软件体系结构

《软件体系结构作业十六》

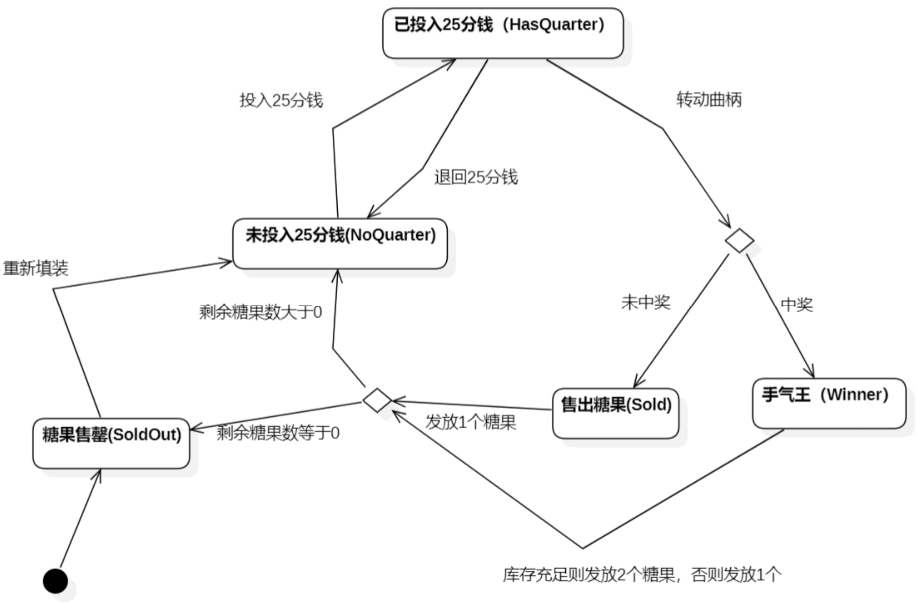
学　号 22920212204396

姓　名 黄子安

2024 年 5 月 19 日

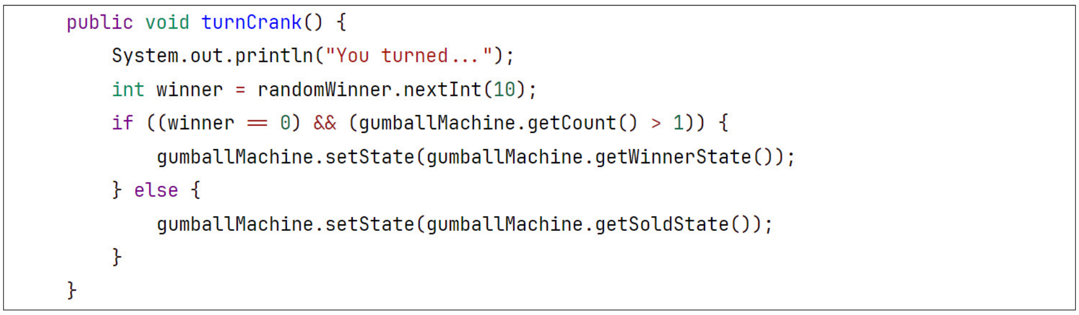
一、阅读 Gumballstate 源码并改写成你想的（GUI）

该例子使用了状态模式，即将糖果机的 **5 个状态使用类来封装**，之后定义对应状态下对于相同行为的不同响应，首先明确一下糖果机的 5 个状态以及对应的状态机图



这 5 个状态类都定义了 5 个行为方法，分别为 insertQuarter，ejectQuarter，turnCrank，dispense，refill 但是只有当前状态与行为匹配的时候才会发生状态转移，否则是不会发生状态转移的

这里状态转移有一个特殊之处是进入 Winner 状态是随机进入的，如果当前糖果数大于 1 就有十分之一的概率拿到两个糖果



```
    public void turnCrank() {
        System.out.println("You turned...");
        int winner = randomWinner.nextInt(10);
        if ((winner == 0) && (gumballMachine.getCount() > 1)) {
            gumballMachine.setState(gumballMachine.getWinnerState());
        } else {
            gumballMachine.setState(gumballMachine.getSoldState());
        }
    }
```

