



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 **09.03.03 Прикладная информатика**

**О Т Ч Е Т**

**По домашнему заданию варианта 10**

**Дисциплина:** Прикладная теория цифровых автоматов

**Название:** Автомат «замок»

автомат реализует  
работу  
принимает задание

Студент

ИУ6-ИМ5Б

(Группа)

18/5/2022

(Подпись, дата)

Ле Куанг Ньуе

(И.О. Фамилия)

Преподаватель

ХОР (245)

(Подпись, дата)

Ю.И. Бауман

(И.О. Фамилия)

10/05/2022

Москва, 2022

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
ОСНОВНАЯ ЧАСТЬ.....	4
1 Спецификация автомат .....	4
2 Тестирование.....	6
ЗАКЛЮЧЕНИЕ .....	9
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	10
ПРИЛОЖЕНИЕ .....	11

## ВВЕДЕНИЕ

В данной работе выполнена реализация цифрового автомата – “замок”.

Существуют 2 способа реализации автомата: программный и аппаратный. Программная реализация выполняется на любом языке высокого уровня. Аппаратная реализация – предусматривает построение устройств памяти для запоминания текущего состояния автомата, в роли которых обычно используются триггеры.

В настоящей работе использован программный способ реализации цифрового автомата, так как этот способ подразумевает вариативность реализации, возможность отладки и тестирования в процессе разработки программы. К программам (в отличие от аппаратной реализации цифровых автоматов) можно добавлять новые функции по мере изменения целей, под которые она разрабатывается.

Конечный цифровой автомат эмитирует работу кодового замка экрана смартфона.

Задание (Вариант 10) : разработать и реализовать конечный автомат – “замок”.

Цель работы – закрепить навыки реализации конечных цифровых автоматов. Для реализации поставленной цели необходимо выполнить следующие задачи.

Задачи:

- Изучить задание в соответствие со своим вариантом;
- Описать автомат, соответствующий условию задачи;
- Изучить способы реализации цифровых автоматов;
- Выбрать один из способов реализации автоматов;
- Реализовать описанный цифровой автомат.

• Тестировать

## ОСНОВНАЯ ЧАСТЬ

### 1 Спецификация автомат

Необходимо реализовать конечный цифровой автомат “замок”.

Кодовый замок подразумевает собой возможность открытия экрана смартфона. Чтобы реализовать данный цифровой автомат, требуется реализовать возможность установки пароля, ввода пароля, открытия и закрытия экрана смартфона. Открытие экрана смартфона происходит при вводе пароля идентичного установленному (пользователь не может закрыть экран смартфона без установленного пароля). Если при вводе некорректного пароля, то количество попыток для открытия уменьшается на единицу. Всего имеется три попытки, чтобы ввести корректный пароль, иначе экран смартфона будет заблокирован навсегда. Пароль состоит из пяти цифр, которые вводятся с клавиатуры.

На основе проведенного анализа задания составим конечный цифровой автомат.

#### 1.1 Состояния автомата

- $q_0$  – замок открыт, не установлен пароль;
- $q_1$  – замок закрыт, пароль установлен;
- $q_2$  – замок открыт, пароль установлен;
- $q_3$  – замок закрыт, только осталось 2 попытки для ввода корректного пароля;
- $q_4$  – замок закрыт, только осталось 1 попытки для ввода корректного пароля;
- $q_5$  – замок закрыт, экран смартфона заблокирован навсегда.

#### 1.2 Сигналы

##### 1.2.1 Входные сигналы

- $a$  – корректный пароль для установки;
- $b$  – некорректный пароль для установки;
- $c$  – корректорный введенный пароль для открытия экрана смартфона;
- $d$  – некорректорный введенный пароль для открытия экрана смартфона;
- $e$  – закрытие экрана смартфона.

