Федеральное государственное образовательное бюджетное учреждение высшего образования

«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ

РОССИЙСКОЙ ФЕДЕРАЦИИ»

(Финансовый университет)

**Колледж информатики и программирования**

Отчёт о проделанной работе  
по практическому занятию № 1 по дисциплине

МДК.02.02 «Криптографические средства и методы защиты информации»

На тему: «Применение алгоритма Евклида для нахождения НОД. Решение линейных диофантовых уравнений»

Студент группы 3ОИБАС*-1221*

|  |  |
| --- | --- |
| Бажанова Д. И | «1» Октябрь 2023 г. |

Основная профессиональная образовательная программа по специальности

10.02.05 Обеспечение информационной безопасности автоматизированных систем

Форма обучения очная

Проверили: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Рой А.В.,

Оглавление

[Введение 3](#_Toc147083644)

[Цель работы 3](#_Toc147083645)

[Теоретическая часть 4](#_Toc147083646)

[Практическая часть 7](#_Toc147083647)

[Задание 1 7](#_Toc147083648)

[Задание 2 7](#_Toc147083649)

[Заключение 9](#_Toc147083650)

[Список литературы 10](#_Toc147083651)

# Введение

В мире математики существует множество задач и проблем, которые требуют нахождения наименьшего общего делителя (НОД) двух чисел или решения линейных диофантовых уравнений. Эти задачи имеют широкое применение в различных областях, включая криптографию, теорию чисел, и даже компьютерные науки. Одним из наиболее известных и эффективных методов для решения таких задач является алгоритм Евклида. В данном исследовании мы рассмотрим основные принципы и приложения алгоритма Евклида, а также рассмотрим способы решения линейных диофантовых уравнений с использованием этого алгоритма. В данной практической работе мы узнаем важность и универсальность методов нахождения НОД и решения линейных диофантовых уравнений в контексте современной математики и информационных технологий.

# Цель работы

1. Написать программу для нахождения НОД по алгоритму Евклида
2. Написать программу для нахождения корней линейного диофантового уравнения

# Теоретическая часть

Алгоритм Евклида — это классический математический метод, который широко применяется для нахождения наименьшего общего делителя (НОД) двух чисел, а также для решения линейных диофантовых уравнений. Этот алгоритм, названный в честь древнегреческого математика Евклида, имеет множество приложений в различных областях математики и инженерии. В данной статье мы рассмотрим теоретические основы алгоритма Евклида и его применение при нахождении НОД и решении линейных диофантовых уравнений.

*Нахождение НОД с помощью алгоритма Евклида*

Алгоритм Евклида предоставляет эффективный способ нахождения НОД для двух целых чисел. Начнем с описания самого алгоритма:

1. Пусть у нас есть два целых числа, назовем их a и b, где a >= b.
2. Делаем деление a на b и находим остаток r. Это выполняется следующим образом: a = bq + r, где q - результат целочисленного деления, а r - остаток.
3. Если r равен нулю, то b — это НОД чисел a и b, и мы завершаем выполнение алгоритма.
4. Если r не равен нулю, то мы заменяем a на b, а b на r и возвращаемся к шагу 2.

Процесс продолжается до тех пор, пока остаток r не станет равен нулю. На этом этапе b будет содержать значение НОД чисел a и b.

Алгоритм Евклида обладает несколькими важными свойствами:

1. **Корректность**: Алгоритм всегда находит НОД двух чисел a и b, даже если они отрицательны или равны нулю.
2. **Эффективность**: Алгоритм имеет линейную сложность, что означает, что количество итераций зависит от размера чисел a и b, но не превосходит линейное значение от их битовой длины.
3. **Удобство расширения**: Алгоритм Евклида может быть легко расширен для нахождения коэффициентов Безу и решения линейных диофантовых уравнений.

*Решение линейных диофантовых уравнений*

Линейное диофантово уравнение имеет следующий вид: ax + by = c, где a, b, и c - целые числа, и x, y - целые неизвестные.

Для решения линейных диофантовых уравнений, связанных с НОД, можно использовать расширенный алгоритм Евклида. Этот алгоритм находит НОД чисел a и b, а также находит целые коэффициенты x и y такие, что ax + by = НОД (a, b).

