Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы» направление подготовки: 09.03.04 – «Программная инженерия»

**Лабораторная работа №1.**

**«Модели алгоритма: нормальные алгоритмы Маркова**

**и машина Тьюринга»**

Выполнил студент гр. РИС-24-3б

Черемных Никита Петрович

Проверил:

Доцент кафедры ИТАС

Ольга Андреевна Полякова

(оценка) (подпись)

(дата)

г. Пермь, 2024

**Нормальные алгоритмы Маркова**

Задача 1:

A = {a, b}. Удалить из непустого слова P его первый символ. Пустое слово не менять.

Правила:

1. \*a |→ **.**

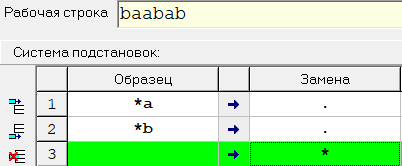
2. \*b |→ **.**

3. → \*

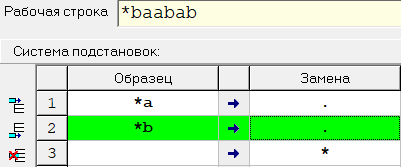
Исходная строка: baabab

Выполнение: baabab→(3) \*baabab→(2) aabab

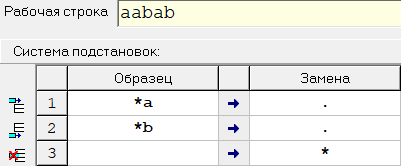
1 шаг:



2 шаг:



Результат:



Задача 2:

A = {a, b, c, d}. В слове P требуется удалить все вхождения символа ‘с’, а затем заменить первое вхождение подслова bb на ddd.

Правила:

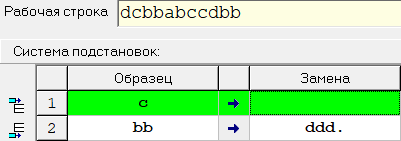
1. c →

2. bb |→ ddd**.**

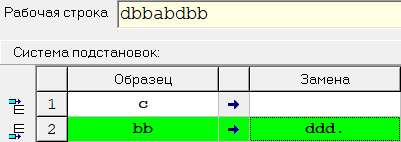
Исходная строка: dcbbabccdbb

Выполнение: dcbbabccdbb→(1) dbbabdbb→(2) ddddabdbb

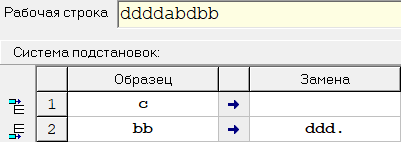
1 шаг:



2 шаг:



Результат:



Задача 3:

A = {a, b}. Требуется приписать символ ‘a’ к концу слова P.

Правила:

1. **\***a→ a\*

2. \*b→ b\*

3. \* |→ a

4. → \*

Исходная строка: aabab

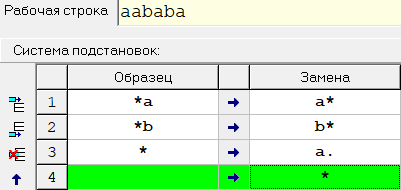
Прежде всего напомним, что формула →*a*приписывает символ *a*слева к слову *P*, а не справа. Чтобы приписать *a*справа, надо сначала пометить конец слова. Для этого воспользуемся спецзнаком, который поместим в конец *P*, а затем заменим его на *a*: P → … → P\* **→** Pa

Что бы поместить спецзнак в конец слова необходимо сначала приписать его слева к слову *P*, а затем «перегнать» звёздочку через все буквы слова:

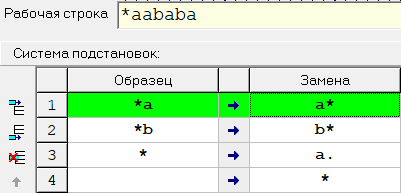
Для того что бы «перегнать» звёздочку через какой-то символ ξ – это замена пары \*ξ на пару ξ\*, которая реализуется формулой \*ξ→ξ\*.

