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: 09.03.01– «Информатика и вычислительная техника»

**Лабораторная работа № 1**

**«Алгоритмы Маркова.»**

Выполнил студент гр. ИВТ-24-1б

Оглезнев Никита Михайлович

Проверил:

Доцент кафедры ИТАС

Д.В. Яруллин

(оценка) (подпись)

(дата)

г. Пермь, 2024

**Алгоритмы Маркова**

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

Ход решения:

1. Возьмем за слово P произвольную строку «abbbaaa»
2. Добавляем в начало слова специальный символ «\*», отличающийся от входящих в алфавит. Это требуется для того, чтобы определить, что заменяемый символ является первым.
3. Заменяем все возможные комбинации, имеющие вид «\*x», где x - любой символ, входящий в алфавит, один раз, после чего принудительно завершаем выполнение алгоритма.

Получим алгоритм (рисунок 1):

1. \*a |->
2. \*b |->
3. -> \*

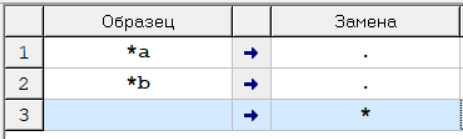


Рисунок 1 – решение алгоритма Маркова 1

Преобразования данной строки будут выглядеть:

«abbbaaa» -> «\*abbbaaa» |-> «bbbaaa»

1. **A = {a, b, c, d}. В слове P требуется удалить все вхождения символа c, а затем заменить первое вхождение подслова «bb» bb на «ddd».**

Ход решения:

1. Возьмем за слово P произвольную строку «acccabbccbbccddcccd»
2. Заменяем все символы «c» на пустую строку
3. Заменяем первое вхождение подстроки «bb» и принудительно завершаем выполнение алгоритма.

Получим алгоритм (рисунок 2):

1. c ->
2. bb |-> ddd

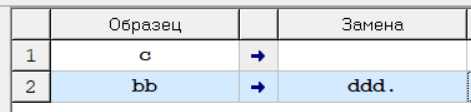


Рисунок 2 – решение алгоритма Маркова 2

Преобразования данной строки будут выглядеть: «acccabbccbbccddcccd» -> «accabbccbbccddcccd» -> «accabbccbbccddcccd» -> «acabbccbbccddcccd» -> «aabbccbbccddcccd» -> «aabbcbbccddcccd» -> «aabbbbccddcccd» -> «aabbbbcddcccd» -> «aabbbbddcccd» -> «aabbbbddccd» -> «aabbbbddcd» -> «aabbbbddd» |-> «aadddbbddd»

1. **A = {a, b}. Требуется приписать символ «a» к концу слова P.**

Ход решения:

1. Возьмем за слово P произвольную строку «abbbaaabab»
2. Добавляем в начало слова специальный символ «\*», отличающийся от входящих в алфавит.
3. Заменяем подстроку вида «\*x» на «x\*», где x – любой символ, входящий в алфавит. Таким образом получаем смещение специального символа на 1 разряд за одну операцию. Выполняем это действие до тех пор, пока специальный символ «\*» не достигнет крайнего положения в слове.
4. Заменяем подстроку «a\*» на «a» и принудительно завершаем выполнение алгоритма.

Получим алгоритм (рисунок 3):

1. \*a -> a\*
2. \*b -> b\*
3. \* |-> a
4. -> \*

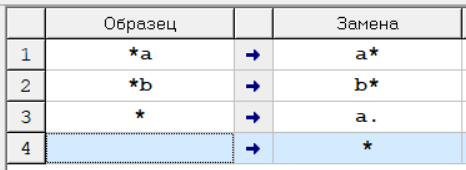


Рисунок 3 – решение алгоритма Маркова 3

Преобразования данной строки будут выглядеть: «abbbaaabab» -> «\*abbbaaabab» -> «a\*bbbaaabab» -> «ab\*bbaaabab» -> «abb\*baaabab» -> «abbb\*aaabab» -> «abbba\*aabab» -> «abbbaa\*abab» -> «abbbaaa\*bab» -> «abbbaaab\*ab» -> «abbbaaaba\*b» -> «abbbaaabab\*» |-> «abbbaaabaa»

