



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

Замечания

- 1 Частичное не соответствие ГОСТ
- 2 Тестирование без учета состояний автомата
- 3 Нет отражены и не выполнена проверка правила последовательной композиции автоматов
- 4 Отсутствуют ссылки на источники НТИ

О Т Ч Е Т

по домашнему заданию № 1

Название: Последовательное соединение автоматов

Дисциплина: Прикладная теория цифровых автоматов

Реализация ЦА принята  
Содержание - ХОРОШО  
Оформление - ХОРОШО  
Изложение - ХОРОШО  
Общая оценка - ХОРОШО

*16.05.22*

Студент

ИУ6-45Б

(Группа)

*КДМ*

12.05.2022

(Подпись, дата)

К.Д.Метелёв

(И.О. Фамилия)

Преподаватель

*Ю.И.Бауман*

*16.05.22*

(Подпись, дата)

Ю.И.Бауман

(И.О. Фамилия)

*МЗ = 24 балла*  
*16.05.22*  
2022 г.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
<del>ОСНОВНАЯ ЧАСТЬ.....</del>	4
Спецификация автомата “школа”.....	4
Матрица автомата “школа”.....	4
Полученный цифровой автомат “школа”.....	5
Спецификация автомата “родители”.....	5
Матрица автомата “родители”.....	6
Полученный цифровой автомат “родители”.....	6
Схема соединения полученных автоматов.....	6
Реализация цифровых автоматов “школа” и “родители”.....	7
Тестирование программы.....	8
ЗАКЛЮЧЕНИЕ.....	10
<u>ИСТОЧНИКИ И ССЫЛКИ.....</u>	11

## ВВЕДЕНИЕ

В настоящей работе выполнена реализация последовательного соединения цифровых автоматов.

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.

Задание (вариант 2): Последовательное соединение автоматов.

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

Задачи:

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

Тестирование

Как проверить это

Это необходимо

Коснуться 3 7

## ОСНОВНАЯ ЧАСТЬ

Создадим два автомата, в которых выходной сигнал первого будет служить входным сигналом второго. Первый автомат назовём школа, а второй родители. Реализуем оба автомата.

Составим спецификацию для данных автоматов:

### Первый автомат(школа):

Состояния автомата:

Q0 – начальное состояние автомата

Q1 – ученик получил двойку

Q2 – ученик получил пятёрку

Входные сигналы:

a – ученик не готов к уроку

b – ученик готов к уроку

c – некорректные данные

Выходные сигналы:

0 – в дневнике стоит двойка

1 – в дневнике стоит пятёрка

2 – ничего не произошло

### Полученный цифровой автомат школа

Ниже представлена таблица 1, которая описывает конечный автомат, составленный по условию задачи.

Таблица 1 – таблица переходов автомата «школа»

Состояние	$\delta$			$\lambda$		
	a	b	c	a	b	c
Q0	Q1	Q2	Q0	0	1	2
Q1	-	-	-	-	-	-
Q2	-	-	-	-	-	-

На рисунке 1 представлен описанный автомат в виде графа.

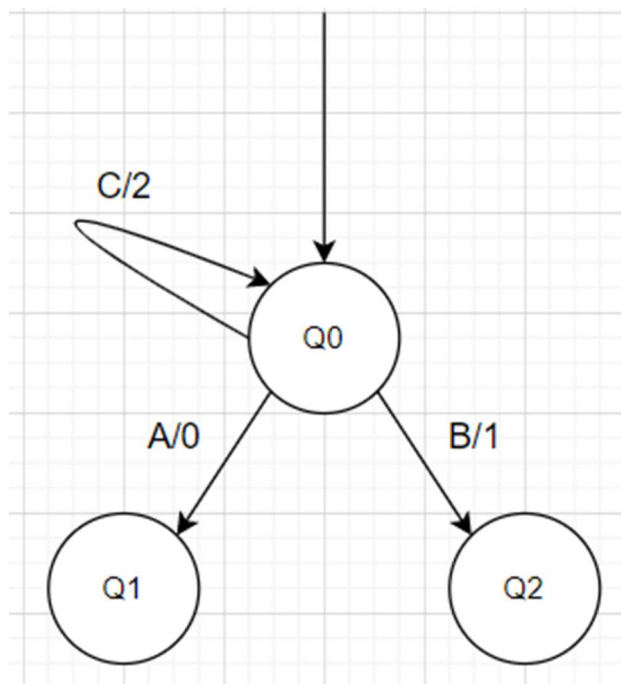


Рисунок 1 – Граф переходов цифрового автомата школа

### **Второй автомат(родители):**

Состояния автомата:

Q3 – начальное состояние автомата

Q4 – родители ругают ученика

Q5 – родители хвалят ученика

Входные сигналы:

e – в дневнике стоит двойка

f – в дневнике стоит пятёрка

g – ничего не произошло

Выходные сигналы:

3 – ученик расстроен

4 – ученик рад

5 – ничего не происходит

### **Полученный цифровой автомат родители**

Ниже представлена таблица 2, которая описывает конечный автомат, составленный по условию задачи.