##### 1.2.2 Выходные сигналы

- 0 - пароль не установлен;
- 1 - экран смартфона открыт;
- 2 - экран смартфона закрыт, только осталось 3 попытки ввода;
- 3 - экран смартфона закрыт, только осталось 2 попытки ввода;
- 4 - экран смартфона закрыт, только осталось 1 попытки ввода;
- 5 – экран смартфона заблокирован.

#### 1.3 Таблица переходов автомата

Составив таблицу, описывающую конечный автомат. Описание конечного автомата представлено на таблице 1.

Таблица 1 - Описание конечного автомата

Состояние	$\delta$					$\lambda$				
	a	b	c	d	e	a	b	c	d	e
<b>q<sub>0</sub></b>	q <sub>1</sub>	q <sub>0</sub>	-	-	-	2	0	-	-	-
<b>q<sub>1</sub></b>	-	-	q <sub>2</sub>	q <sub>3</sub>	-	-	-	1	3	-
<b>q<sub>2</sub></b>	q <sub>1</sub>	q <sub>2</sub>	-	-	q <sub>1</sub>	2	1	-	-	2
<b>q<sub>3</sub></b>	-	-	q <sub>2</sub>	q <sub>4</sub>	-	-	-	1	4	-
<b>q<sub>4</sub></b>	-	-	q <sub>2</sub>	q <sub>5</sub>	-	-	-	1	5	-
<b>q<sub>5</sub></b>	-	-	-	-	-	-	-	-	-	-

#### 1.4 Конечный автомат

Описанный конечный автомат в виде графа переходов. Он представлен на рисунке 1.

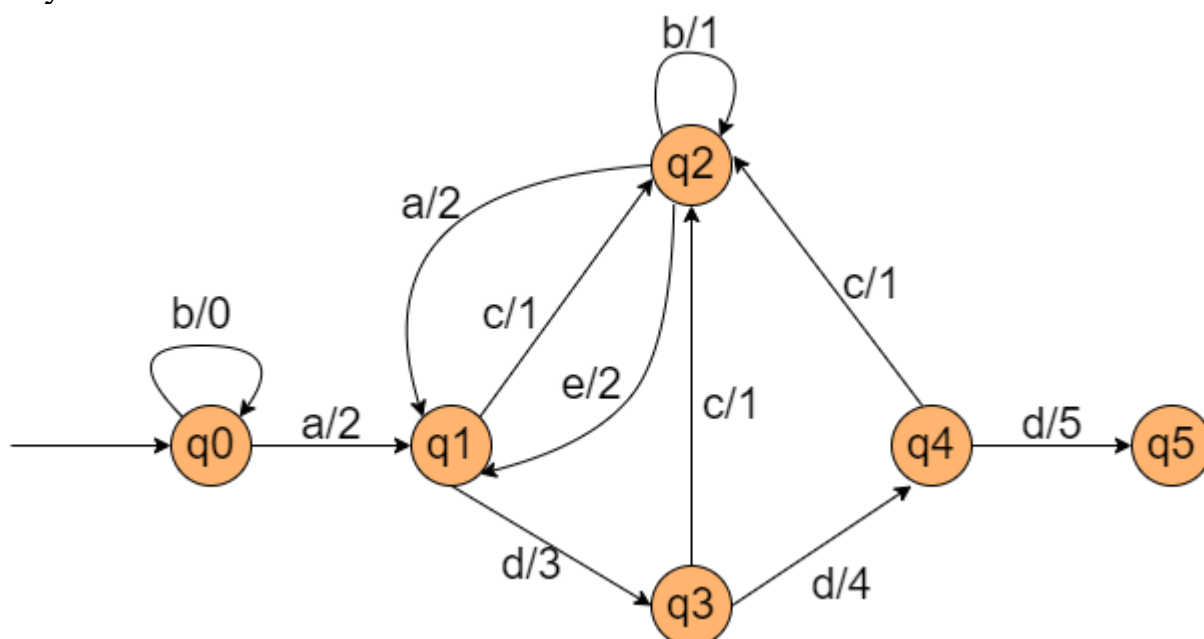


Рисунок 1 – Граф переходов цифрового автомата

#### 1.5 Реализация цифрового автомата «замок»

Для реализации описанного цифрового автомата ~~была~~ разработана схема алгоритма (рисунок 2) и написана программа на языке C++.

