



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
им. 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

Название: «Игра с камешками»

Дисциплина: Прикладная теория цифровых автоматов

Студент

ИУ6-45Б

(Группа)

Преподаватель

Handwritten signature

(Подпись, дата)

М. М. Козлова

(И.О. Фамилия)

Ю. И. Бауман


(Подпись, дата)

(И.О. Фамилия)

Москва, 2022

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	2
ВВЕДЕНИЕ.....	3
Спецификация автомата	4
Полученный цифровой автомат.....	6
Реализация цифрового автомата «Игра с камешками»	8
Листинг программы	9
Тестирование программы	10
ЗАКЛЮЧЕНИЕ	12
ИСПОЛЬЗОВАННАЯ ЛИТЕРАТУРА.....	13



ВВЕДЕНИЕ

В данной работе выполнена реализация цифрового автомата «Игра с камешками».

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 данной 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, ради которых она разрабатывается.

Задание (вариант 17): Два игрока поочередно берут камешки из кучи, в которой сначала 8 (n) камешков. За один ход можно взять не более трёх камешков. Выигрывает тот, кто возьмет последний камень.

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

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

Спецификация автомата

Пусть игрок, который делает ход первым – игрок А, а вторым – игрок В.

1. Состояния автомата

S_0 – начальное состояние автомата

A_6 – когда в куче лежат 6 камней, и следующий ход игрока А

A_5 – когда в куче лежат 5 камней, и следующий ход игрока А

A_4 – когда в куче лежит 4 камня, и следующий ход игрока А

A_3 – когда в куче лежит 3 камня, и следующий ход игрока А

A_2 – когда в куче лежит 2 камня, и следующий ход игрока А

A_1 – когда в куче лежит 1 камень, и следующий ход игрока А

A_0 – когда в куче не осталось камней и выигрывает В

B_7 – когда в куче лежат 7 камней, и следующий ход игрока В

B_6 – когда в куче лежат 6 камней, и следующий ход игрока В

B_5 – когда в куче лежат 5 камней, и следующий ход игрока В

B_4 – когда в куче лежит 4 камня, и следующий ход игрока В

B_3 – когда в куче лежит 3 камня, и следующий ход игрока В

B_2 – когда в куче лежит 2 камня, и следующий ход игрока В

B_1 – когда в куче лежит 1 камень, и следующий ход игрока В

B_0 – когда в куче не осталось камней и выигрывает А

2. Входные сигналы

1 – любой из игроков берет 1 камень

2 – любой из игроков берет 2 камня

3 – любой из игроков берет 3 камня

3. Выходные сигналы

А, 6 – игрок В сделал ход, и в куче осталось 6 камней, следующий ход – А

А, 5 – игрок В сделал ход, и в куче осталось 5 камней, следующий ход – А

А, 4 – игрок В сделал ход, и в куче осталось 4 камня, следующий ход – А

А, 3 – игрок В сделал ход, и в куче осталось 3 камня, следующий ход - А
А, 2 – игрок В сделал ход, и в куче осталось 2 камня, следующий ход - А
А, 1 – игрок В сделал ход, и в куче осталось 1 камень, следующий ход - А
А, 0 – игрок В сделал ход, и в куче осталось 0 камней, победа – В
В, 7 – игрок А сделал ход, и в куче осталось 7 камней, следующий ход – В
В, 6 – игрок А сделал ход, и в куче осталось 6 камней, следующий ход - В
В, 5 – игрок А сделал ход, и в куче осталось 5 камней, следующий ход - В
В, 4 – игрок А сделал ход, и в куче осталось 4 камня, следующий ход - В
В, 3 – игрок А сделал ход, и в куче осталось 3 камня, следующий ход - В
В, 2 – игрок А сделал ход, и в куче осталось 2 камня, следующий ход - В
В, 1 – игрок А сделал ход, и в куче осталось 1 камень, следующий ход - В
В, 0 – игрок А сделал ход, и в куче осталось 0 камней, победа – А

Полученный цифровой автомат

Ниже представлена таблица 1, которая описывает конечный автомат, составленный по условию задачи. Алгоритм работы прост – на любой ход игрока А приходится либо ответный ход игрока В, либо победа одного из них, и наоборот.

