

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 ЛАБОРАТОРНОЙ РАБОТЕ №2
дисциплины «Алгоритмизация»

Выполнил:
Гайчук Дарья Дмитриевна
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», очная
форма обучения

(подпись)

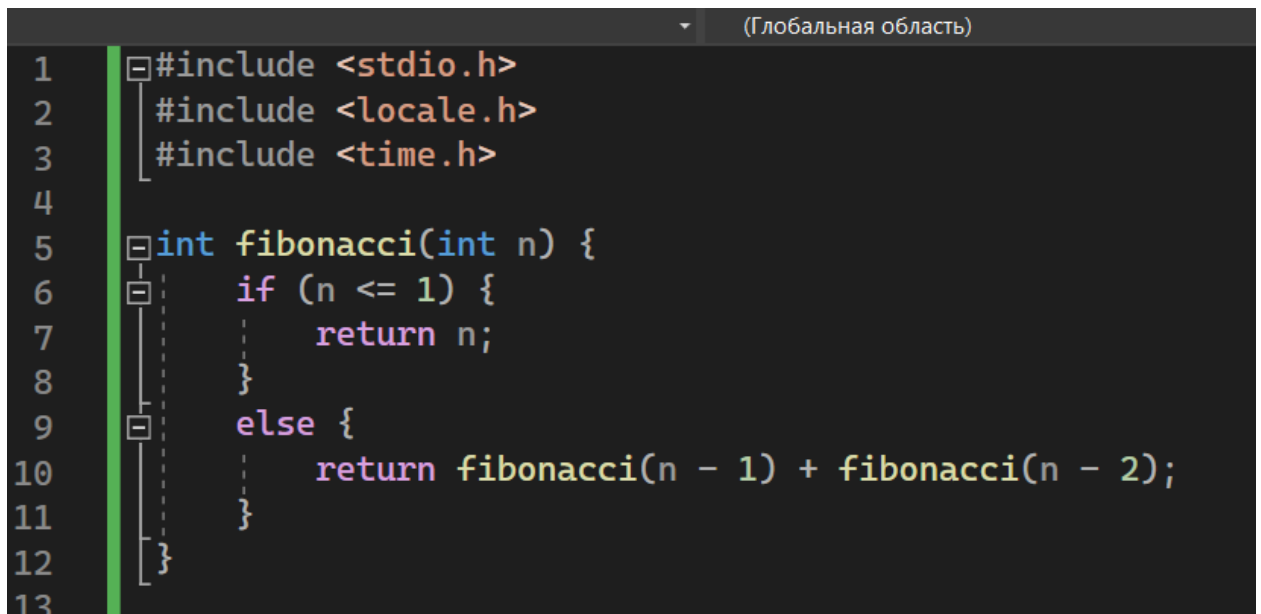
Руководитель практики:
Воронкин Р.А.-доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

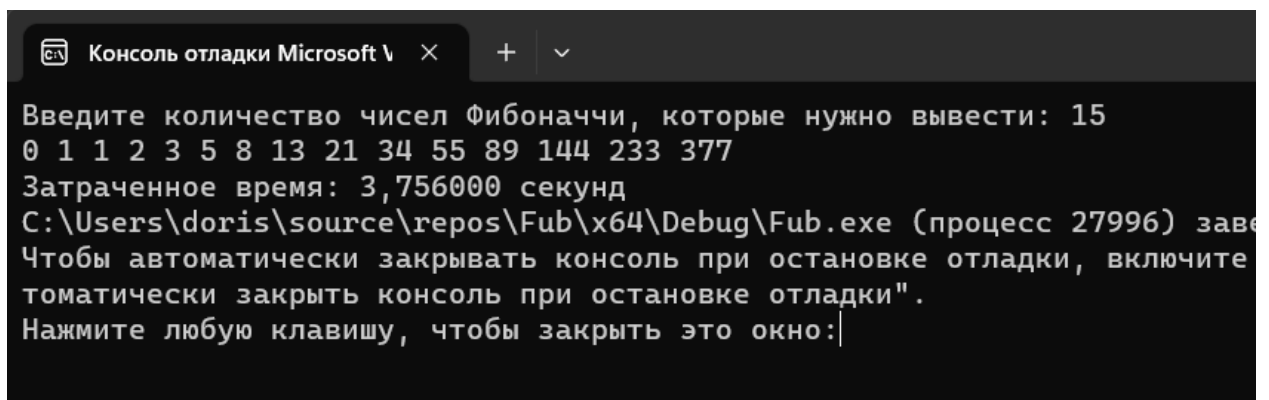
Ставрополь, 2023 г.