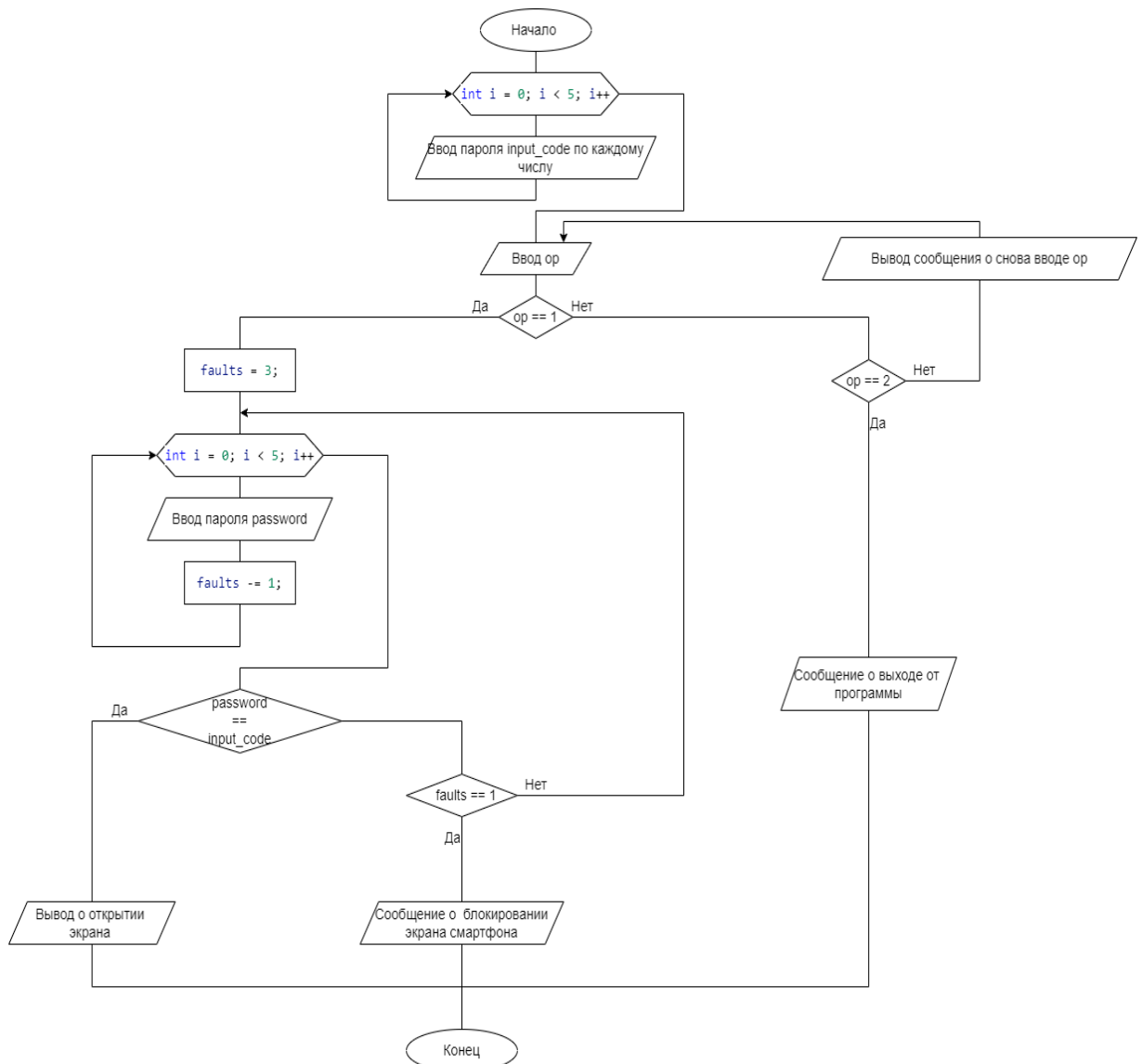


Рисунок 2 – Схема алгоритма

Полный текст программы представлен в приложении.