**Результаты**

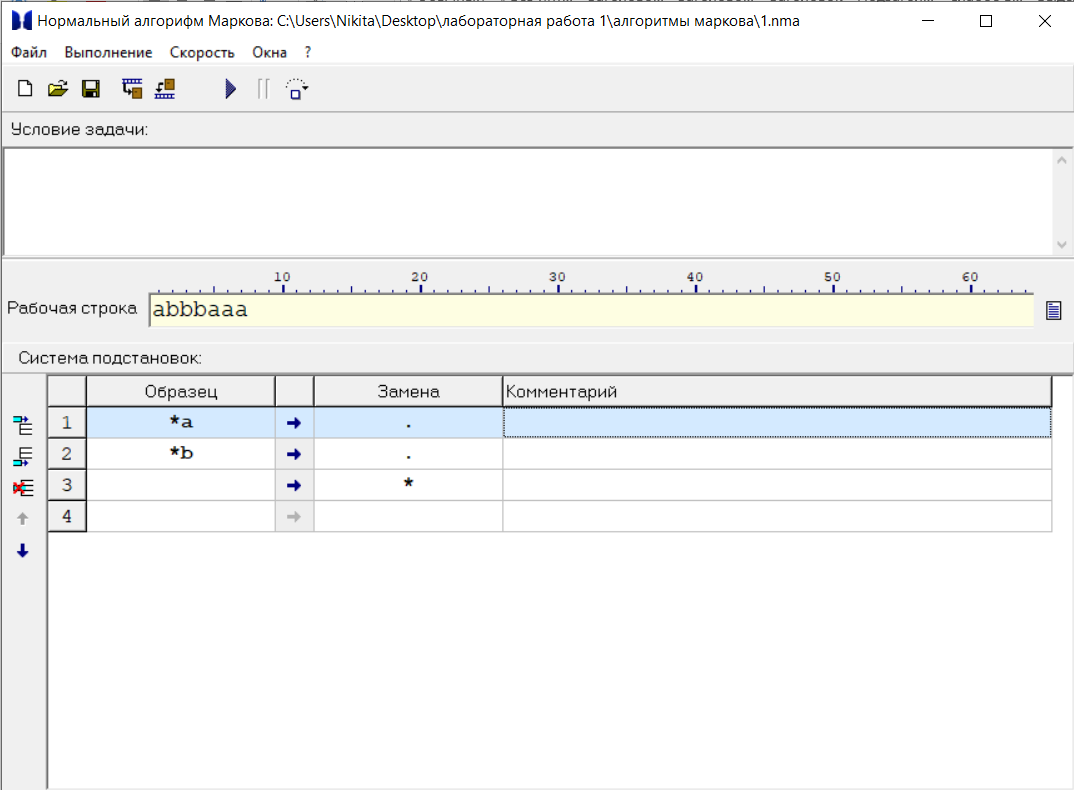


Рисунок 7 – Исходное значение задачи 1

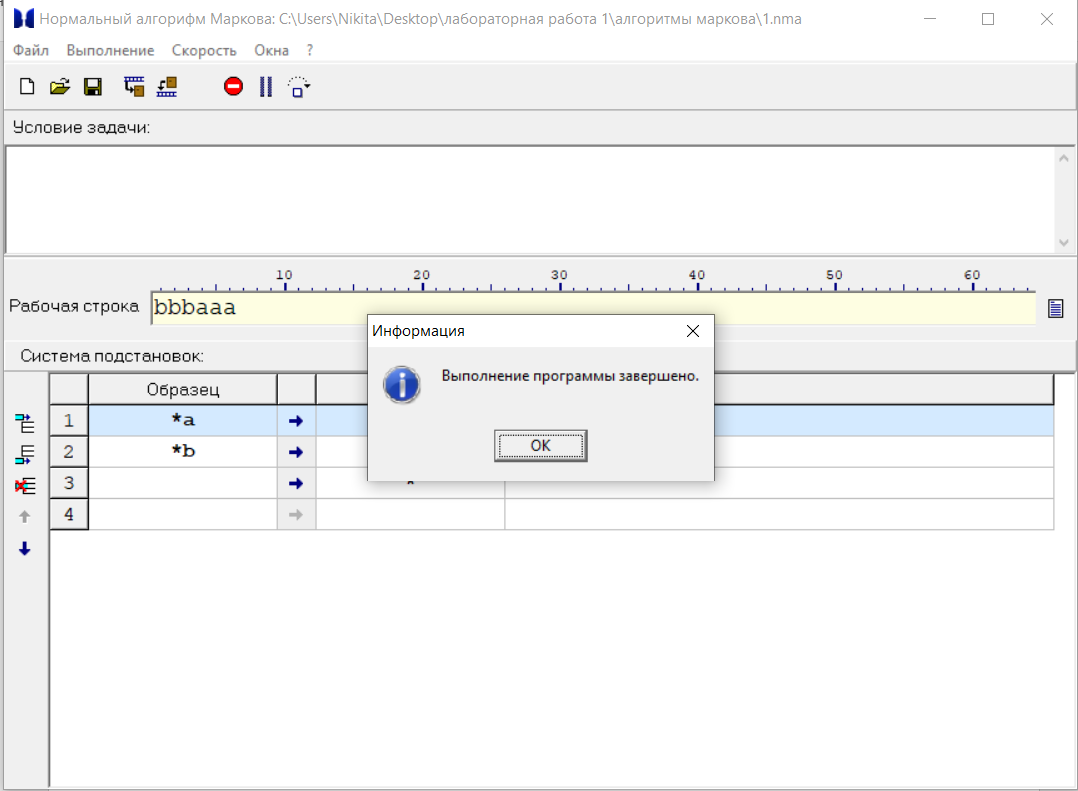


Рисунок 8 – Результат выполнения задачи 1