Задание 1. Написать программу для вычисления чисел Фибоначчи на основе псевдокода, сравнить время выполнения наивной и улучшенной программы, построить графики сравнения роста времени в обоих случаях.



```
1  #include <stdio.h>
2  #include <locale.h>
3  #include <time.h>
4
5  int fibonacci(int n) {
6      if (n <= 1) {
7          return n;
8      }
9      else {
10         return fibonacci(n - 1) + fibonacci(n - 2);
11     }
12 }
13
```

Рисунок 1. Рекурсивный подход



```
Консоль отладки Microsoft V  x  +  v
Введите количество чисел Фибоначчи, которые нужно вывести: 15
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
Затраченное время: 3,756000 секунд
C:\Users\doris\source\repos\Fub\x64\Debug\Fub.exe (процесс 27996) заведе
Чтобы автоматически закрывать консоль при остановке отладки, включите
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

Рисунок 2. Результат программы

```

cout << "Введите количество чисел Фибоначчи для генерации:";
cin >> n;
fib = new int[n];

if (n <= 0) {
    cout << " Пожалуйста, введите положительное целое число.\n";
    return 0;
}

fib[0] = 0;
fib[1] = 1;

for (int i = 2; i < n; i++) {
    fib[i] = fib[i - 1] + fib[i - 2];
}

```

Рисунок 3. Программа с использованием массива

```

Консоль отладки Microsoft V
Введите количество чисел Фибоначчи для генерации:15
Последовательность Фибоначчи:
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
Затраченное время: 3,210000 секунд
C:\Users\doris\source\repos\FibArray\x64\Debug\FibArray.exe (пр
Чтобы автоматически закрывать консоль при остановке отладки, вкл
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:

```

Рисунок 4. Результат программы

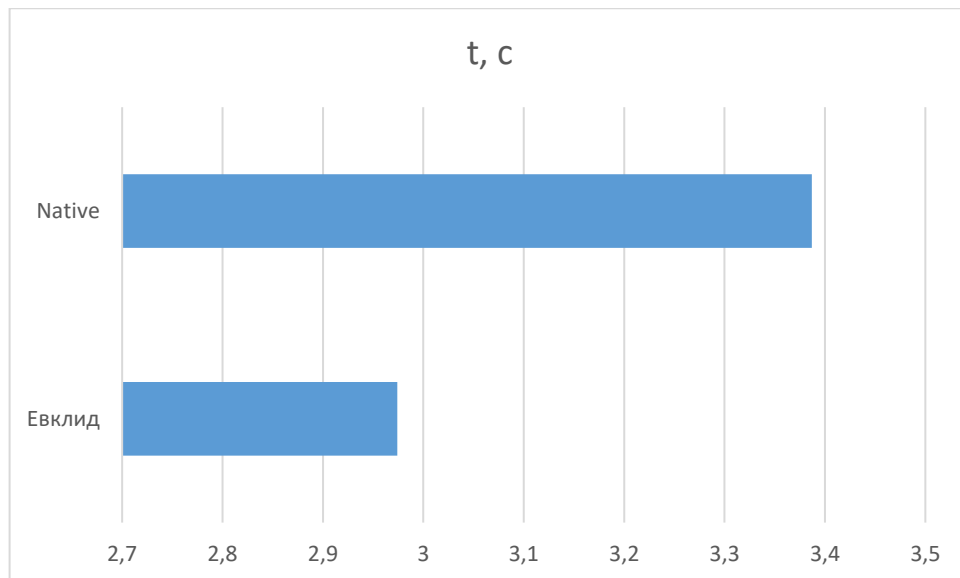


График 1. Сравнение времени выполнения двух программ

Задание 2. Написать программу для вычисления НОД на основе псевдокода, сравнить время выполнения наивной и улучшенной программы, построить графики сравнения роста времени в обоих случаях.