Таблица 2 – таблица переходов автомата «школа»

Состояние	$\delta$			$\lambda$		
	e	f	g	e	f	g
Q3	Q4	Q5	Q3	3	4	5
Q4	-	-	-	-	-	-
Q5	-	-	-	-	-	-

На рисунке 2 представлен описанный автомат в виде графа.

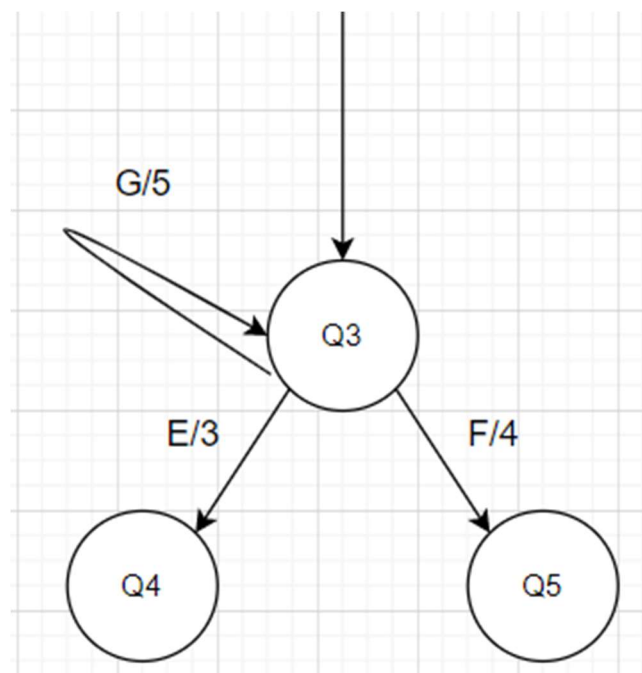


Рисунок 2 – Граф переходов цифрового автомата родители

На рисунке 3 представлена схема соединения описанных автоматов, где  $z_1$  – автомат школа,  $z_2$  – автомат родители,  $A_1$  – входные сигналы автомата школа,  $A_2$  – выходные сигналы автомата школа,  $B_1$  – входные сигналы автомата родители и  $B_2$  – выходные сигналы автомата родители. Так как по заданию автоматы соединены последовательно, то  $A_2 = B_1$ .

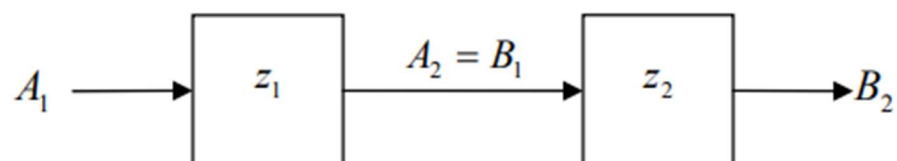


Рисунок 3 – Схема соединения реализованных автоматов

## Реализация цифровых автоматов “школа” и “родители”

Для реализации описанных цифровых автоматов была разработана схема алгоритма, представленная на рисунке 4, и написана программа на языке C++