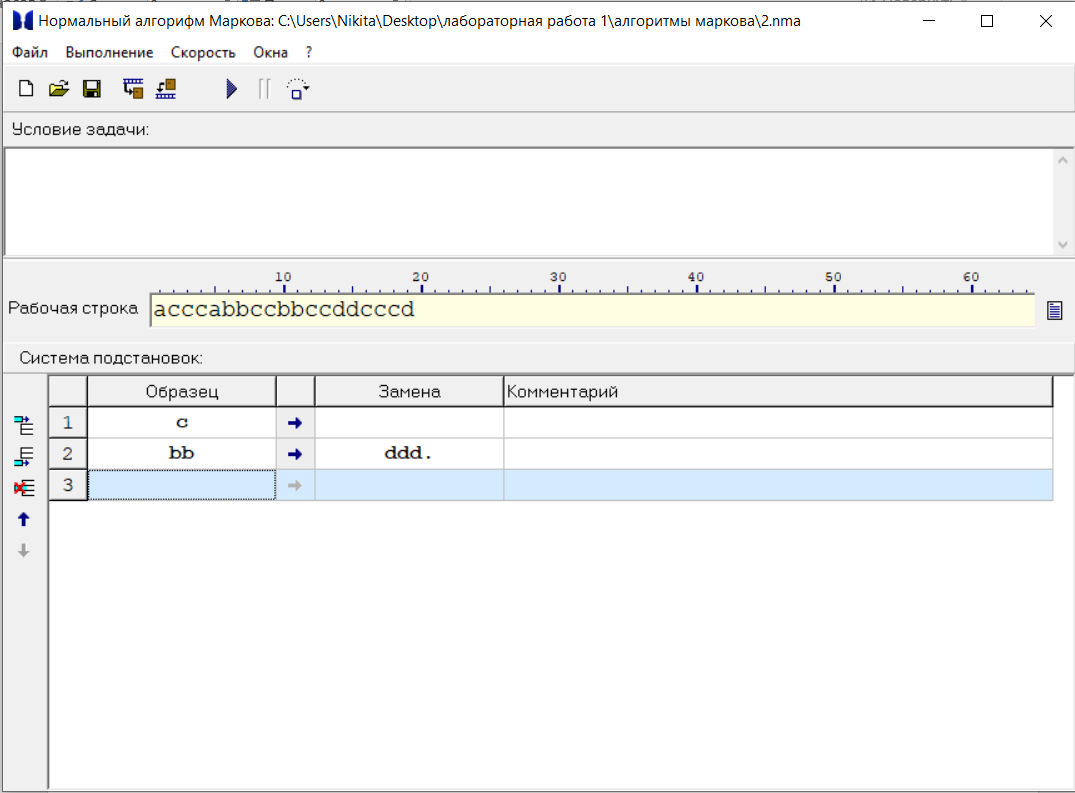


Рисунок 9 – Исходное значение задачи 2

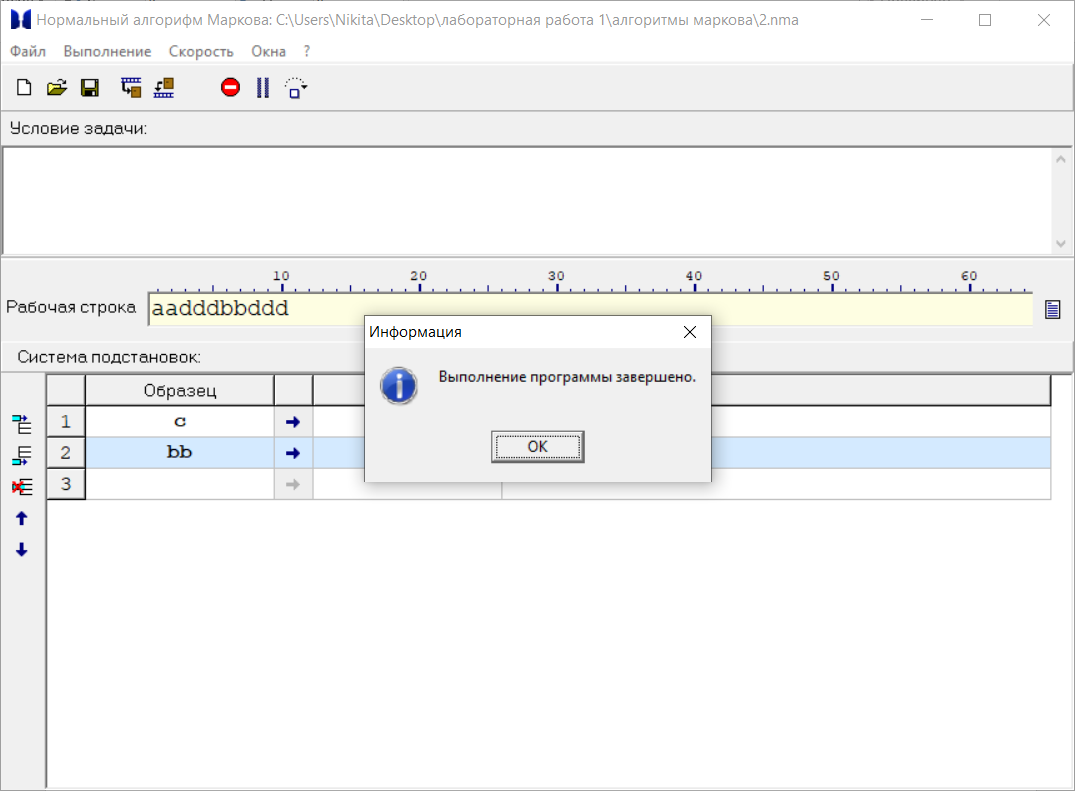


Рисунок 10 – Результат выполнения задачи 2