## 2 Тестирование

Протестируем написанную программу (рисунки 3 - 7).

Для успешного тестирования необходимо проверить все ветви условий, по которым может переходить программа.

1) Введено 5 чисел. У нас будут 2 выбора: ввод пароля для открытия экрана; или выход от программы. При выбора выхода от программы, результат получен на рисунке 3. В случае выбора ввода пароля, у нас будут 3 попытки (рисунок 4);

2) Если в первой попытки, некорректный пароль введен, у нас тоже будут два выбора: продолжаем второй раз вводить пароль (рисунок 4); или выход от программы (рисунок 5);

3) В случае успешного введённого корректного пароля, можно установить новый пароль или закрыть экран смартфона (рисунок 6). В худшем случае пароль будет введен неверно 3 раза, экран будет заблокирован навсегда (Рисунок 7).

```
Smartphone screen is open, set your password
[1]: 1
[2]: 1
[3]: 1
[4]: 1
[5]: 1

Password successfully saved!
Smartphone screen is closed.
1 - Open Smartphone screen
2 - Exit the programm
Input: 2
See you later!
```

Рисунок 3 – Установка пароля и затем выход от программы выбран

```
Password successfully saved!
Smartphone screen is closed.
1 - Open Smartphone screen
2 - Exit the programm
Input: 1
```

```
Enter your password to open the Smartphone screen!
[1]: 1
[2]: 2
[3]: 3
[4]: 4
[5]: 5
```

```
You enter incorrect password! Smartphone screen is closed (you have 2 attempts)
1 - Open Smartphone screen
2 - Exit the programm
Input: █
```

Рисунок 4 – Первый раз ввода некорректного пароля

Enter your password to open the Smartphone screen!

[1]: 1  
[2]: 2  
[3]: 3  
[4]: 4  
[5]: 5

You enter incorrect password! Smartphone screen is closed (you have 2 attemps)

1 - Open Smartphone screen

2 - Exit the programm

Input: 2

See you later!

Рисунок 5 – Выбор выхода от программы после первого раза некорректного пароля

Enter your password to open the Smartphone screen!

[1]: 1  
[2]: 1  
[3]: 1  
[4]: 1  
[5]: 1

You enter correct password! Smartphone screen is opened

1 - Close Smartphone screen

2 - Set password

Input: █

Рисунок 6 – Корректный пароль введен

You enter incorrect password! Smartphone screen is closed (you have 1 attemps)

1 - Open Smartphone screen

2 - Exit the programm

Input: 1

Enter your password to open the Smartphone screen!

[1]: 4  
[2]: 4  
[3]: 4  
[4]: 4  
[5]: 4

Your Smartphone screen is blocked forever!!!!!!!!!!!!!!

PS D:\Прикладная теория цифровых автоматов> █

Рисунок 7 – Экран смартфона заблокирован



## ЗАКЛЮЧЕНИЕ

- 1) При выполнении домашнего задания изучен программный способ реализации конечных цифровых автоматов.
- 2) В ходе выполнения данного домашнего задания спроектирован и реализован конечный автомат “Замок”. Создана программная реализация автомата в среде Visual Studio Code на языке C++.
- 3) Закреплены навыки подготовки и оформления отчета по проделанной работе с учетом требований ГОСТ 7.32

