|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

**О Т Ч Е Т**

по домашнему заданию №\_1\_

**Название**:\_ Автомат-Банкомат \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Дисциплина**:\_Прикладная теория цифровых автоматов\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-42б |  | 26.04.2021 | И.С. Марчук |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Г.С. Иванова |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |

*2021 г.*

**Автомат-Банкомат**

**Задание:**

Создать программу, которая с помощью конечного автомата выполняет работу банкомата.

**Цель работы:**

Изучение получение опыта работы с автоматами на практике.



Рисунок 1 – Схема автомата

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | c | k | b |
| Q1 | Q1 | Q1 | Q1 | Q1 | Q1 | Q1 | Q1 | Q1 | Q1 | Q1 | **Q2** | Q1 | Q1 |
| Q2 | Q2 | **Q3** | **Q4** | **Q5** | **Q6** | **Q7** | Q2 | Q2 | Q2 | Q2 | **Q1** | Q2 | Q2 |
| Q3 | **Q2** | **Q2** | **Q2** | **Q2** | **Q2** | **Q2** | **Q2** | **Q2** | **Q2** | **Q2** | Q3 | **Q2** | **Q2** |
| Q4 | Q4 | Q4 | Q4 | Q4 | Q4 | Q4 | Q4 | Q4 | Q4 | Q4 | Q4 | **Q2** | **Q2** |
| Q5 | Q5 | Q5 | Q5 | Q5 | Q5 | Q5 | Q5 | Q5 | Q5 | Q5 | Q5 | **Q2** | **Q2** |
| Q6 | Q6 | Q6 | Q6 | Q6 | Q6 | Q6 | Q6 | Q6 | Q6 | Q6 | Q6 | **Q2** | **Q2** |
| Q7 | Q7 | Q7 | Q7 | Q7 | Q7 | Q7 | Q7 | Q7 | Q7 | Q7 | Q7 | **Q2** | **Q2** |

Таблица 1 – Таблица переходов цифрового автомата

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | c | k | b |
| Q1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **10** | 0 | 0 |
| Q2 | 0 | **101000** | **101001** | **101010** | **101011** | **101100** | 0 | 0 | 0 | 0 | **1** | 0 | 0 |
| Q3 | **101000** | **101000** | **101000** | **101000** | **101000** | **101000** | **101000** | **101000** | **101000** | **101000** | 0 | **101000** | **101000** |
| Q4 | 101001 | 101001 | 101001 | 101001 | 101001 | 101001 | 101001 | 101001 | 101001 | 101001 | 0 | **101111** | **100000** |
| Q5 | 101010 | 101010 | 101010 | 101010 | 101010 | 101010 | 101010 | 101010 | 101010 | 101010 | 0 | **101111** | **100000** |
| Q6 | 101011 | 101011 | 101011 | 101011 | 101011 | 101011 | 101011 | 101011 | 101011 | 101011 | 0 | **101111** | **100000** |
| Q7 | 101100 | 101100 | 101100 | 101100 | 101100 | 101100 | 101100 | 101100 | 101100 | 101100 | 0 | **101111** | **100000** |

Таблица 2 – Таблица выходов цифрового автомата

**Пояснение работы автомата:**

**Входные сигналы:**

- «с» – Клиент вставил/вытащил карту;

- «0»,«1»..«8»,«9» – Ввод цифр;

- «k» - Ввод;

- «b» – Кнопка назад.

**Состояния:**

«Q1» – Вставьте карту;

«Q2» – Главный экран;

«Q3» – Баланс;

«Q4» – Снять наличные;

«Q5» – Пополнить карту;

«Q6» – Перевести деньги на другую карту;

«Q7» – Оплатить телефон.

**Выходные сигналы:**

«10» – Клиент вошел в аккаунт;

«1» - Клиент вышел;

«0» – Нет доступа к такой операции;

«10 1000» – Клиент запросил баланс;

«10 1001» – Клиент хочет снять наличные;

«10 1010» – Клиент хочет пополнить карту;

«10 1011» – Клиент хочет перевести деньги другому пользователю;

«10 1100» – Клиент хочет оплатить телефон;

«10 0000» – Отмена операции;

«10 1111» – Подтверждение операции.

**Ограничения:**

Карту нельзя вытащить или вставить, если банкомат находится в состояниях Q3-Q7;

Карту нельзя вытащить, когда она еще не вставлена и нельзя вставить, когда она уже вставлена;

Номер карты – 5 цифр;

Номер телефона – 9 цифр;

Пользователь не может снять средств больше, чем у него есть на карте;

Максимальный баланс пользователя 50000 единиц;

**Код программы:**

public class MainActivity extends AppCompatActivity {

// ключи расшифровки ответов банкомата

Map<Integer, String> backTranscript = new HashMap<>();