之后使用修改成 GUI 模式，这里使用 JavaFX 作为框架，先编写对应的 FXML 文件，主要包括几个按钮和一个仿真的显示屏，代码如下：

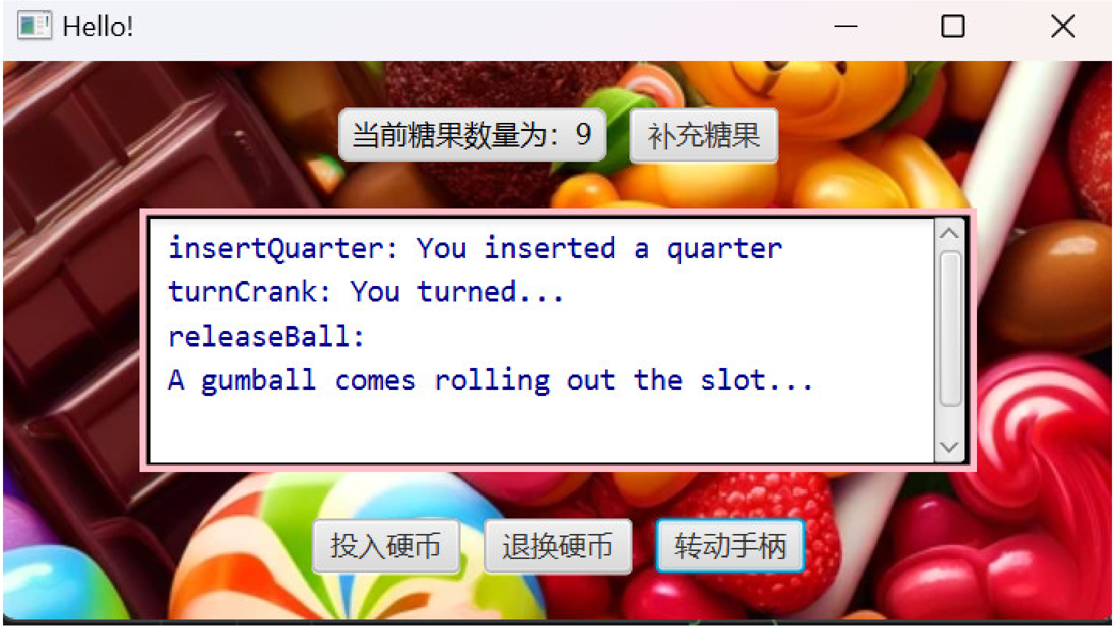
```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.*?>
<?import javafx.scene.control.*?>
<?import javafx.scene.layout.*?>

<VBox fx:id="rootVBox" alignment="CENTER" spacing="20.0" xmlns:fx="http://javafx.com/fxml/1"
xmlns="http://javafx.com/javafx/17.0.2-ea"
fx:controller="cn.xmu.edu.gumball.GumballController">
    <padding>
        <Insets bottom="20.0" left="20.0" right="20.0" top="20.0" />
    </padding>
    <HBox alignment="CENTER" spacing="10.0">
        <Label fx:id="textLabel" style="
    -fx-background-color: linear-gradient(#f2f2f2, #d6d6d6);
    -fx-background-radius: 5;
    -fx-background-insets: 0;
    -fx-text-fill: black;
    -fx-padding: 3 5 3 5;
    -fx-border-color: #bdbdbd;
    -fx-border-width: 1px;
    -fx-border-radius: 5;"/>
        <Button fx:id="refillButton" mnemonicParsing="false" text="补充糖果"
onAction="#refillButtonPressed" />
    </HBox>
    <TextArea fx:id="textArea" prefHeight="200.0" minWidth="360.0" maxWidth="360.0"
              style="-fx-background-color: black;
                     -fx-text-fill: darkblue;
                     -fx-font-family: 'Consolas';
                     -fx-font-size: 13px;
                     -fx-border-color: pink;
                     -fx-border-width: 3px;
                     -fx-control-inner-background: white;"/>

    <VBox prefHeight="80.0" prefWidth="100.0" spacing="5.0">
        <HBox prefHeight="45.0" prefWidth="200.0" alignment="CENTER" spacing="10.0">
            <Button fx:id="insertQuarterButton" mnemonicParsing="false" text="投入硬币"
                    onAction="#insertQuarterButtonPressed"/>
            <Button fx:id="ejectQuarterButton" mnemonicParsing="false" text="退换硬币"
                    onAction="#ejectQuarterButtonPressed"/>
            <Button fx:id="turnCrankButton" mnemonicParsing="false" text="转动手柄"
                    onAction="#turnCrankButtonPressed"/>
        </HBox>
    </VBox>
</VBox>
```