Линейные диофантовые уравнения имеют несколько важных свойств:

1. **Существование решений**: Если НОД (a, b) делит c (то есть c делится на НОД(a, b)), то линейное диофантово уравнение ax + by = c имеет хотя бы одно целочисленное решение. Это свойство известно как теорема о существовании решений линейных диофантовых уравнений.
2. **Бесконечное количество решений**: если ax + by = c имеет хотя бы одно целочисленное решение (x₀, y₀), то у этого уравнения бесконечно много решений. Все они могут быть получены путем добавления к x₀ и y₀ целых кратных b и a соответственно.
3. **Единственность решений (в случае выполнения определенных условий)**: В общем случае линейное диофантово уравнение может иметь бесконечно много решений, и они не всегда будут единственными. Однако, если НОД (a, b) равен 1 (то есть a и b взаимно просты), то решение (x₀, y₀) будет единственным. Это следует из мультипликативной инверсии в кольце вычетов.
4. **Нахождение общего решения**: Для нахождения всех решений линейного диофантова уравнения ax + by = c можно использовать расширенный алгоритм Евклида для нахождения НОД (a, b), а затем применить обратный ход этого алгоритма для нахождения целых коэффициентов x и y, удовлетворяющих уравнению.

# Практическая часть

## Задание 1

Напишите программу вычисления НОД. Проверьте работоспособность ещё на нескольких произвольных примерах. Поместите в отчёт исходный текст работающей версии программы и необходимые копии экрана или консоли.

class Evklid:  
 def \_\_init\_\_(self, a, b):  
 self.a = a  
 self.b = b  
  
 def algorotim(self):  
 while self.a != 0 and self.b != 0:  
 if self.a > self.b:  
 self.a %= self.b  
 else:  
 self.b %= self.a  
 return (self.a + self.b)  
  
  
result = Evklid(int(input()), int(input()))  
print(result.algorotim())

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

## Задание 2

С помощью алгоритма Евклида найдите частные корни линейного диофантова уравнения a\*x + b\*y = c, где a = (n+1)\*(n+2), b=(n+2)\*(n+3), c=(n+2)^3, n - номер по списку в журнале. Если Вы докажите, что решения для Вашего варианта не существует, то решите Диофантово уравнение 8х+16у=64. Подробно запишите процесс вычисления в редакторе электронных таблиц, поместив таблицу (снимок экрана) в отчёт.

def find\_num(n):  
 a = (n + 1) \* (n + 2)  
 b = (n + 2) \* (n + 3)  
 c = (n + 2) \*\* 3  
 diophantine\_equation(a, b, c)  
  
  
def evklid(a, b):  
 if b == 0:  
 return a, 1, 0  
 else:  
 d, x, y = evklid(b, a % b)  
 return d, y, x - (a // b) \* y  
  
  
def diophantine\_equation(a, b, c):  
 gcd, x0, y0 = evklid(a, b)  
  
 if c % gcd != 0:  
 print("[INFO] No solves")  
 else:  
 k = c // gcd  
 x = x0 \* k  
 y = y0 \* k  
  
 print(x, y)  
  
  
find\_num(int(input()))

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

# Заключение

В данной практической работе мы успешно освоили процесс написания программ, предназначенных для нахождения наименьшего общего делителя (НОД) с использованием алгоритма Евклида. Кроме того, мы приобрели навыки в решении линейных диофантовых уравнений, благодаря использованию расширенного алгоритма Евклида. Эти умения и знания оказались весьма полезными и могут быть применены в различных областях математики и информатики, а также в решении прикладных задач.

# Список литературы

1. [3.11.5 Documentation (python.org)](https://docs.python.org/3/index.html)
2. [Урок 50. линейные диофантовы уравнения - Алгебра - 7 класс - Российская электронная школа (resh.edu.ru)](https://resh.edu.ru/subject/lesson/7275/conspect/293629/)
3. [Онлайн-школа Фоксфорд (foxford.ru)](https://foxford.ru/wiki/matematika/algoritm-evklida?ysclid=ln7pyo9wsd594787934)
4. [Расширенный алгоритм Евклида - Алгоритмика (algorithmica.org)](https://ru.algorithmica.org/cs/modular/extended-euclid/?ysclid=ln7pz5q4i4376469880)