

## **Документ продукта по учебной практике**

### **Спецификация**

#### **Задание:**

Задачей проекта является создание интерактивного приложения на языке Kotlin, которое отображает работу алгоритма Ахо-Корасик и позволяет взаимодействовать с ним, используя графический интерфейс.

#### **Работа программы:**

Приложение открывает окно с двумя ячейками для ввода текста, в которые пользователь вводит текст в первое окно и набор шаблонов во второе окно(рис. 2), строит по ним бор в терминах алгоритма Ахо-Корасик, после чего выводит на экран изображение конечного автомата(рис. 3). Через дополнительное окно вводятся два номера состояния по форме(“номер1”->”номер2”)(рис.4), после чего в автомат добавляется ребро из первого состояния во второе, автомат перестраивается, приложение анализирует изменённый граф и по нему собирает набор шаблонов в зависимости от данных введённых пользователем(рис. 5).

#### **Описание пользовательского интерфейса:**

При запуске приложение имеет два поля для ввода текста(рис. 1), в одно вписывается текст, в другое набор шаблонов разделённых символом “#”(пример набора шаблонов: #ab#ba#aba)(рис. 2). После подтверждения данных на экран выводится граф представляющий собой конечный автомат, который и является графическим отображением бора с выводом всех вхождений шаблонов в текст по принципу ("номер шаблона" "позиция вхождения")(рис. 3). В окне с графом также имеется окно для ввода текста, благодаря которому можно добавлять рёбра по форме(“номер1”->”номер2”)(рис. 4 и 5).

**Step 1** начало работы

Рис. 1 – примерная иллюстрация открывшегося приложения

**Step 2** ввод данных

Рис. 2 – ввод данных для построения бора и нахождения подстрок в строке

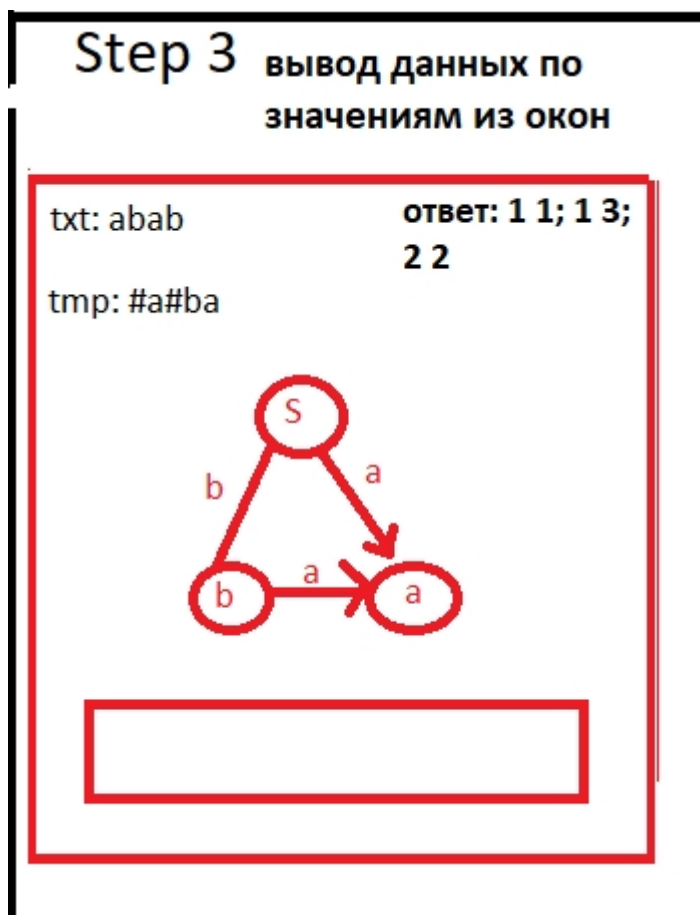


Рис. 3 – иллюстрация вывода

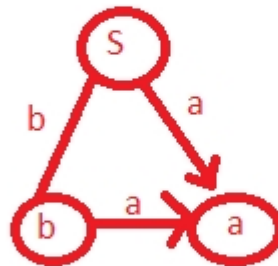
**Step 4** взаимодействие с экраном  
добавление вершины

txt: abab

ответ: 1 1; 1 3;

tmp: #a#ba

2 2



**a->b**

Рис. 4 – иллюстрация записи добавления вершины

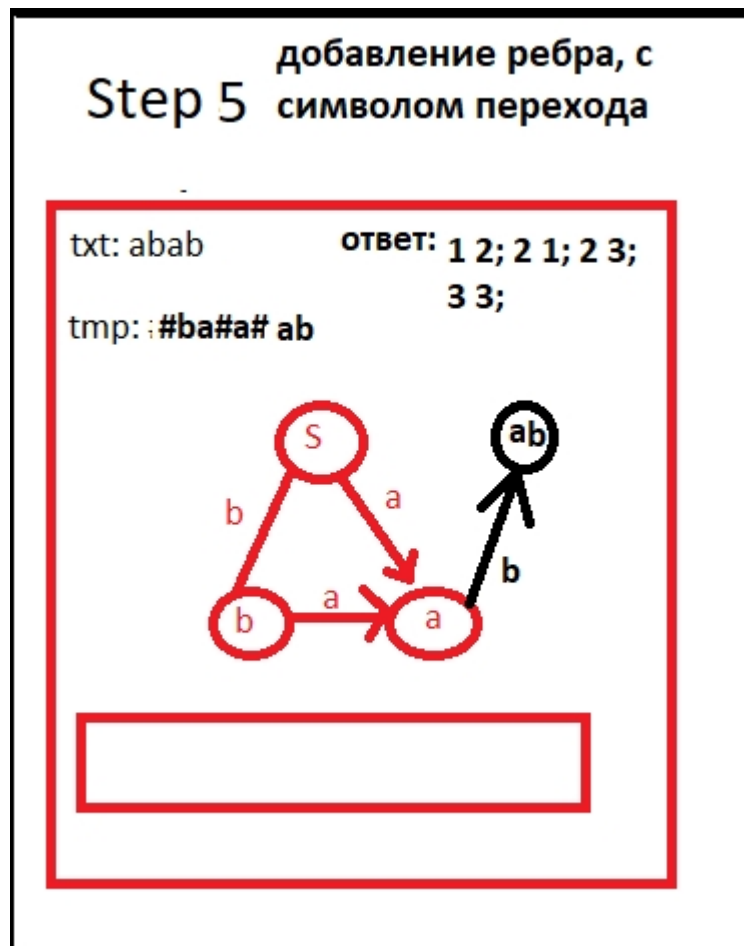


Рис. 5 – иллюстрация изменённого автомата после добавления ребра

### **Формат входных и выходных данных**

**Вход:** строка представляющая собой текст в первом окне, серия строк разделённых разделительным знаком “#” во втором окне.

**Вывод:** Все вхождения шаблонов в текст по принципу ("номер шаблона" "позиция вхождения"), а также рисунок конечного автомата по заданным данным.

### **План разработки**

5 июля: согласование спецификации и плана разработки.

7 июля: создание рабочей версии программы и создание заготовки пользовательского интерфейса; возможно создание полноценной версии пользовательского интерфейса.

8 июля: создание полноценной версии пользовательского интерфейса

9 июля: доработка программы различными функциями

10 июля: сдача финальной версии с отчётом

**Разделение обязанностей в группе,**

**Голов – общение с преподавателем, оформление документов,  
алгоритмическая часть**

**Токун – Работа с графическим интерфейсом.**