Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

**Отчет по лабораторной работе №1**

**Дисциплина:** Низкоуровневое программирование

**Тема:** машина Тьюринга

Выполнил

студент гр. 3530901/90003 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Руднев А.К.

(подпись)

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Алексюк А.О.

(подпись)

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Санкт-Петербург   
2021

**Оглавление**

[**ТЕХНИЧЕСКОЕ ЗАДАНИЕ** 3](#_Toc26066265)

[**МЕТОД РЕШЕНИЯ** 3](#_Toc26066266)

[**ОПИСАНИЕ СОСТОЯНИЙ** 3](#_Toc26066281)

[**РАБОТА ПРОГРАММЫ** 4](#_Toc26066265)

# **ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

Написать программу для умножения двух чисел в унарном коде. Способ задания: 11111\*111111

**МЕТОД РЕШЕНИЯ**

Унарные числа представляются в виде количества единиц: 1-1, 11-2, 111-3 и т.д. Требования к исходным данным: дано выражение, состоящее из первого числа, знака умножения, а также второго числа.

Этапы решения: пример 11\*11

**1)** Ищем конец выражения и записываем туда знак “=” (только для удобства): 11\*11=

**2)** Возвращаемся к первому числу и устанавливаем единицу самого младшего разряда в ноль: 10\*11=

**3)** Переходим ко второму числу и устанавливаем единицу старшего разряда в ноль: 10\*01=

**4)** Переходим за знак “=” и устанавливаем на пустом месте 1: 10\*01=1

**5)** Возвращаемся и повторяем шаги 3, 4, 5, пока второе число не станет состоять полностью из нулей. (подается 10\*01=1 и далее 10\*00=11)

**6)** Восстанавливаем второе число и запускаем пункт 2, пока первое число не станет состоять полностью из нулей: Восстановление - 10\*11=11, цикл с пункта 2: 00\*01=111, 00\*00=1111, 00\*11=1111.

**7)** Стираем всё, что написано ответа – ответ готов: 00\*11=1111 -> 1111

**ОПИСАНИЕ СОСТОЯНИЙ**

**Алфавит**: 01\*=

0 – обозначение уже переписанного числа

1 – обозначение не переписанного числа, а также единица в унарном коде

= - ставится для удобства вычисления

\* - знак, разграничивающий первое и второе унарные числа

**Описание состояний**:

Q1 – читаем все выражение и вставляем в конце знак “=” для удобства ->Q2

Q2 – Возвращаем головку до знака “\*” - > Q3

Q3 – Установка 1 в младшем разряде первого числа в 0 -> Q4

Q4 – Установка 1 в старшего разряде второго числа в 0 -> Q6

Q5 – Возвращение до знака \* -> Q4

Q6 – путь к знаку “=” -> Q9 (если головка становится на “=”) или часть пути возвращения к знаку “\*” -> Q7 (если головка становится на знак “\*”)

Q7 – Проход через 0 в младших разрядах первого числа и установка 0 в младшем разряде первого числа -> Q4, при наличии всех нулей в первом числе – головка доходит до -1 места в начале выражения -> Q8

Q8 – стирание всех символов до знака “=” включительно -> **Завершение работы**

Q9 – Установка 1 после знака всех установленных единиц после знака “=” ->Q5 (возвращаемся до знака “\*”)

**РАБОТА ПРОГРАММЫ**

Пример 11\*11, решение представлено на рисунках 1-5.

Пример 1111\*111, решение представлено на рисунках 6-7.

|  |
| --- |
| Рисунок 1 – Начальные данные №1 |
| Рисунок 2 – Процесс выполнения программы |
| Рисунок 3 – Программа в процессе выполнения |
| Рисунок 4 – Программа в процессе выполнения |
| Рисунок 5 – Полученные данные №1 |
| Рисунок 6 – Начальные данные №2 |
| Рисунок 7 – Полученные данные №2 |

**Вывод:** в ходе выполнения работы была написана программа для умножения двух унарных чисел с помощью симулятора машины Тьюринга, на вход подавалась лента с двумя унарными числами и знаком “\*”, а в итоге на ленте был напечатан ответ. Были получены навыки создание программы с помощью машины Тьюринга, а также опыт работы с ее симулятором.