```
int NativeGCD(int a, int b) {
    if (b == 0) {
        return a;
    }
    return NativeGCD(b, a % b);
}

int main() {
    double time_spent = 0.0;
    clock_t begin = clock();

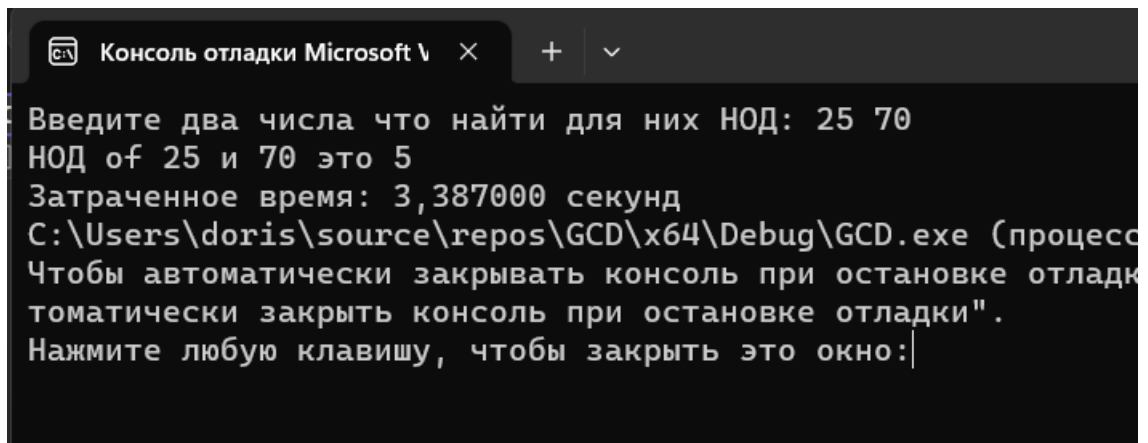
    setlocale(LC_ALL, "Rus");
    int num1, num2;

    std::cout << "Введите два числа что найти для них НОД: ";
    std::cin >> num1 >> num2;

    int gcd = NativeGCD(num1, num2);

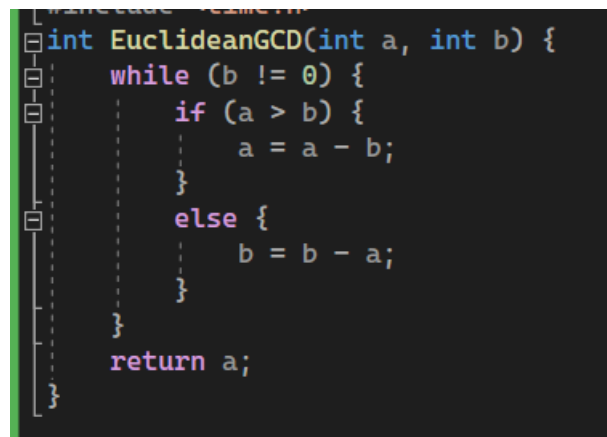
    std::cout << "НОД of " << num1 << " и " << num2 << " это " << gcd << std::endl;
    clock_t end = clock();
    time_spent = (double)(end - begin) / CLOCKS_PER_SEC;
    std::cout << "Время на вычисления: " << time_spent << std::endl;
}
```

Рисунок 5. Программа с использованием наивного алгоритма



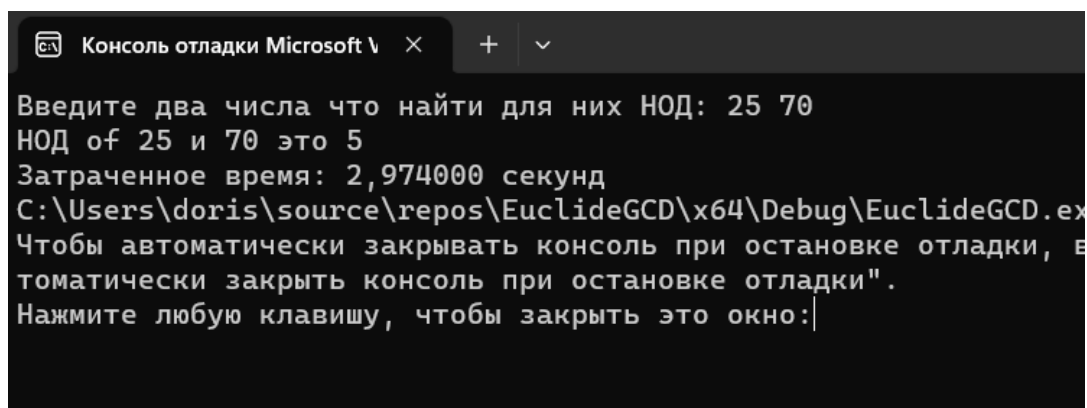
```
Консоль отладки Microsoft V  X + v
Введите два числа что найти для них НОД: 25 70
НОД of 25 и 70 это 5
Затраченное время: 3,387000 секунд
C:\Users\doris\source\repos\GCD\x64\Debug\GCD.exe (процесс
Чтобы автоматически закрывать консоль при остановке отладки, в
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