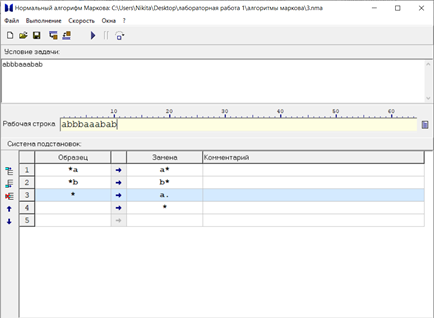


Рисунок 11 – Исходное значение задачи 3

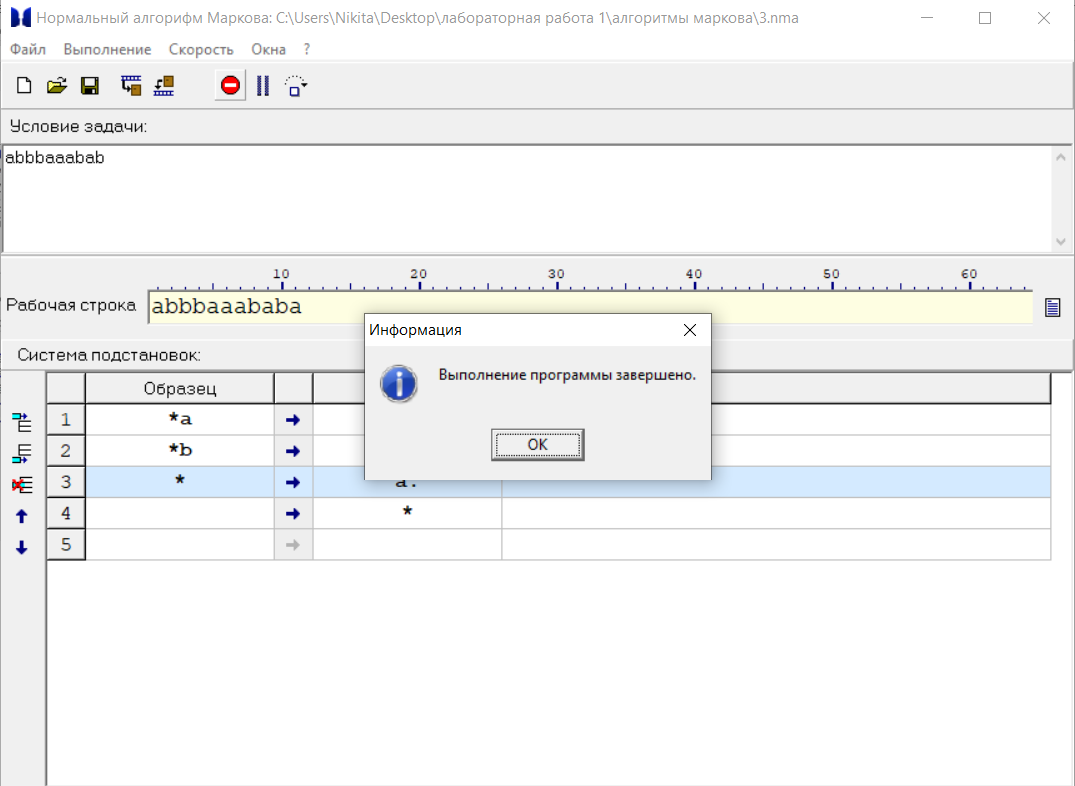


Рисунок 12 – Результат выполнения задачи 3