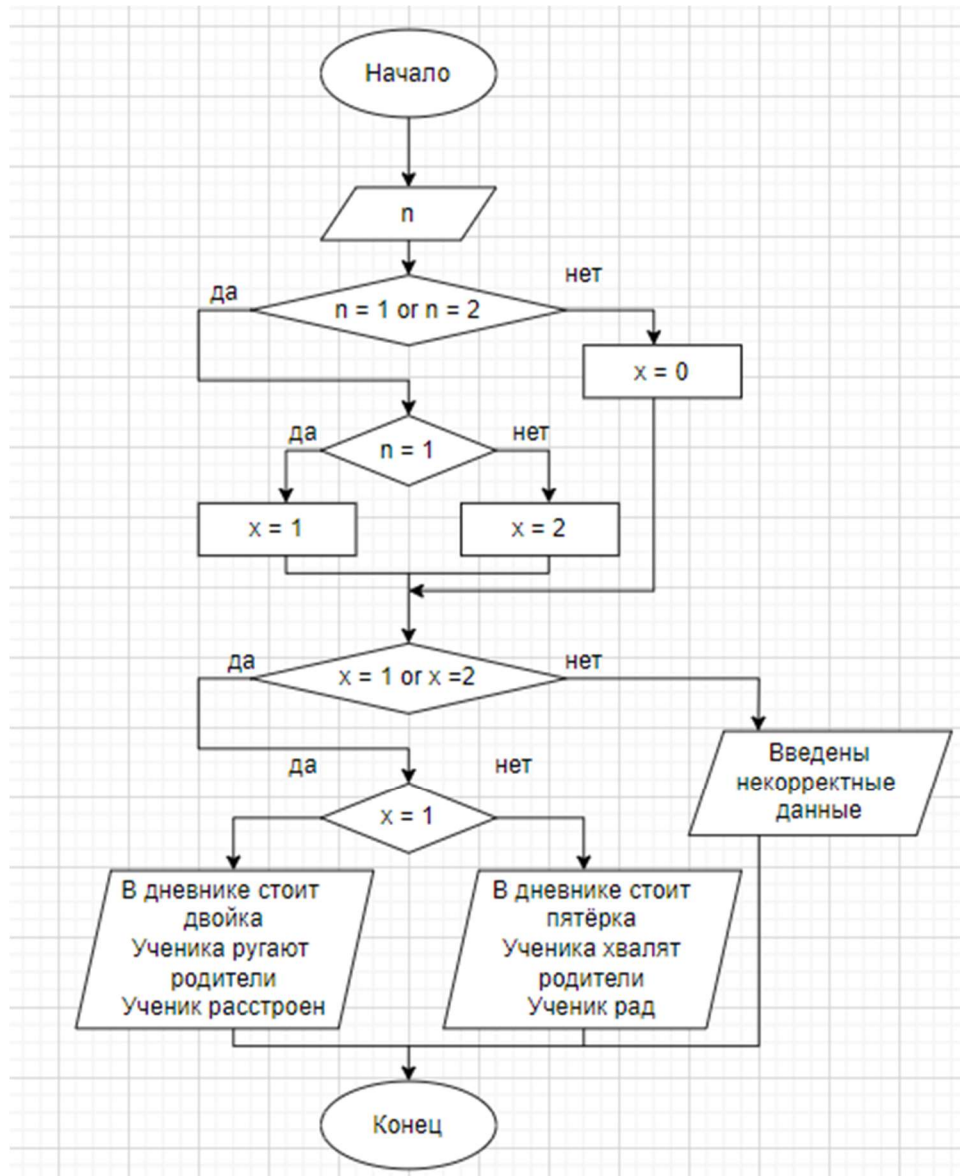


Рисунок 4 – схема алгоритма

Текст программы:

```
#include "iostream"
```

```
using namespace std;
```

```
int main(){
```

```
    int n,x;
```

```
    setlocale(LC_ALL, ".1251");
```

```
    cout << "Введите 1 если ученик не готов к уроку и 2 если ученик готов к уроку"<<endl;
```

```
    cin >> n;
```

```
    if ((n==1) or (n==2)){
```

```
        if (n==1){
```

миссия 7

```

    x = 1;
}
else {
    x = 2;
}
}
else{
    x = 0;
}
if ((x==1) or (x==2)){
    if (x==1){
        cout << "В дневнике стоит двойка\nУченика ругают родители\nУченик расстроен";
    }
    else{
        cout << "В дневнике стоит пятёрка\nУченика хвалят родители\nУченик рад";
    }
}
else{
    cout << "Введены некорректные данные";
}
};

```

## Тестирование программы

Протестируем написанную программу (рисунки 5-7).

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

- 1) Введено число 1 – первый автомат должен выдать, что в дневнике стоит двойка, второй автомат должен выдать, что ученик расстроен. (рисунок 5)
- 2) Введено число 2 – первый автомат должен выдать, что в дневнике стоит пятёрка, второй автомат должен выдать, что ученик рад. (рисунок 6)
- 3) Введено не 2 и не 1 – первый автомат не выводит ничего, второй автомат ничего не выводит, в конце выводится что данные введены некорректно. (рисунок 7)

```

Введите 1 если ученик не готов к уроку и 2 если ученик готов к уроку
1
В дневнике стоит двойка
Ученика ругают родители
Ученик расстроен

```

Рисунок 5 – Введено число 1 – ожидаемый результат вывод “ В дневнике стоит двойка Ученика ругают родители Ученик расстроен”



```
Введите 1 если ученик не готов к уроку и 2 если ученик готов к уроку
2
В дневнике стоит пятёрка
Ученика хвалят родители
Ученик рад
```

Рисунок 6 – Введено число 2 – ожидаемый результат вывод “ В дневнике стоит пятёрка Ученика хвалят родители Ученик рад”

```
Введите 1 если ученик не готов к уроку и 2 если ученик готов к уроку
3
Введены некорректные данные
```

Рисунок 7 – Введено некорректное число – ожидаемый результат вывод “Введены некорректные данные”

## ЗАКЛЮЧЕНИЕ

- 1) При выполнении домашнего задания изучен программный способ реализации конечных цифровых автоматов.
- 2) В ходе выполнения данного домашнего задания спроектированы и реализованы конечные автоматы для ~~демонстрации последовательного~~ соединения автоматов. Создана программная реализация автомата в среде разработки C++
- 3) Закреплены навыки подготовки и оформления отчета по проделанной работе с учетом требований ГОСТ 7.32.

↓  
тестирование  
программы + тесты

## ИСТОЧНИКИ И ССЫЛКИ

2025

- 1) [Теория автоматов К.В.Сафонов](#)
- 2) Конспект лекций Губаря А.М. по дисциплине прикладная теория цифровых автоматов
- 3) Конспект семинарских занятий Баумана Ю.И. по дисциплине прикладная теория цифровых автоматов

В тексте не  
ссылка -