

МИНИСТЕРСТВО НАУКИ ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
Институт цифрового развития  
Кафедра инфокоммуникаций.

**ОТЧЁТ**  
**по лабораторной работе №2**  
Дисциплина: «Алгоритмизация»

Выполнил:  
студент 2 курса группы ИВТ-б-о-22-1  
Бабенко Артём Тимофеевич  
Работа защищена с оценкой \_\_\_\_\_  
Подпись студента \_\_\_\_\_

Проверил:  
Воронкин Роман Александрович  
\_\_\_\_\_

Ставрополь, 2023

## Ход работы

### Задание

Нахождение чисел Фибоначчи

Сделать 2 способа реализации, не улучшенный и улучшенный и создать графики времени выполнения для них.

В конце сделать вывод роста времени.

```
1  #include <bits/stdc++.h>
2  using namespace std;
3  clock_t time_req;
4
5  int fib(int n)
6  {
7      if (n <= 1)
8          return n;
9      return fib(n - 1) + fib(n - 2);
10 }
11
12 int main()
13 {
14     int n = 40;
15     cout << fib(n);
16     getchar();
17     cout << "runtime = " << clock()/1000.0 << endl;
18     time_req = clock();
19     cout << (float)time_req/CLOCKS_PER_SEC << " seconds" << endl;
20     return 0;
21 }
```

Рисунок 1. Код программы

```
102334155
runtime = 1308.07
1.30812 seconds

...Program finished with exit code 0
Press ENTER to exit console.[]
```

Рисунок 2. вывод

Количество тиков, от N40 до N30

1308.07,  
809.43,  
398.034,  
214.438,  
165.929,  
134.99,  
41.494,  
39.761,  
24.314,  
15.951,  
9.721