Выполнение: bbab →(4) \*bbab →(2) b\*bab →(2) bb\*ab →(1) bba\*b →(2) bbab\* →(3) bbaba

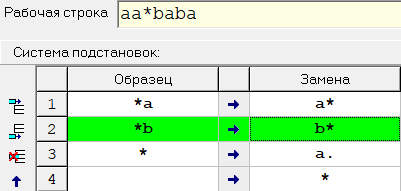
1 шаг:



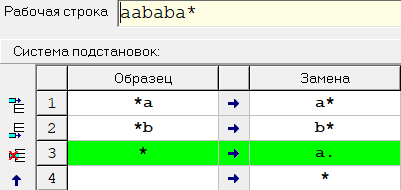
2 шаг:



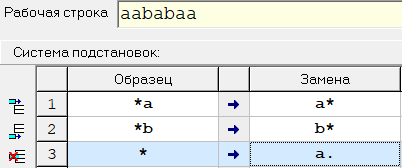
3 шаг:



4 шаг:



Результат:



**Машина Тьюринга**

Задача 1:

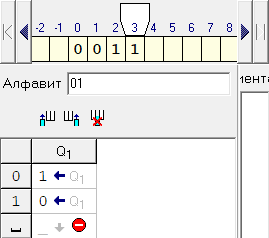
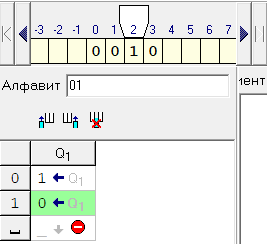
Дано число, состоящее из 0 и 1. Заменить все 0 на 1 и 1 на 0.

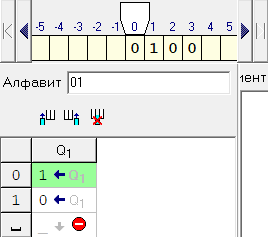
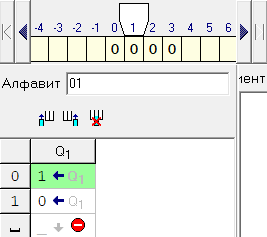
Таблица команд:

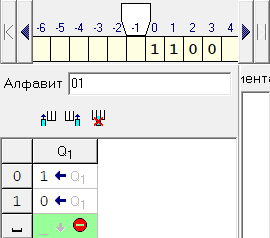
|  |  |
| --- | --- |
|  | Q1 |
| 0 | 1<1 |
| 1 | 0<1 |
| ˽ | ˽.0 |

Выполнение: 0011 → 0010 → 0000 → 0100 → 1100

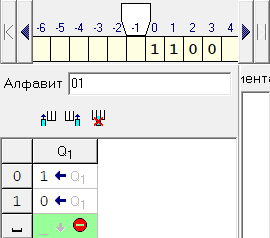
Положение головы на ленте:





Результат:



Задача 2:

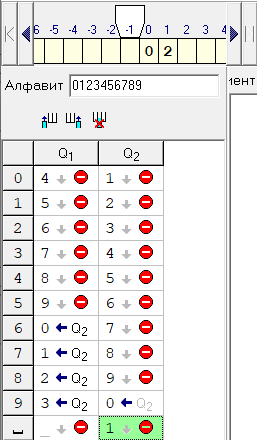
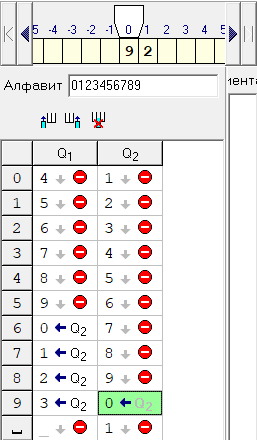
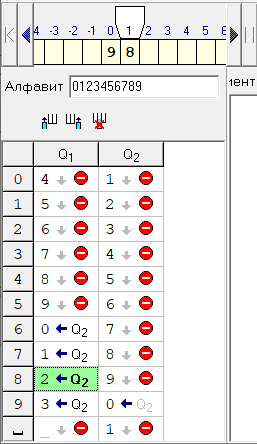
На ввод подаётся случайное число. Прибавить четыре.

Таблица команд:

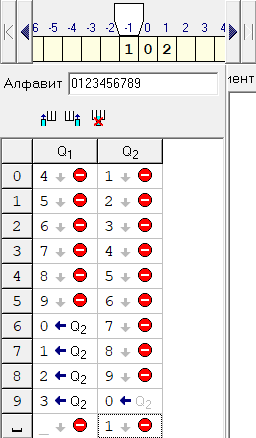
|  |  |  |
| --- | --- | --- |
|  | Q1 | Q2 |
| 0 | 4.0 | 1.0 |
| 1 | 5.0 | 2.0 |
| 2 | 6.0 | 3.0 |
| 3 | 7.0 | 4.0 |
| 4 | 8.0 | 5.0 |
| 5 | 9.0 | 6.0 |
| 6 | 0<2 | 7.0 |
| 7 | 1<2 | 8.0 |
| 8 | 2<2 | 9.0 |
| 9 | 3<2 | 0<2 |
| ˽ |  | 1.0 |

Выполнение: 98 → 92 → 02 → 102

Положение головы на ленте:



Результат:



Алгоритм решения:  
  
1. **Голова** располагается на последней цифре числа. Цифра, на которую указывает **голова**, считывается.

