Отчёт по лабораторной работе 4

МОЗИиИБ

Папикян Гагик Тигранович

Содержание

[1 Цель работы 1](#_Toc89536437)

[2 Задание 1](#_Toc89536438)

[3 Теоретическое введение 1](#_Toc89536439)

[3.1 Алгоритм Евклида 1](#_Toc89536440)

[3.2 Бинарный Алгоритм Евклида 2](#_Toc89536441)

[4 Выполнение лабораторной работы 2](#_Toc89536442)

[5 Выводы 4](#_Toc89536443)

# 1 Цель работы

Познакомиться с алгоритмами поиска Наибольшего Общего Делителя(НОД)

# 2 Задание

1. Реализовать алгоритм Евклида
2. Реализовать бинарный алгоритм Евклида
3. Реализовать расширенный бинарный алгоритм Евклида

# 3 Теоретическое введение

## 3.1 Алгоритм Евклида

Древнегреческие математики называли этот алгоритм ἀνθυφαίρεσις или ἀνταναίρεσις — «взаимное вычитание». Этот алгоритм не был открыт Евклидом, так как упоминание о нём имеется уже в Топике Аристотеля (IV век до н. э.). В «Началах» Евклида он описан дважды — в VII книге для нахождения наибольшего общего делителя двух натуральных чисел и в X книге для нахождения наибольшей общей меры двух однородных величин. В обоих случаях дано геометрическое описание алгоритма, для нахождения «общей меры» двух отрезков.

Историками математики было выдвинуто предположение, что именно с помощью алгоритма Евклида (процедуры последовательного взаимного вычитания) в древнегреческой математике впервые было открыто существование несоизмеримых величин (стороны и диагонали квадрата, или стороны и диагонали правильного пятиугольника). Впрочем, это предположение не имеет достаточных документальных подтверждений. Алгоритм для поиска наибольшего общего делителя двух натуральных чисел описан также в I книге древнекитайского трактата Математика в девяти книгах.

Суть алгоритма заключается в последовательном делении большего числа на меньшее. Если остаток от деления равен нулю, то делитель - это НОД Иначе за большее числа принимается делитель, а за меньшее - остаток, а алгоритм повторяется

## 3.2 Бинарный Алгоритм Евклида

Бинарный алгоритм Евклида использует смещения, а не деление, что дает прирост в производительности Так же существует расширенный бинарный алгоритм Евклида, дающий значения х и у, удовлетворяющие уравнению ах+by=d

# 4 Выполнение лабораторной работы

Был написан следующий скрипт на javascript

// Алгоритм Евклида   
const A = 106, B = 16  
console.log(`A=${A} B=${B}`)  
let a = A, b = B  
let rem  
while(rem != 0){  
 rem = a%b   
 a = b   
 b = rem  
}  
console.log(`d=${a}`)  
  
// Бинарный Алгоритм Евклида   
let g = 1  
a = A, b = B  
while(!a%2 && !b%2){  
 a /= 2  
 b /= 2  
 g \*= 2  
}  
let u = a, v = b   
while(u){  
 if(!u%2){  
 u/=2  
 }  
 if(!v%2){  
 v/=2  
 }  
 if(u>=v){  
 u -= v  
 }else{  
 v -= u  
 }  
}  
console.log(`d=${g\*v}`)  
  
// Расширенный Бинарный Алгоритм Евклида   
g = 1  
a = A, b = B  
while(!a%2 && !b%2){  
 a /= 2  
 b /= 2  
 g \*= 2  
}  
u = a, v = b   
let a\_ = 1, b\_ = 0,c\_ = 0, d\_= 1  
while(u){  
 if(!u%2){  
 u/=2  
 if(!a\_%2 && !b\_%2){  
 a\_ /= 2  
 b\_ /= 2  
 }else{  
 a\_ = (a\_+b) / 2  
 b\_ = (b\_-a) / 2  
 }  
 }  
 if(!v%2){  
 v/=2  
 if(!c\_%2 && !d\_%2){  
 c\_ /= 2  
 d\_ /= 2  
 }else{  
 c\_ = (c\_+b) / 2  
 d\_ = (d\_-a) / 2  
 }  
 }  
 if(u>=v){  
 u -= v  
 a\_ -= c\_  
 b\_ -= d\_  
 }else{  
 v -= u  
 c\_ -= a\_  
 d\_ -= b\_  
 }  
}  
console.log(`d=${g\*v} x=${c\_} y=${d\_}  
преверка(ax + by = d): ${A\*c\_}+${B\*d\_} = ${g\*v}   
`)

Результат исполнения скрипта приведен на рисунке 1 (рис. 1)



Figure 1: Выполнение лабораторной работы

# 5 Выводы

Был реализован алгоритм Евклида, бинарный и расширенный бинарный алгоритмы Евклида Для примера были использованы числа A = 106, B = 16, а на рис 4.1 видно, что их НОД = 2