```
package com.example.finitestatemachine;
import javafx.stage.Stage;
import java.io.IOException;
public class Main extends Application {
    @Override
        FXMLLoader fxmlLoader = new
       Scene scene = new Scene(fxmlLoader.load(), 807, 300);
import javafx.scene.control.Alert.AlertType;
       Alert alert = new Alert(type);
       alert.setTitle(title);
       alert.setHeaderText(header);
       alert.setContentText(content);
       alert.show();
```

```
import java.io.IOException;
   @FXML
   private Button testButton; // Button to start the test.
   @FXML
   private TextField textInput; // Text field to input the automata.
   private TextArea textInfo; // Text area to show the operations.
   @FXML
   private Label opLabel; // Label to show the Operations/Operações
   @FXML
   private ImageView creatorLogo; // ImageView to show the creator
   @FXML
   ImageView afdImage; // ImageView to show the finite state design.
   private Label outputLabel; // Label to show below the text area
   private Button helpButton; // Button to show the help page.
   @FXML
   private Label helpLabel; // Label to show the text indicating the
   @FXML
   private Label creatorFooter; // Label to show the credits about
   @FXML
   private Button preset1; // Button to show an example of an
   @FXML
   private Button preset2; // Button to show an example of a non-
   protected void onPreset1ButtonClickAction() {
       String automata = "0100101101";
       operations (automata);
   @FXML
       String automata = "01101001";
       operations (automata);
```

```
@FXML
        String automata = textInput.getText();
        if (automata.equals("")) {
teste.
                     """, Alert.AlertType.WARNING);
            automata = textInput.getText();
        } else if (!automataValidation(automata)) {
certifique-se de que a cadeia possui apenas números (i.e:
00101000110).
OBS: Certifique-se de utilizar o alfabeto \{0,\ 1\} (ou seja, apenas números 1 e 0). Caso contrário, a cadeia será negada
            automata = textInput.getText();
            operations (automata);
    @FXML
aceita cadeias com quantidades ímpares de números 1.
                As operações realizadas pelo AFD são mostradas no
quadro ao lado. Abaixo do quadro será exibida a informação de
aceitação ou negação do AFD inserido.
                Estado Inicial: {q0} | Estado de Aceitação: {q1}
                Para ver um rápido exemplo do funcionamento deste
simulador, faça uso dos botões "Preset 1" e "Preset 2". O botão
botão "Preset 2" irá exemplificar uma cadeia inválida (negada).
                 Para iniciar o teste, insira um AFD no bloco de texto
"Informe o autômato" e clique no botão Testar que fica logo abaixo do
    private boolean automataValidation(String automata) {
        boolean result = true;
             if (!Character.isDigit(automata.charAt(i))) {
                 result = false;
                result = false;
```

```
return result;
        int countOp = 2;
        textInfo.setText("Cadeia informada: " + automata);
        textInfo.appendText("\nEstado Inicial: q0");
            textInfo.appendText("\n1a Operação: q0 -> q0");
            currentState = "q0";
            textInfo.appendText("\n1a Operação: q0 -> q1");
            currentState = "q1";
                if (currentState.equals("q0") && automata.charAt(i) ==
                    textInfo.appendText("\n" + countOp + "a Operação:
q0 -> q0");
                    currentState = "q0";
                else if (currentState.equals("q0") &&
                    textInfo.appendText("\n" + countOp + "a Operação:
                    currentState = "q1";
                    textInfo.appendText("\n" + countOp + "a Operação:
                    currentState = "q1";
                else if (currentState.equals("q1") &&
                    textInfo.appendText("\n" + countOp + "a Operação:
                    currentState = "q0";
                countOp++;
        textInfo.appendText("\nEstado Final: " + currentState);
        if (currentState.equals("q1")) {
            outputLabel.setText("Cadeia aceita!");
            outputLabel.setText("Cadeia negada!");
    public void initialize(URL url, ResourceBundle rb) {
Image(getClass().getResourceAsStream("com/example/finitestatemachine/c
l gnl.png"));
        Image afd = new
Image(getClass().getResourceAsStream("com/example/finitestatemachine/a
fd_rps.png"));
        creatorLogo.setImage(cl);
```

```
afdImage.setImage(afd);
<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TextArea?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image??</pre>
<?import javafx.scene.image.ImageView?>
          javafx.scene.layout.Pane?>
<?import javafx.scene.text.Font?>
<Pane fx:id="Window" maxHeight="-Infinity" maxWidth="-Infinity"
minHeight="-Infinity" minWidth="-Infinity" prefHeight="300.0"</pre>
prefWidth="803.0" xmlns="http://javafx.com/javafx/17"
xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.finitestatemachine.Controller">
      <ImageView fx:id="afdImage" fitHeight="474.0" fitWidth="612.0"</pre>
preserveRatio="true">
             <Image url="@afd rps.png" />
      <TextField fx:id="textInput" layoutX="41.0" layoutY="181.0"</pre>
promptText="Informe o autômato" />
      <TextArea fx:id="textInfo" layoutX="565.0" layoutY="42.0"</pre>
      <Label fx:id="outputLabel" layoutX="620.0" layoutY="244.0"</pre>
      <Button fx:id="helpButton" layoutX="212.0" layoutY="181.0"</pre>
mnemonicParsing="false" onAction="#onHelpButtonClickAction"
      <Label fx:id="helpLabel" layoutX="41.0" layoutY="107.0"</pre>
prefHeight="61.0" prefWidth="200.0" text="Para utilizar o programa de
       <Label fx:id="creatorFooter" layoutX="170.0" layoutY="284.0"</pre>
text="Criado por: Danilo " qiqa" Silva | Ciência da
Computação - Universidade Católica de Pernambuco (Unicap).">
             <Font size="10.0" />
      <ImageView fx:id="creatorLogo" fitHeight="91.0" fitWidth="108.0"</pre>
layoutX="87.0" layoutY="10.0" nodeOrientation="INHERIT"
             <Image url="@cl gnl.png" />
```