а) Если считанная цифра находится в диапазоне от 0 до 5, то:

* 0 заменяется на 4, 1 – на 5, 2 - на 6, 3 – на 7, 4- на 8, 5 - на 9.
* **Голова** остается на месте.
* Выполнение программы завершается.

б) Если считанная цифра находится в диапазоне от 6 до 9, то:

* К цифре добавляется 4.
* В результате сложения может образоваться новый разряд (например, 6 + 4 = 10, 7 + 4 = 11, 8 + 4 = 12, 9 + 4 = 13).
* 6 заменяется на 0, 7 - на 1, 8 – на 2, 9 – на 3.
* **Голова** сдвигается влево, переходя в состояние q2.

2. **Если программа не завершилась**, учитывается новый разряд (если он образовался). Цифра, на которую указывает **голова**, считывается.  
  а) Если считанная цифра находится в диапазоне от 0 до 8, то:

* Она заменяется цифрой, на 1 больше.
* **Голова** остается на месте.
* Выполнение программы завершается.

б) Если считанная цифра равна 9, то:

* К ней добавляется 1 из нового разряда, что приводит к образованию ещё одного нового разряда (9 + 1 = 10).
* 9 меняется на 0.
* **Голова** сдвигается влево, переходя в состояние q2.

3. **Если программа не завершилась**, и **голова** находится на пустой строке, то:

* Пустая строка заменяется на 1.
* **Голова** остается на месте.
* Выполнение программы завершается.

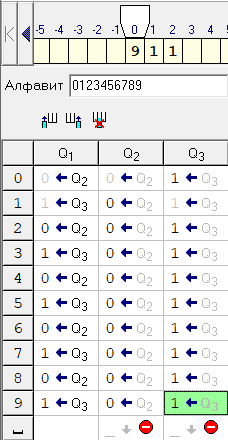
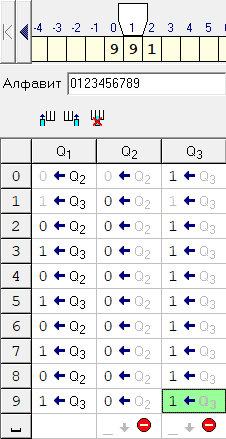
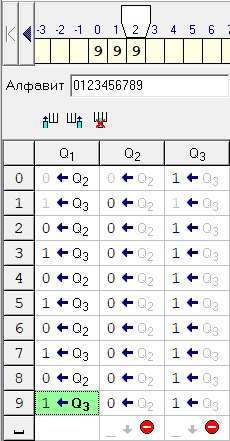
Задача 3:

На ввод подаётся случайное число. Если число чётное – заменить в нём все цифры на 0, иначе на 1.

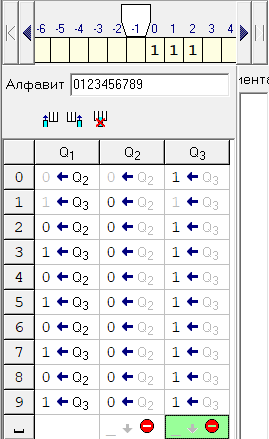
Таблица команд:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Q1 | Q2 | Q3 |
| 0 | 0<2 | 0<2 | 1<3 |
| 1 | 1<3 | 0<2 | 1<3 |
| 2 | 0<2 | 0<2 | 1<3 |
| 3 | 1<3 | 0<2 | 1<3 |
| 4 | 0<2 | 0<2 | 1<3 |
| 5 | 1<3 | 0<2 | 1<3 |
| 6 | 0<2 | 0<2 | 1<3 |
| 7 | 1<3 | 0<2 | 1<3 |
| 8 | 0<2 | 0<2 | 1<3 |
| 9 | 1<3 | 0<2 | 1<3 |
| □ |  | \_.0 | \_.0 |

Выполнение: 999 → 991 → 911 → 111



Результат:



Алгоритм решения:  
 **Голова** располагается на последней цифре числа. Цифра, на которую указывает **голова**, считывается.

а) Если считанная цифра является чётной (0, 2, 4, 6, 8), то:

* Цифра заменяется на 0.
* **Голова** сдвигается влево, переходя в состояние q2
* В состоянии q2 все последующие элементы числа заменяются на 0

б) Если считанная цифра является нечётной (1, 3, 5, 7, 9), то:

* Цифра заменяется на 1.
* Голова сдвигается влево, переходя в состояние q3.
* В состоянии q3 все последующие элементы числа заменяются на 1