Рисунок 6. Результат программы



```
int EuclideanGCD(int a, int b) {
    while (b != 0) {
        if (a > b) {
            a = a - b;
        }
        else {
            b = b - a;
        }
    }
    return a;
}
```

Рисунок 7. Программа с использованием метода Евклида



```
Консоль отладки Microsoft V  X + v
Введите два числа что найти для них НОД: 25 70
НОД of 25 и 70 это 5
Затраченное время: 2,974000 секунд
C:\Users\doris\source\repos\EuclideanGCD\x64\Debug\EuclideanGCD.exe
Чтобы автоматически закрывать консоль при остановке отладки, в
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:
```

Рисунок 8. Результат программы

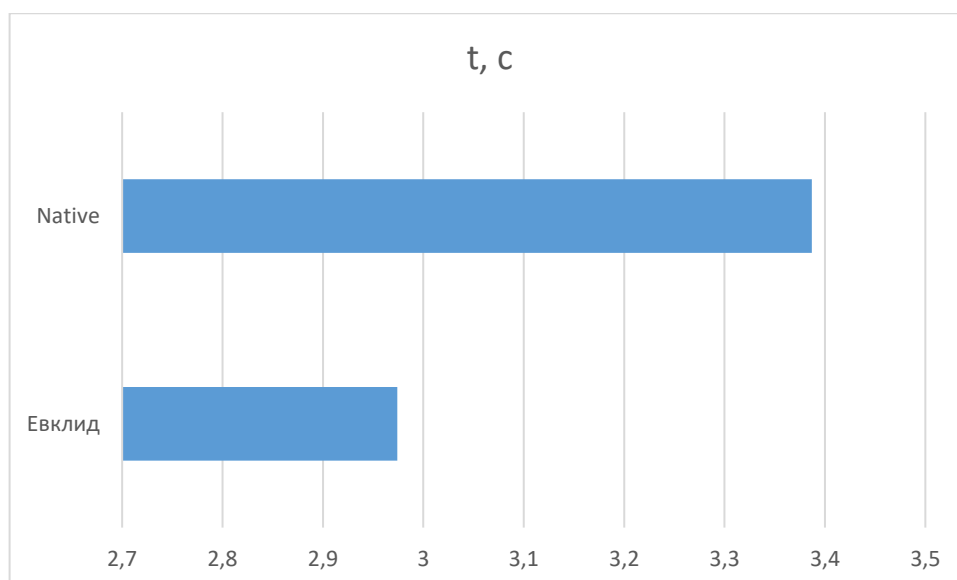


График 2. Сравнение времени, затраченное на выполнение программ

Вывод: в работе было проведено сравнение наивного и улучшенного алгоритма. В процессе сравнения была рассмотрена эффективность и точность работы обоих алгоритмов, на различных наборах данных. Улучшенный алгоритм демонстрирует лучшие результаты по сравнению с наивным алгоритмом, обеспечивает более быструю работу и требует меньшего объема ресурсов, что делает его более эффективным в практическом применении.