

Вычисление наибольшего общего делителя

Гаджиев Нурсултан Тофик оглы ¹

2022 Moscow, Russia

¹RUDN University, Moscow, Russian Federation

Цель работы

Реализация алгоритмов вычисления наибольшего общего делителя (Евклида).

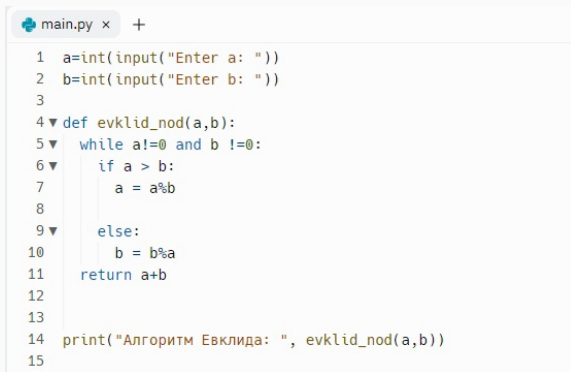
Задачи

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

Реализация

Реализация алгоритма Евклида

Функция `evklid_nod` для вычисления алгоритма Евклида. (рис. 1)



```
main.py x +
1 a=int(input("Enter a: "))
2 b=int(input("Enter b: "))
3
4 def evklid_nod(a,b):
5     while a!=0 and b !=0:
6         if a > b:
7             a = a%b
8
9         else:
10            b = b%a
11    return a+b
12
13
14 print("Алгоритм Евклида: ", evklid_nod(a,b))
15
```

Figure 1: Функция для вычисления алгоритма Евклида

Реализация бинарного алгоритма Евклида

Функция `evklid_binary` для вычисления бинарного алгоритма Евклида.
(рис. 2)

```
17 ▼ def evklid_binary(a,b):
18     g=1
19 ▼   while a%2 == 0 and b%2 == 0:
20       a=a/2
21       b=b/2
22       g=2*g
23   u=a
24   v=b
25 ▼   while u !=0:
26 ▼       if u % 2 ==0:
27           u= u/2
28 ▼       if v % 2 ==0:
29           v= v/2
30
31 ▼       if u>=v:
32           u=u-v
33
34 ▼       else:
35           v=v-u
36   d= g*v
37   return d
38
39 print("Бинарный алгоритм Евклида: ",evklid_binary(a,b))
```

Figure 2: Функция для вычисления бинарного алгоритма Евклида

Реализация расширенного алгоритма Евклида.

Функция `evklid_extend` для вычисления расширенного алгоритма Евклида. (рис. 3)

```
40
41 ▼ def evklid_extend(a,b):
42 ▼     if a== 0:
43         return(b,0,1)
44 ▼     else:
45         div,x,y= evklid_extend(b%a,a)
46         return(div,y-(b//a)*x,x)
47
48 print("Расширенный алгоритм Евклида: ",evklid_extend(a,b))
49
```

Figure 3: Функция для вычисления вычисления расширенного алгоритма Евклида.

Реализация расширенного бинарного алгоритма Евклида.

Функция `evklid_binary_extend` для вычисления расширенного бинарного алгоритма Евклида. (рис. 4) (рис. 5)

```
50 def evklid_binary_extend(a,b):
51     g=1
52     while a%2 == 0 and b%2 == 0:
53         a=a/2
54         b=b/2
55         g=2*g
56     u=a
57     v=b
58     A=1
59     B=0
60     C=0
61     D=1
62
63     while u !=0:
64         if u % 2 ==0:
65             u= u/2
66         if A%2==0 and B%2==0:
67             A=A/2
68             B=B/2
69         else:
70             A=(A+b)/2
71             B=(B-a)/2
72     if v % 2 ==0:
73         v= v/2
74     if C%2==0 and D%2==0:
75         C=C/2
76         D=D/2
```

Figure 4: Функция для вычисления расширенного бинарного алгоритма Евклида. Первая часть

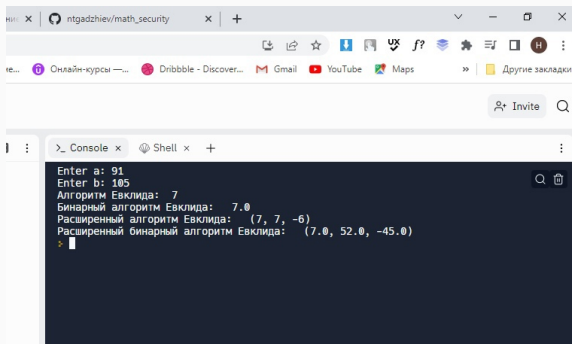
```

70     A=(A+b)/2
71     B=(B-a)/2
72 ▼   if v % 2 ==0:
73       v= v/2
74 ▼   if C%2==0 and D%2==0:
75       C=C/2
76       D=D/2
77 ▼   else:
78       C=(C+b)/2
79       D=(D-a)/2
80 ▼   if u>=v:
81       u=u-v
82       A=A-C
83       B=B-D
84
85 ▼   else:
86       v=v-u
87       C=C-A
88       D=D-B
89   d= g*v
90   x=C
91   y=D
92   return d,x,y
93
94 print("Расширенный бинарный алгоритм Евклида: ",evklid_binary_extend(a,b))

```

Figure 5: Функция для вычисления расширенного бинарного алгоритма Евклида. Вторая часть

Результат



The screenshot shows a web browser window with a terminal interface. The terminal displays the results of three algorithms applied to inputs a=91 and b=105. The results are as follows:

```
Enter a: 91
Enter b: 105
Алгоритм Евклида: 7
Бинарный алгоритм Евклида: 7.0
Расширенный алгоритм Евклида: (7, 7, -6)
Расширенный бинарный алгоритм Евклида: (7.0, 52.0, -45.0)
```

Figure 6: Результат алгоритмов

Реализовал алгоритм вычисления наибольшего общего делителя (Евклида).

Спасибо за внимание