Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития

Кафедра инфокоммуникаций

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОЙ РАБОТЕ №2**

**дисциплины «Алгоритмизация»**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Выполнил:  Лейс Алексей Вячеславович  2 курс, группа ИВТ-б-о-22-1,  09.03.01 «Информатика и вычислительная техника», направленность (профиль) «Программное обеспечение средств вычислительной техники и автоматизированных систем»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | | Руководитель практики кандидат тех. наук доцент кафедры инфокоммуникаций: Воронкин Р.А  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | |  | |

Отчет защищен с оценкой \_\_\_\_\_\_\_\_\_\_\_ Дата защиты\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ставрополь, 2022 г.

**Порядок выполнения работы:**

**№1 Наивная реализация Фибоначчи**

**Код:**

**using System;**

**using System.Diagnostics;**

**class HelloWorld**

**{**

**static void Main()**

**{**

**var sw = new Stopwatch();**

**sw.Start();**

**Console.WriteLine("Числа фибоначе сколько вывести: ");**

**ulong n = Convert.ToUInt64(Console.ReadLine());**

**if (n <= 1)**

**{**

**Console.WriteLine($"F[{n}] = {n}");**

**}**

**else**

**{**

**Console.WriteLine($"Fib [{n}] = {Fib(n)}");**

**}**

**sw.Stop();**

**Console.WriteLine($"Время выполнения: {sw.Elapsed}"); // Здесь логируем**

**Console.ReadLine();**

**}**

**static ulong Fib(ulong n)**

**{**

**if (n <= 1)**

**{**

**return n;**

**}**

**else**

**{**

**return Fib(n - 1) + Fib(n - 2);**

**}**

**}**

**}**

**№2 Фибоначчи улучшенная**

**Код:**

**using System;**

**using System.Diagnostics;**

**class HelloWorld {**

**static void Main()**

**{**

**var sw = new Stopwatch();**

**sw.Start();**

**Console.WriteLine("Размерность массива и сколько чисел фибоначи вывести: ");**

**int n = Convert.ToInt32(Console.ReadLine());**

**int [] arr = new int[n];**

**arr[0] = 0;**

**arr[1] = 1;**

**for(int i = 2 ; i < n; i++)**

**{**

**arr[i] = arr[i - 1 ] + arr[i - 2];**

**Console.WriteLine($"F[{i+1}] = {arr[i]}");**

**}**

**// Измеряемый код**

**sw.Stop();**

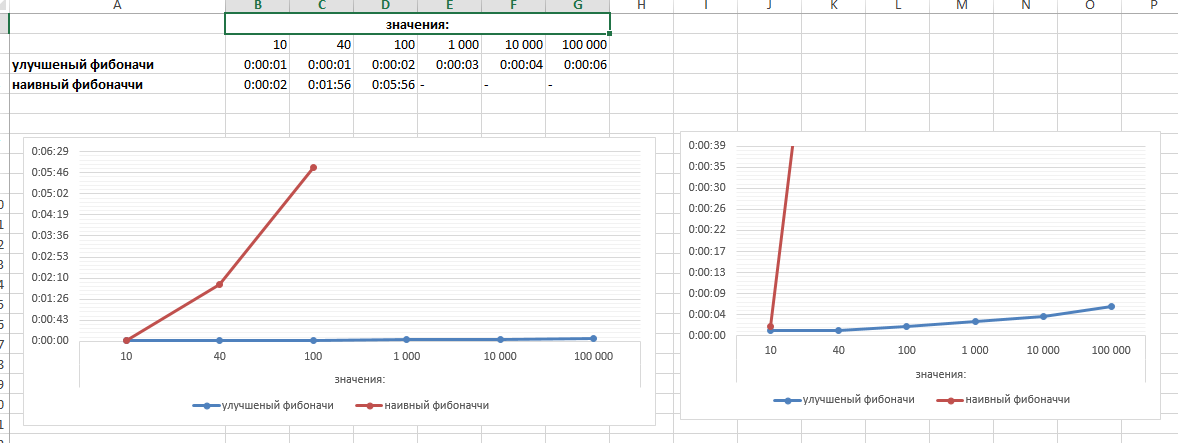
**Console.WriteLine($"Время выполнения: {sw.Elapsed}"); // Здесь логируем**

**Console.ReadLine();**

**}**

**}**

**График Фибоначчи:**



**№3 НОД наивный**

**Код:**

using System;

using System.Diagnostics;

class HelloWorld

{

static void Main()

{

var sw = new Stopwatch();

sw.Start();

Console.Write("Введите первое число: ");

int num1 = int.Parse(Console.ReadLine());

Console.Write("Введите второе число: ");

int num2 = int.Parse(Console.ReadLine());

int gcd = FindGCD(num1, num2);

Console.WriteLine($"Наибольший общий делитель чисел {num1} и {num2} равен {gcd}");

sw.Stop();

Console.WriteLine($"Время выполнения: {sw.Elapsed}"); // Здесь логируем

}

static int FindGCD(int a, int b)

{

int gcd = 1; // Инициализируем НОД значением 1

// Перебираем все числа от 1 до минимального из a и b

for (int i = 1; i <= Math.Min(a, b); i++)

{

// Если i является делителем и a и b, обновляем НОД

if (a % i == 0 && b % i == 0)

{

gcd = i;

}

}

return gcd;

}

}

**№4 НОД улучшенный**

using System;

using System.Diagnostics;

class HelloWorld

{

static void Main()

{

var sw = Stopwatch.StartNew();

sw.Start();

Console.Write("Введите первое число: ");

int num1 = int.Parse(Console.ReadLine());

Console.Write("Введите второе число: ");

int num2 = int.Parse(Console.ReadLine());

int gcd = FindGCD(num1, num2);

Console.WriteLine($"Наибольший общий делитель чисел {num1} и {num2} равен {gcd}");

sw.Stop();

Console.WriteLine($"Время выполнения: {sw.Elapsed}"); // Здесь логируем

}

static int FindGCD(int a, int b)

{

while (b != 0)

{

int remainder = a % b;

a = b;

b = remainder;

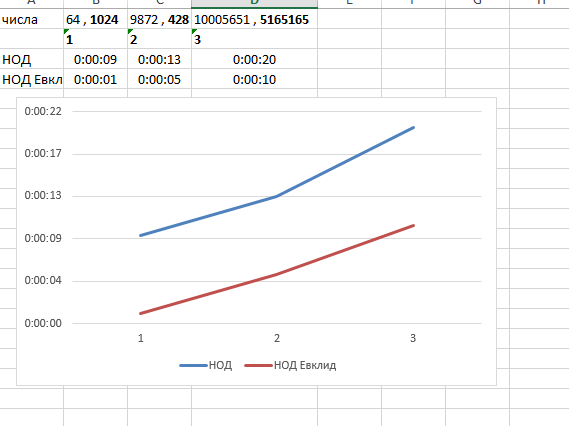
}

return a;

}

}

**График НОД:**



Вывод: В результате выполнения практической работы было произведено сравнение разных алгоритмов работы поиска числа Фибоначи и НОД по результатам была выявлена полезность усовершенствованного алгоритма и его эффективность по времени.