиальный делитель числа $ n = 1359331 $.  
Положим $ c = 1 $ и $ f(x) = x^2 + 5 $. Работа алгоритма иллюстрируется следующей таблицей:

| $ i $ | $ a $ | $ b $ | $ d = (a - b, n) $ |
| --- | --- | --- | --- |
| 1 | 1 | 1 | 1 |
| 2 | 6 | 41 | 1 |
| 3 | 41 | 123939 | 1 |
| 4 | 1686 | 391594 | 1 |
| 5 | 123939 | 438157 | 1 |
| 6 | 435426 | 582738 | 1 |
| 7 | 391594 | 1144026 | 1 |
| 8 | 1090062 | 885749 | 1181 |

Результат:  
1181 является нетривиальным делителем числа $ 1359331 $.

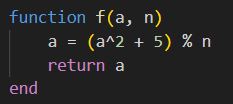
# Выполнение лабораторной работы

Для использования данного алгоритма необходимо находить НОД, для этого я взяла уже сделанную программу из прошлой лабораторной работы и записала её в виде отдельной функции. А именно, для нахождения НОД я использовала “Расширенный алгоритм Евклида”



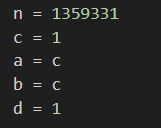
Расширенный алгоритм Евклида

При вычисслении значений мы считаем, что у нас есть функция, которая является отображением, я записала её в виде отдельной фкнуии для удобства, само случайное отображение я брала из условия



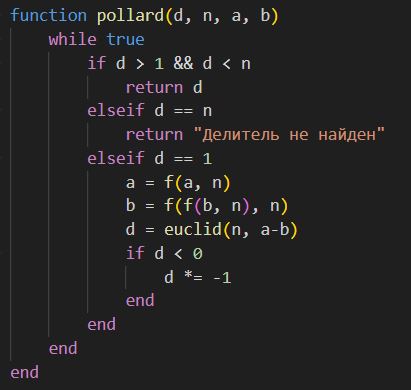
Функция f

Далее я прописала все парметры, которые используются для решения данной задачи



Входные данные

После этого я приступила к написанию функции, реализующей р-алгортм Полланда, используя аглоритм описанный в теоретической части, важно заметить, что там присутствует опечатка и когда мы рассчитываем значение параметра b, то примменяем отображение дважды. Также, так как значение b > a, то при нахождении НОД мы могли получиться отрицательное число, я добавила проверку на отрицательность и в таком случае поставила умножение d на -1



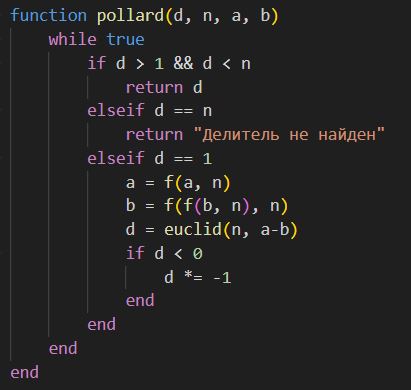
р-алгоритм Полланда

Закончив с написанием функций, я написала последние строки для запуска функции и записи результат её выполнения в переменную res, а после выод резальтата в терминал

Запуск функции и вывод результата

Запуск функции и вывод результата

Запустив данную программу в терминале я получила следующий вывод, на последним выводом сверзу можно видеть ппредыдущий с проверочными println, которые я использовала для того, чтобы быстро понимать, где расчеты идут неверно



Результат в консоли

# Выводы

В процессе выполнения работы, я реализовала разложение на множители для заданного числа, а именно реализовала р-алгоритм Полланда на языке программирования Julia.

# Список литературы

::: Пособие по лабораторной работе 5 {file:///C:/Users/bermu/Downloads/lab06.pdf}