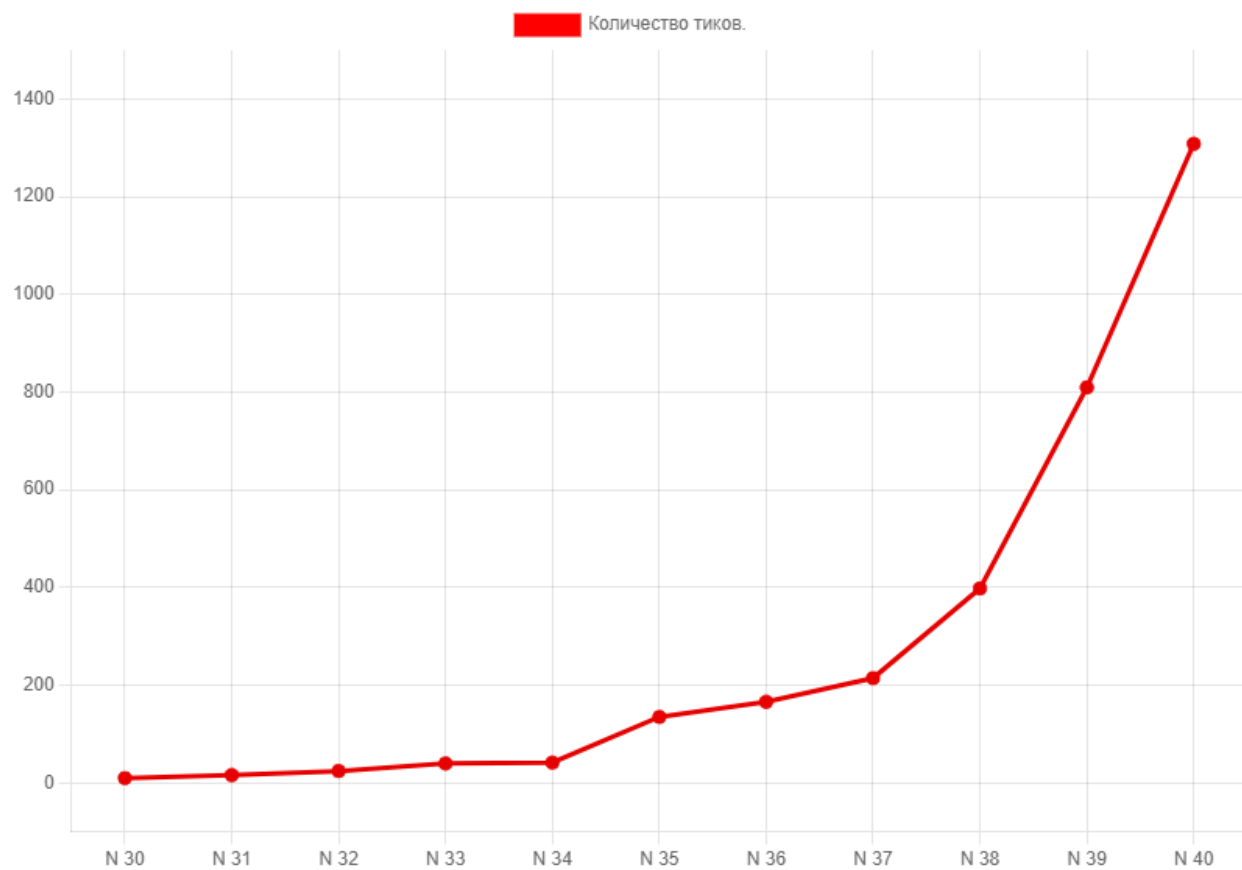


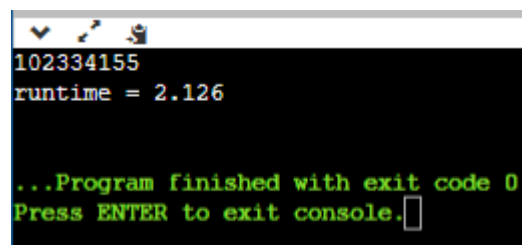
Рисунок 3. График зависимости времени (тиков) от номера числа Фибоначчи

```

1 // C++ program for Fibonacci Series
2 // using Dynamic Programming
3 #include<bits/stdc++.h>
4 using namespace std;
5
6 class GFG{
7
8 public:
9 int fib(int n)
10 {
11
12     // Declare an array to store
13     // Fibonacci numbers.
14     // 1 extra to handle
15     // case, n = 0
16     int f[n + 2];
17     int i;
18
19     // 0th and 1st number of the
20     // series are 0 and 1
21     f[0] = 0;
22     f[1] = 1;
23
24     for(i = 2; i <= n; i++)
25     {
26
27         //Add the previous 2 numbers
28         // in the series and store it
29         f[i] = f[i - 1] + f[i - 2];
30     }
31     return f[n];
32 }
33 };
34
35 // Driver code
36 int main ()
37 {
38     GFG g;
39     int n = 40;
40
41     cout << g.fib(n)<<endl;
42     cout << "runtime = " << clock()/1000.0 << endl; // время работы программы
43     return 0;
44 }

```

Рисунок 4. Код программы



```

102334155
runtime = 2.126

...Program finished with exit code 0
Press ENTER to exit console.

```

Рисунок 5. Вывод

Количество тиков, от N40 до N30

2.126,

1.749,  
1.068,  
1.655,  
1.337,  
0.999,  
2.089,  
2.218,  
1.399,  
1.714,  
1.949.

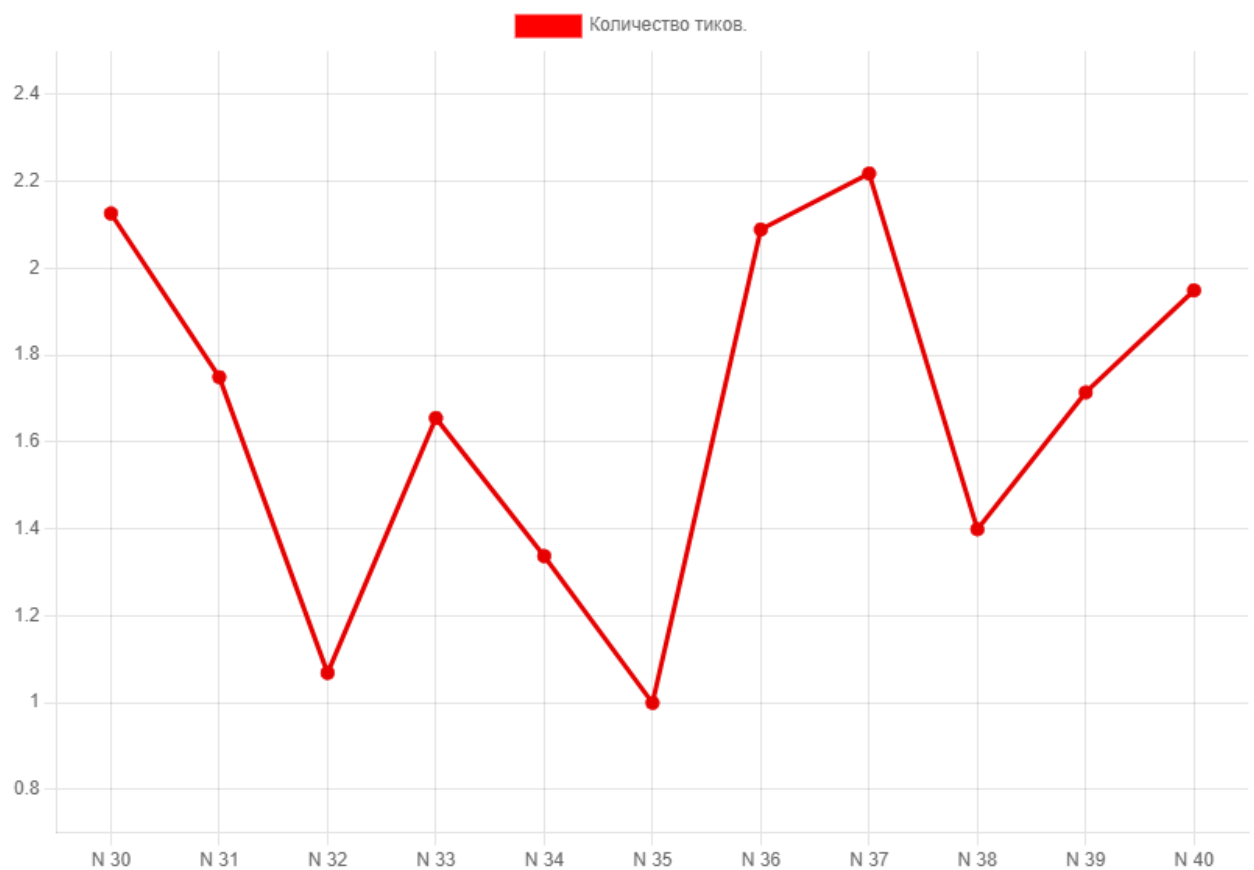


Рисунок 6. График зависимости времени (тиков) от номера числа Фибоначчи, улучшенный алгоритм

Вывод: В первом алгоритме можно заметить экспоненциальный рост времени выполнения алгоритма, во втором же рост времени выполнения не наблюдается.