Лабораторная работа №4

Вычисление наибольшего общего делителя

Кубасов В.Ю.

Содержание

# 1 Цель работы

Реализовать предложенные вариации алгоритма Евклида

# 2 Задание

Реализовать: - Алгоритм Евклида - Бинарный алгоритм Евклида - Расширенный алгоритм Евклида - Расширенный бинарный алгоритм Евклида

# 3 Теоретическое введение

Алгоритм Евклида в различных его вариациях - способ нахождения НОДа нескольких чисел. Его роль в криптографии определяется вычислением закрытых ключей в различных алогритмах шифрования. Так, например, алгоритм Евклида используется[1] в RSA для вычисления закрытого ключа.  
Данный алгоритм - способ вычисления НОДа за приемлемые[2] количество итераций и время.

# 4 Выполнение лабораторной работы

## 4.1 Алгоритм Евклида

По предложенному алгоритму был выполнен следующий код:

ri\_1 = a; ri = b; i = 1;  
  
while (true)   
 riplus1 = ri\_1 % ri;  
 if (riplus1 == 0)  
 break;  
 end;  
 ri\_1 = ri;  
 ri = riplus1;  
end;

где ri\_1 - ri - 1, ri - ri, riplus1 - ri + 1

## 4.2 Бинарный Евклида

По предложенному алгоритму был выполнен следующий код:

while ((a % 2 == 0) && (b % 2 == 0))  
 a /= 2;  
 b /= 2;  
 g \*= 2;  
end;  
  
u = a;  
v = b;  
  
while (u % 2 == 0)  
 u /= 2;  
end;  
  
while (v % 2 == 0)  
 v /= 2;  
end;  
  
while (u != 0)  
 if (u >= v)  
 u = u - v;  
 else  
 v = v - u;  
 end;  
end;

Обозначения соответствуют предыдущему пункту. Данный алгоритм устраняет излишние вычисления при четности чисел.

## 4.3 Расширенный алгоритм Евклида

По предложенному алгоритму был выполнен следующий код:

ri\_1 = a; ri = b; i = 1;  
xi\_1 = 1; xi = 0;  
yi\_1 = 0; yi = 1;  
  
while (true)   
 riplus1 = ri\_1 % ri;  
 q = (ri\_1 - riplus1) / ri;  
 xiplus1 = xi\_1 - q \* xi;  
 yiplus1 = yi\_1 - q \* yi;  
 if (riplus1 == 0)  
 break;  
 end;  
 ri\_1 = ri;  
 ri = riplus1;  
 xi\_1 = xi;  
 xi = xiplus1;  
 yi\_1 = yi;  
 yi = yiplus1;  
   
end;

Данный алгортим дополняет стандартный евклидовский разложением на множители.

## 4.4 Расширенный алгоритм Евклида

По предложенному алгоритму был выполнен следующий код:

while ((a % 2 == 0) && (b % 2 == 0))  
 a /= 2;  
 b /= 2;  
 g \*= 2;  
end;  
  
  
u = a; v = b;  
As = 1; Bs = 0; C = 0; D = 1;  
  
while (u % 2 == 0)  
 u /= 2;  
 if ((As % 2 == 0) && (Bs % 2 == 0))  
 As /= 2;  
 Bs /= 2;  
 else  
 As += b; As /= 2;  
 Bs -= a; Bs /= 2;  
 end;  
end;  
  
while (v % 2 == 0)  
 v /= 2;  
 if ((C % 2 == 0) && (D % 2 == 0))  
 C /= 2;  
 D /= 2;  
 else  
 C += b; C /= 2;  
 D -= a; D /= 2;  
 end;  
end;  
  
if (u >= v)   
 u =- v;  
 As -= C;  
 Bs -= D;  
else  
 v -= u;  
 C -= As;  
 D -= Bs;  
end;

Данный алгортим дополняет бинарный алгоритм евклида разложением на множители.

# 5 Выводы

Реализовали по предложенным алгоритмическим описаниям: - Алгоритм Евклида - Бинарный алгоритм Евклида - Расширенный алгоритм Евклида - Расширенный бинарный алгоритм Евклида

# Список литературы

1. Косс В., Потапчик А. Применение расширенного алгоритма Евклида в алгоритме шифрования RSA. БГТУ, 2023.

2. Абрамов С.А. Некоторые оценки, связанные с алгоритмом Евклида // Журнал вычислительной математики и математической физики. Российская академия наук, Отделение математических наук, 1979. Т. 19, № 3. С. 756–760.