Отчёт по лабораторной работе №4

Алгоритм Евклида

Кодже Лемонго Арман

Содержание

[Цель работы 1](#_Toc180857228)

[Теоретические сведения 1](#_Toc180857229)

[Наибольший общий делитель 1](#_Toc180857230)

[Алгоритм Евклида 2](#_Toc180857231)

[Бинарный алгоритм Евклида 2](#_Toc180857232)

[Расширенный алгоритм Евклида 3](#_Toc180857233)

[Выполнение работы 3](#_Toc180857234)

[Реализация алгоритмов 3](#_Toc180857235)

[Контрольный пример 5](#_Toc180857236)

[Выводы 5](#_Toc180857237)

[Список литературы 5](#_Toc180857238)

Список иллюстраций

[Работа алгоритмов 5](#_Toc180857239)

# Цель работы

Изучение алгоритма Евклида нахождения Наибольший общий делитель и его вариаций.

# Теоретические сведения

## Наибольший общий делитель

Наибольший общий делитель (НОД) – это наибольшее целое число, на которое два или более целых числа можно поделить без остатка. Например, НОД чисел 12 и 18 равен 6, потому что 6 является наибольшим числом, которое делит оба этих числа на целое.

## Алгоритм Евклида

Алгоритм Евклида позволяет с легкостью вычислить наибольший общий делитель для двух положительных чисел. Формулировки и доказательство алгоритма Евклида мы привели в разделе « Наибольший общий делитель: определитель, примеры ».

Суть алгоритма заключается в том, чтобы последовательно проводить деление с остатком, в ходе которого получается ряд равенств вида:

Алгоритм Евклида

Вход. Целые числа . Выход. НОД.

* шаг 1. Положить , , .
* шаг 2. Найти остаток от деления на .
* шаг 3. Если , то положить . В противном случае положить и вернуться на шаг 2.
* шаг 4. Результат: .

Пример: Найти НОД для 64 и 48.

Конец: НОД (64,48) – это делитель 16.

## Бинарный алгоритм Евклида

Бинарный алгоритм Евклида вычисления НОД оказывается более быстрым при реализации этого алгоритма на компьютере, поскольку использует двоичное представление чисел а и b. Бинарный алгоритм Евклида основан на следующих свойствах наибольшего общего делителя (считаем, что 0 < b ≤ а):

* Вход. Целые числа .
* Выход. HOД.

1. Положить .
2. Пока оба числа и четные, выполнять до получения хотя бы одного нечетного значения или .
3. Положить .
4. Пока , выполнять следующие действия.
   * Пока четное, полагать .
   * Пока четное, полагать .
   * При положить . В противном случае положить .
5. Положить .
6. Результат:

## Расширенный алгоритм Евклида

Расширенный алгоритм Евклида находит наибольший общий делитель d чисел а и b и его линейное представление, т. е. целые числа x и у, для которых ах + by = d, и не требует «возврата», как в рассмотренном примере. Пусть d – НОД для a и b, т. е. d = (a, b), где a > b. Тогда существуют такие целые числа x и y, что d = ax +by. Иными словам, НОД двух чисел можно представить в виде линейной комбинации этих чисел с целыми коэффициентами

* Вход. Целые числа .
* Выход: НОД; такие целые числа , что .

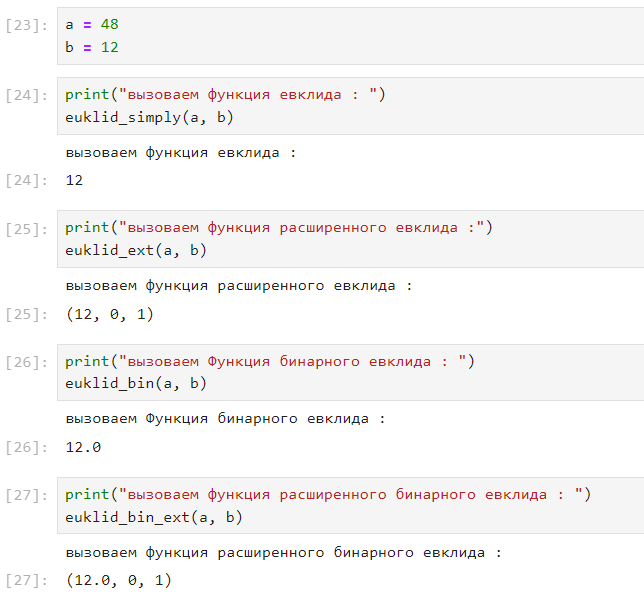
1. Положить
2. Разделить с остатком на :
3. Если , то положить , , . В противном случае положить , , и вернуться на шаг 2.
4. Результат: .

# Выполнение работы

## Реализация алгоритмов

def euklid\_simply(a, b):  
 while a!=0 and b!=0:  
 if a>=b:  
 a %= b  
 else:  
 b %=a  
 return a or b  
   
def euklid\_extended(a, b):  
 if a == 0:  
 return (b, 0, 1)  
 else:  
 div, x, y = euklid\_extended(b%a, a)  
 return (div, y-(b//a)\*x, x)  
   
def euklid\_binary(a, b):  
 g = 1  
 while(a%2 == 0 and b%2 == 0):  
 a = a/2  
 b = b/2  
 g = 2\*g  
 u,v = a,b  
 while u != 0:  
 if u%2 == 0:  
 u = u/2  
 if v%2 == 0:  
 v = v/2  
 if u>=v:  
 u = u-v  
 else:  
 v = v-u  
 d = g\*v  
 return d  
   
def euklid\_bin\_extended(a, b):  
 g = 1  
 while(a%2 == 0 and b%2 == 0):  
 a = a/2  
 b = b/2  
 g = 2\*g  
 u = a  
 v = b  
 A = 1  
 B = 0  
 C = 0  
 D = 1  
 while u!=0:  
 if u%2 == 0:  
 u = u/2  
 if A%2 == 0 and B%2 == 0:  
 A = A/2  
 B = B/2  
 else:  
 A = (A+b)/2  
 B = (B-a)/2  
 if v%2 == 0:  
 v = v/2  
 if C%2 == 0 and D%2 == 0:  
 C = C/2  
 D = D/2  
 else:  
 C = (C+b)/2  
 D = (D-a)/2  
 if u>=v:  
 u = u-v  
 A = A-C  
 B = B-D  
 else:  
 v = v-u  
 C = C-A  
 D = D-B  
 d = g\*v  
 x = C  
 y = D  
 return (d, x, y)

## Контрольный пример



Работа алгоритмов

# Выводы

Изучил алгоритм Евклида нахождения Наибольший общий делитель.

# Список литературы

1. [Нахождение НОД по алгоритму Евклида](https://zaochnik-com.com/spravochnikmatematika/delimost/nahozhdenie-nod/)
2. [Наибольший общий делитель](https://habr.com/ru/post/464949/)