// данные автомата

final static int[][][] digitalMachine = {

/\*Q1\*/{{1, 0}, {1, 0}, {1, 0}, {1, 0}, {1, 0}, {1, 0},

{1, 0}, {1, 0}, {1, 0}, {1, 0}, {2, 10}, {1, 0}, {1, 0}},

/\*Q2\*/{{2, 0}, {3, 101000}, {4, 101001}, {5, 101010}, {6, 101011}, {7, 101100}, {2, 0}, {2, 0}, {2, 0}, {2, 0}, {1, 1}, {2, 0}, {2, 0}},

/\*Q3\*/{{2, 101000}, {2, 101000}, {2, 101000}, {2, 101000}, {2, 101000}, {2, 101000}, {2, 101000}, {2, 101000}, {2, 101000}, {2, 101000}, {3, 0}, {2, 101000}, {2, 101000}},

/\*Q4\*/{{4, 101001}, {4, 101001}, {4, 101001}, {4, 101001}, {4, 101001}, {4, 101001}, {4, 101001}, {4, 101001}, {4, 101001}, {4, 101001}, {4, 0}, {2, 101111}, {2, 100000}},

/\*Q5\*/{{5, 101010}, {5, 101010}, {5, 101010}, {5, 101010}, {5, 101010}, {5, 101010}, {5, 101010}, {5, 101010}, {5, 101010}, {5, 101010}, {5, 0}, {2, 101111}, {2, 100000}},

/\*Q6\*/{{6, 101011}, {6, 101011}, {6, 101011}, {6, 101011}, {6, 101011}, {6, 101011}, {6, 101011}, {6, 101011}, {6, 101011}, {6, 101011}, {6, 0}, {2, 101111}, {2, 100000}},

/\*Q7\*/{{7, 101100}, {7, 101100}, {7, 101100}, {7, 101100}, {7, 101100}, {7, 101100}, {7, 101100}, {7, 101100}, {7, 101100}, {7, 101100}, {7, 0}, {2, 101111}, {2, 100000}},

};

static int currentState = 1;

static int currentOutput = 0;

// сосотояние счета

static int balance = 1000;

// буфер для ввода пользователя

static int enteredNumber = 0;

// экран банкомата

TextView screen;

ImageView cardView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

// ключи расшифровки ответов банкомата

backTranscript.put(10, "Клиент вошел в аккаунт");

backTranscript.put(1, "Клиент вышел");

backTranscript.put(0, "Нет доступа к такой операции");

backTranscript.put(101000, "Клиент запросил баланс");

backTranscript.put(101001, "Клиент хочет снять наличные");

backTranscript.put(101010, "Клиент хочет пополнить карту");

backTranscript.put(101011, "Клиент хочет перевести деньги");

backTranscript.put(101100, "Клиент хочет оплатить телефон");

backTranscript.put(100000, "Отмена операции");

backTranscript.put(101111, "Подтверждение операции");

// выход автомата

TextView output = findViewById(R.id.main\_otput);

ScrollView outputScroll = findViewById(R.id.main\_otput\_scroll);

// кнопки банкомата в разметке

View[] buttons = new View[13];

buttons[0] = findViewById(R.id.main\_backplate\_button\_0);

buttons[1] = findViewById(R.id.main\_backplate\_button\_1);

buttons[2] = findViewById(R.id.main\_backplate\_button\_2);

buttons[3] = findViewById(R.id.main\_backplate\_button\_3);

buttons[4] = findViewById(R.id.main\_backplate\_button\_4);

buttons[5] = findViewById(R.id.main\_backplate\_button\_5);

buttons[6] = findViewById(R.id.main\_backplate\_button\_6);

buttons[7] = findViewById(R.id.main\_backplate\_button\_7);

buttons[8] = findViewById(R.id.main\_backplate\_button\_8);

buttons[9] = findViewById(R.id.main\_backplate\_button\_9);

buttons[10] = findViewById(R.id.main\_card\_image);// c

buttons[11] = findViewById(R.id.main\_backplate\_button\_ok);// k

buttons[12] = findViewById(R.id.main\_backplate\_button\_back);// b

// экран банкомата

screen = findViewById(R.id.main\_screen);

cardView = (ImageView) buttons[10];

screenOut(currentState);

// обработка нажатий

for (int buttonI = 0; buttonI < buttons.length; buttonI++) {

int finalButtonI = buttonI;

buttons[buttonI].setOnClickListener((view) -> {

// при нажатии на кнопку меняем состояние автомата

int lastState = currentState;

currentOutput = digitalMachine[currentState - 1][finalButtonI][1];// 2-1 / 1

currentState = digitalMachine[currentState - 1][finalButtonI][0];

// выход автомата

output.append("[" + lastState + "->" + currentState + "][" + currentOutput + "]:"

+ backTranscript.get(currentOutput) + '\n');

outputScroll.fullScroll(ScrollView.FOCUS\_DOWN);