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) Брауэр В. Введение в теорию конечных автоматов; пер. с нем. – М.: Радио и связь, 1987.– 392 с.
- 2) Савельев А.Я. Прикладная теория цифровых автоматов: Учебник для вузов. – М.:Высшая школа, 1987. – 272 с.
- 3) Гилл А. Введение в теорию конечных автоматов; пер. с англ. – М.: Наука, 1966. – 272 с.
- 4) ГОСТ 7.32-2017. СИБИБД. Отчет о научно-исследовательской работе. Структура и правила оформления = System of standards on information, librarianship and publishing. The research report. Structure and rules of presentation : Национальный стандарт РФ : Введ. 01.07.2018. - М.Стандартинформ, 2017. - [32 л.]. - URL: <https://docs.cntd.ru/document/1200157208> (дата обращения: 09.05.2022). - Текст: электронный.
- 5) C++ Structures (struct), [Электронный ресурс], - URL: [https://www.w3schools.com/cpp/cpp\\_structs.asp#:~:text=Structures%20\(also%20called%20structs\)%20are,%2C%20bool%2C%20etc.\)](https://www.w3schools.com/cpp/cpp_structs.asp#:~:text=Structures%20(also%20called%20structs)%20are,%2C%20bool%2C%20etc.)). (Дата обращения: 11.05.2022).

## ПРИЛОЖЕНИЕ

К домашнему заданию прилагается код в консоле C++.

```
#include <bits/stdc++.h>

using namespace std;

struct Safe
{
    bool is_correct;
    string password;
    void init()
    {
        cout << "\nSmartphone screen is open, set your password\n";
        is_correct = 1;
        password = "";
    }
    void setpassword()
    {
        string input_code = "";
        for(int i = 0; i < 5; i++)
        {
            string digit;
            cout << "[" << i + 1 << "]: ";
            cin >> digit;
            while (digit.size() > 1 || !isdigit(digit[0]))
            {
                cout << "You can use only simple digit\n";
                cout << "[" << i + 1 << "]: ";
                cin >> digit;
            }
            input_code += digit;
        }
        password = input_code;
        cout << "\nPassword successfully saved!\n";
        cout << "Smartphone screen is closed.\n";
    }
}

bool open_safe()
{
    cout << "\nEnter your password to open the Smartphone screen!\n";
    string input_code = "";
    for(int i = 0; i < 5; i++)
    {
        string digit;
        cout << "[" << i + 1 << "]: ";
        cin >> digit;
        while (digit.size() > 1 || !isdigit(digit[0]))
        {
```

```

        cout << "You can use only simple digit\n";
        cout << "[" << i + 1 << "]: ";
        cin >> digit;
    }
    input_code += digit;
}
if (password == input_code)
    is_correct = 1;
else is_correct = 0;
return is_correct;
}

void close_safe()
{
    cout << "\nYour Smartphone screen is closed!\n";
}

void block_safe()
{
    cout << "\nYour Smartphone screen is blocked forever!!!!!!!!!!!!!!";
}
}LeSafe;

int main()
{
    LeSafe.init();
    LeSafe.setpassword();
    int faults = 3;
    while(1)
    {
        cout << "1 - Open Smartphone screen\n";
        cout << "2 - Exit the programm\n";
        int op;
        cout << "Input: ";
        cin >> op;
        if (op == 1)
        {
            if (LeSafe.open_safe())
            {
                faults = 3;
                cout << "\nYou enter correct password! Smartphone screen is
opened\n";

                cout << "1 - Close Smartphone screen\n";
                cout << "2 - Set password\n";
                cout << "Input: ";
                int num;
                cin >> num;
                if (num != 1 && num != 2)
                    cout << "Please, choose from Menu!!!\n";
                else if(num == 1)

```

```

        LeSafe.close_safe();
    else
    {
        cout << "\nSet your new password.\n";
        LeSafe.setpassword();
    }
}
else if(faults == 1)
{
    LeSafe.block_safe();
    break;
}
else
{
    cout << "\nYou enter incorrect password! Smartphone screen is closed
(you have " << faults - 1 << " attempts)\n";
    faults -= 1;
}
}
else if (op == 2)
{
    cout << "See you later!\n";
    break;
}
else
{
    cout << "Enter correct number!\n";
}
}
return 0;
}

```