

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ
И ПРОГРАММНОЙ ИНЖЕНЕРИИ (КАФЕДРА №43)

ОТЧЕТ ЗАЩИЩЕН С ОЦЕНКОЙ: _____

ПРЕПОДАВАТЕЛЬ:

(должность, учёная степень, звание) / _____ / _____ / _____
(подпись) (дата защиты) (инициалы, фамилия)

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №2

«Изучение принципов функционирования машины Тьюринга»

ПО КУРСУ: « Теория вычислительных процессов»

РАБОТУ ВЫПОЛНИЛА СТУДЕНТКА:

(номер группы) / _____
(инициалы, фамилия)

/ _____ / _____
(подпись студента) (дата отчета)

Санкт-Петербург 2023

1. Цель работы:

Изучить принципы функционирования машины Тьюринга.

2. Постановка задачи:

$$36. x1 * x2 + y$$

В первой части лабораторной работы необходимо написать программу для машины Тьюринга, реализующую вычисление арифметической функции согласно выданному варианту задания. Должна быть составлена совокупность команд **P**. Для выполнения данного задания следует использовать приложение Algo2000

Во второй части лабораторной работы требуется создать программу на языке высокого уровня имитирующую работу машины Тьюринга.

3. Листинг программы на языке высокого уровня с комментариями

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;

namespace TuringMachine
{
    class Program
    {
        static string ReadExample()
        {
            string example = "";
            using (StreamReader exampleFile = new
StreamReader("C:\\Users\\User\\source\\repos\\TVP LR 2\\TVP LR 2\\input.txt"))
            {
                example = exampleFile.ReadLine();
            }
            return example;
        }

        static List<List<string>> ReadCommand()
        {
            List<List<string>> twoDimArray = new List<List<string>>();
            using (StreamReader inputFile = new
StreamReader("C:\\Users\\User\\source\\repos\\TVP LR 2\\TVP LR 2\\command.txt"))
            {
                string line;
                while ((line = inputFile.ReadLine()) != null)
                {
                    List<string> row = line.Split(';').ToList();
                    twoDimArray.Add(row);
                }
            }
            return twoDimArray;
        }

        static void PrintHead(string line, int head)
```

```

    {
        using (StreamWriter outputFile = new
StreamWriter("C:\\Users\\User\\source\\repos\\TVP LR 2\\TVP LR 2\\output.txt",
true))
        {
            outputFile.WriteLine(line);
            for (int i = 0; i < head; i++)
            {
                outputFile.Write(" ");
            }
            outputFile.WriteLine("^");
        }
    }

static void Turing()
{
    int head = 0;
    int stroka = -1;
    int status = 1;
    string line = ReadExample();
    Console.WriteLine(line);

    List<List<string>> data = ReadCommand();
    foreach (var row in data)
    {
        foreach (var element in row)
        {
            Console.Write(element + " ");
        }
        Console.WriteLine();
    }

    while (true)
    {
        PrintHead(line, head);
        Console.WriteLine("Пример: " + line);
        string currentSymbol = line[head].ToString();
        Console.WriteLine("Текущий символ в строке: " + currentSymbol);

        for (int i = 0; i < 6; i++)
        {
            Console.WriteLine("Поиск по " + i + " столбцу матрицы: " +
data[i][0] + " = " + currentSymbol);
            if (data[i][0] == currentSymbol)
            {
                stroka = i;
                break;
            }
        }

        if (stroka == -1)
        {
            Console.WriteLine("Ошибка: Не найдено соответствующее
правило!");
            return;
        }

        Console.WriteLine("Индекс строки = " + stroka);
        string currentCell = data[stroka][status];
        using (StreamWriter outputFile = new
StreamWriter("C:\\Users\\User\\source\\repos\\TVP LR 2\\TVP LR 2\\output.txt",
true))
        {
            outputFile.WriteLine(currentCell);
        }
    }
}

```

```

        if (currentCell == " ")
        {
            Console.WriteLine("Правило не найдено! Пустая ячейка");
            return;
        }

        char replase = currentCell[0];
        char perehod = currentCell[1];
        char status1 = currentCell[2];
        status = int.Parse(status1.ToString()) + 1;
        Console.WriteLine("q " + status);
        Console.WriteLine("1) " + replase + " 2) " + perehod + " 3) " +
status);

        Console.WriteLine("Ячейка в массиве: " + currentCell);
        line = line.Remove(head, 1).Insert(head, replase.ToString());
        if (perehod == '>')
        {
            head++;
        }
        if (perehod == '<')
        {
            head--;
        }
        if (perehod == '!')
        {
            Console.WriteLine("Программа успешно завершена!");
            return;
        }
    }
}

static void Main()
{
    using (StreamWriter outputFile = new
StreamWriter("C:\\Users\\User\\source\\repos\\TVP LR 2\\TVP LR 2\\output.txt"))
    {
        outputFile.Write("");
    }
    Turing();
}
}
}

```

4. Файлы формата .txt

```

1;_>1;1>1;a>1;1<3;1>4;a>6;1>6;1<7
+; ;>1;+<2;+<3;+>5; ; ;
=; ;=>1; ;=<3; ;=!5;=>6;=<7
*,*>4;*>2;*<3; ; ; ; ;
a; ; ;1<2;a>2; ; ; ;a>5
_;_>0;1<3; ;_>0; ; ;1<7;

```

Рисунок 1 «Файл с командами»

```

_11*111+11=_____

```

Рисунок 2 «Файл с примером»

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

⏮

⏪

⏩

⏭

Внешний алфавит: 1+=*a

A\Q	Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7
1	→ Q1	→ Q1	a → Q1	← Q3	→ Q4	a → Q6	→ Q6	← Q7
+		→ Q1	← Q2	← Q3	→ Q5			
=		→ Q1		← Q3		→ Q5	→ Q6	← Q7
*	→ Q4	→ Q2	← Q3					
a			← Q2	a → Q2				a → Q5
Пробел	→ Q0	← Q3		→ Q0			← Q7	

Машина Тьюринга

Программа завершена!

Ок

Рисунок 5 «Лента после запуска»

5. Вывод

В результате выполнения лабораторной работы была написана программа для машины Тьюринга в Algo2000, а также реализован симулятор машины Тьюринга на C#.