增加一个 Controller 和 JavaFX 的全局启动类

```
package cn.xmu.edu.gumball;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.scene.control.*;
import javafx.scene.image.Image;
import javafx.scene.layout.*;

import java.io.File;

public class GumballController {
    @FXML
    private VBox rootVBox;
    public TextArea textArea;
    public Button insertQuarterButton;
    public Button ejectQuarterButton;
    public Button turnCrankButton;
    public Button dispenseButton;
    public Button refillButton;
    public Button exitButton;
    public Label textLabel;

    private GumballMachine gumballMachine;

    @FXML
    public void initialize() {
        Image backgroundImage = new Image(getClass().getResource("/img.png")
                                .toExternalForm());
        BackgroundImage background = new BackgroundImage(backgroundImage,
                        BackgroundRepeat.NO_REPEAT,
                        BackgroundRepeat.NO_REPEAT,
                        BackgroundPosition.CENTER,
                        BackgroundSize.DEFAULT);
        rootVBox.setBackground(new Background(background));
        gumballMachine = new GumballMachine(10);
        showLabel();
    }


    public void insertQuarterButtonPressed(ActionEvent actionEvent) {
        textArea.appendText(gumballMachine.insertQuarter());
    }

    public void ejectQuarterButtonPressed(ActionEvent actionEvent) {
        textArea.appendText(gumballMachine.ejectQuarter());
    }

    public void turnCrankButtonPressed(ActionEvent actionEvent) {
        textArea.appendText(gumballMachine.turnCrank());
        showLabel();
    }

    public void dispenseButtonPressed(ActionEvent actionEvent) {

    }

    public void refillButtonPressed(ActionEvent actionEvent) {
        TextInputDialog dialog = new TextInputDialog();
        dialog.setTitle("补充糖果机");
        dialog.setHeaderText("请输入补充的糖果的个数:");
        dialog.setContentText("糖果个数:");

        dialog.showAndWait().ifPresent(input -> {
            try {
                int numGumballs = Integer.parseInt(input);
                if (numGumballs > 0) {
                    textArea.appendText(gumballMachine.refill(numGumballs));
                    showLabel();
                } else {
                    Alert alert = new Alert(Alert.AlertType.ERROR);
                    alert.setTitle("Error");
                    alert.setHeaderText(null);
                    alert.setContentText("Please enter a number greater than 0.");
                    alert.showAndWait();
                }
            } catch (NumberFormatException e) {
                Alert alert = new Alert(Alert.AlertType.ERROR);
                alert.setTitle("Error");
                alert.setHeaderText(null);
                alert.setContentText("Invalid input. Please enter a valid number.");
                alert.showAndWait();
            }
        });
    }

    public void exitButtonPressed(ActionEvent actionEvent) {
        System.exit(0);
    }

    public void showLabel() {
        textLabel.setText("当前糖果数量为: " + gumballMachine.getCount());
    }
}
```

```
package cn.xmu.edu.gumball;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.stage.Stage;

import java.io.IOException;

public class GumballApplication extends Application {
    @Override
    public void start(Stage stage) throws IOException {
        FXMLLoader fxmlLoader = new FXMLLoader(GumballApplication.class.
                                               getResource("gumball-view.fxml"));
        Scene scene = new Scene(fxmlLoader.load(), 480, 240);
        stage.setTitle("Hello!");
        stage.setScene(scene);
        stage.show();
    }

    public static void main(String[] args) {
        launch();
    }
}
```

最后运行结果如下图所示：



```
insertQuarter: You inserted a quarter
turnCrank: You turned...
releaseBall:
A gumball comes rolling out the slot...
```