// ввод цифр в буфер

if (finalButtonI < 10 && lastState >= 4 && lastState <= 7 && currentState == lastState) {

enteredNumber = enteredNumber \* 10 + finalButtonI;

switch (currentState) {

case 4:// Снять наличные;

if (balance < enteredNumber) enteredNumber = balance;

break;

case 5:// Пополнить карту;

if (balance + enteredNumber > 50000)

enteredNumber = 50000 - balance;

break;

case 6:// Перевести деньги на другую карту;

if (enteredNumber > 99999)

enteredNumber = enteredNumber % 100000;

break;

case 7:// Оплатить телефон.

if (enteredNumber > 999999999)

enteredNumber = enteredNumber % 1000000000;

break;

}

}

// чистка числа

if (currentOutput == 100000) {// отмена

enteredNumber = 0;

} else if (currentOutput == 101111) {// подтверждение

switch (lastState) {

case 4:// Снять наличные;

balance = balance - enteredNumber;

break;

case 5:// Пополнить карту;

balance = balance + enteredNumber;

break;

case 6:// Перевести деньги на другую карту;

case 7:// Оплатить телефон.

balance = balance - 100;

if (balance < 0) balance = 0;

break;

}

enteredNumber = 0;

}

// --------- графика ---------

// вывод картинки карты

if (lastState == 1 && currentState == 2) {// вставили карту

cardView.setImageResource(R.drawable.card\_full);

} else if (lastState == 2 && currentState == 1) {// вытащили карту

cardView.setImageResource(R.drawable.card\_empty);

} else if (currentState > 2) {// в остальных случаях

cardView.setImageResource(R.drawable.card\_gray);

} else if (lastState > 2) {

cardView.setImageResource(R.drawable.card\_full);

}

// вывод информации на экран

screenOut(lastState);

});

}

}

void screenOut(int lastState) {

switch (currentState) {

case 1:// введите карту

screen.setText("Вставьте пожалуйста карту");

break;

case 2:// Главный экран;

screen.setText("Добро пожаловать! Что вы хотите сделать?\n 1 - Узнать баланс\n 2 - Снять наличные\n 3 - Пополнить карту\n 4 - Перевести деньги на другую карту\n 5 - Оплатить телефон\n");

break;

case 3:// Баланс;

screen.setText("Ваш баланс:" + balance + " единиц");

break;

case 4:// Снять наличные;

screen.setText("Введите сумму которую хотите снять(тек. баланс:" + balance + "):\n" + enteredNumber);

break;

case 5:// Пополнить карту;

screen.setText("Введите сумму которую хотите положить:\n" + enteredNumber);

break;

case 6:// Перевести деньги на другую карту;

screen.setText(String.format("Введите номер карты для перевода 100 единиц:\n%05d", enteredNumber));

break;

case 7:// Оплатить телефон.

screen.setText(String.format("Введите номер телефона для перевода 100 единиц:\n%09d", enteredNumber));

break;

}

}

@Override

protected void onDestroy() {

currentState = 1;

super.onDestroy();

}

}



Рисунок 2 – Состояние автомата Q1

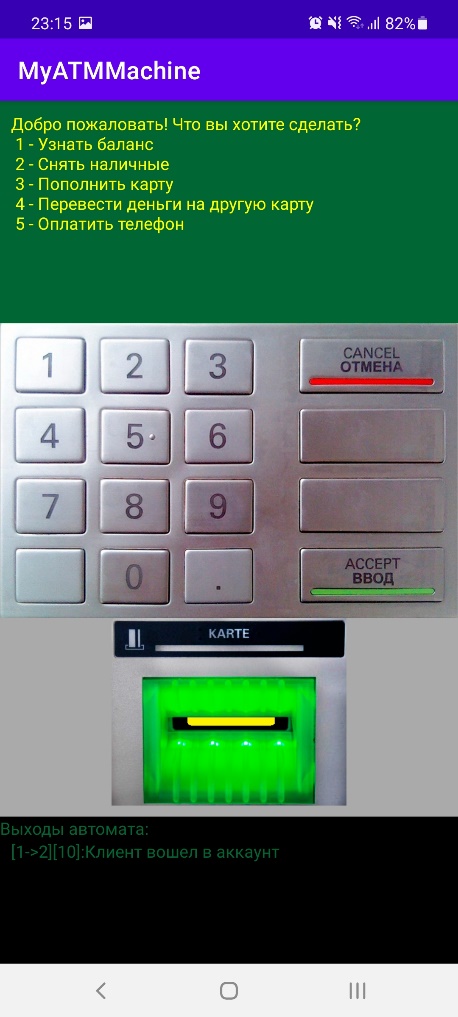


Рисунок 3 – Состояние автомата Q2



Рисунок 4 – Состояние автомата Q3



Рисунок 5 – Состояние автомата Q4



Рисунок 6 – Состояние автомата Q5



Рисунок 7 – Состояние автомата Q6



Рисунок 8 – Состояние автомата Q7

**Вывод:**

Я изучил опытным путем проверил свои знания в теории цифровых автоматов. И выяснил, что автомат для банкомата вполне применим и удовлетворяет всем необходимым условиям эксплуатации банкоматов.