Таблица 1 – таблица переходов автомата «Игра с камешками»

Состояние	δ			λ		
	1	2	3	1	2	3
S₀	B ₇	B ₆	B ₅	B, 7	B, 6	B, 5
A₆	B ₅	B ₄	B ₃	B, 5	B, 4	B, 3
A₅	B ₄	B ₃	B ₂	B, 4	B, 3	B, 2
A₄	B ₃	B ₂	B ₁	B, 3	B, 2	B, 1
A₃	B ₂	B ₁	B ₀	B, 2	B, 1	B, 0
A₂	B ₁	B ₀	-	B, 1	B, 0	-
A₁	B ₀	-	-	B, 0	-	-
A₀	-	-	-	-	-	-
B₇	A ₆	A ₅	A ₄	A, 6	A, 5	A, 4
B₆	A ₅	A ₄	A ₃	A, 5	A, 4	A, 3
B₅	A ₄	A ₃	A ₂	A, 4	A, 3	A, 2
B₄	A ₃	A ₂	A ₁	A, 3	A, 2	A, 1
B₃	A ₂	A ₁	A ₀	A, 2	A, 1	A, 0
B₂	A ₁	A ₀	-	A, 1	A, 0	-
B₁	A ₀	-	-	A, 0	-	-
B₀	-	-	-	-	-	-

На рисунке 1 представлен описанный автомат в виде графа

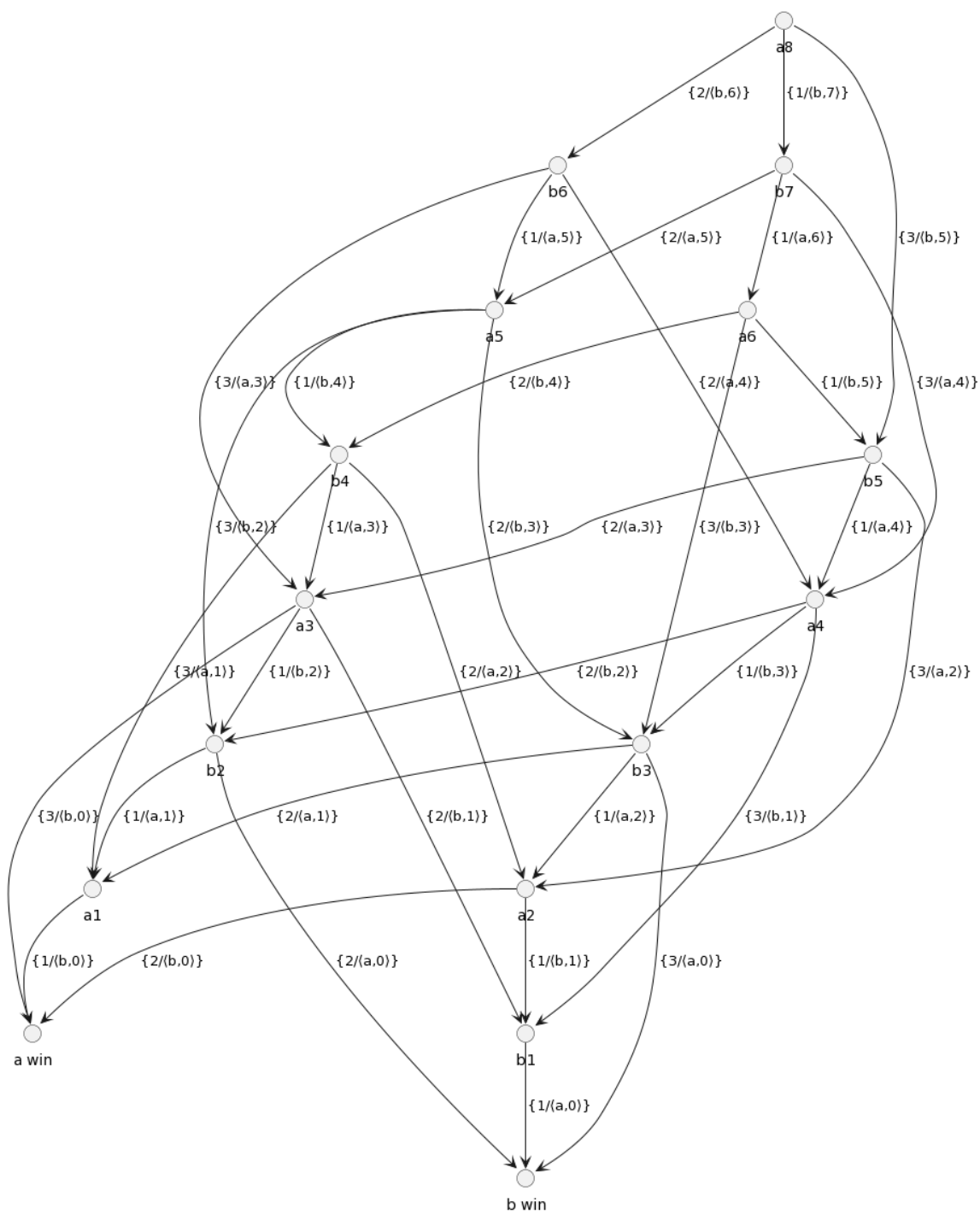


Рисунок 1 – Граф переходов цифрового автомата

Реализация цифрового автомата «Игра с камешками»

Для реализации описанного цифрового автомата ~~была~~ разработана схема алгоритма, представленная на рисунке 2, и написана программа на языке C++.

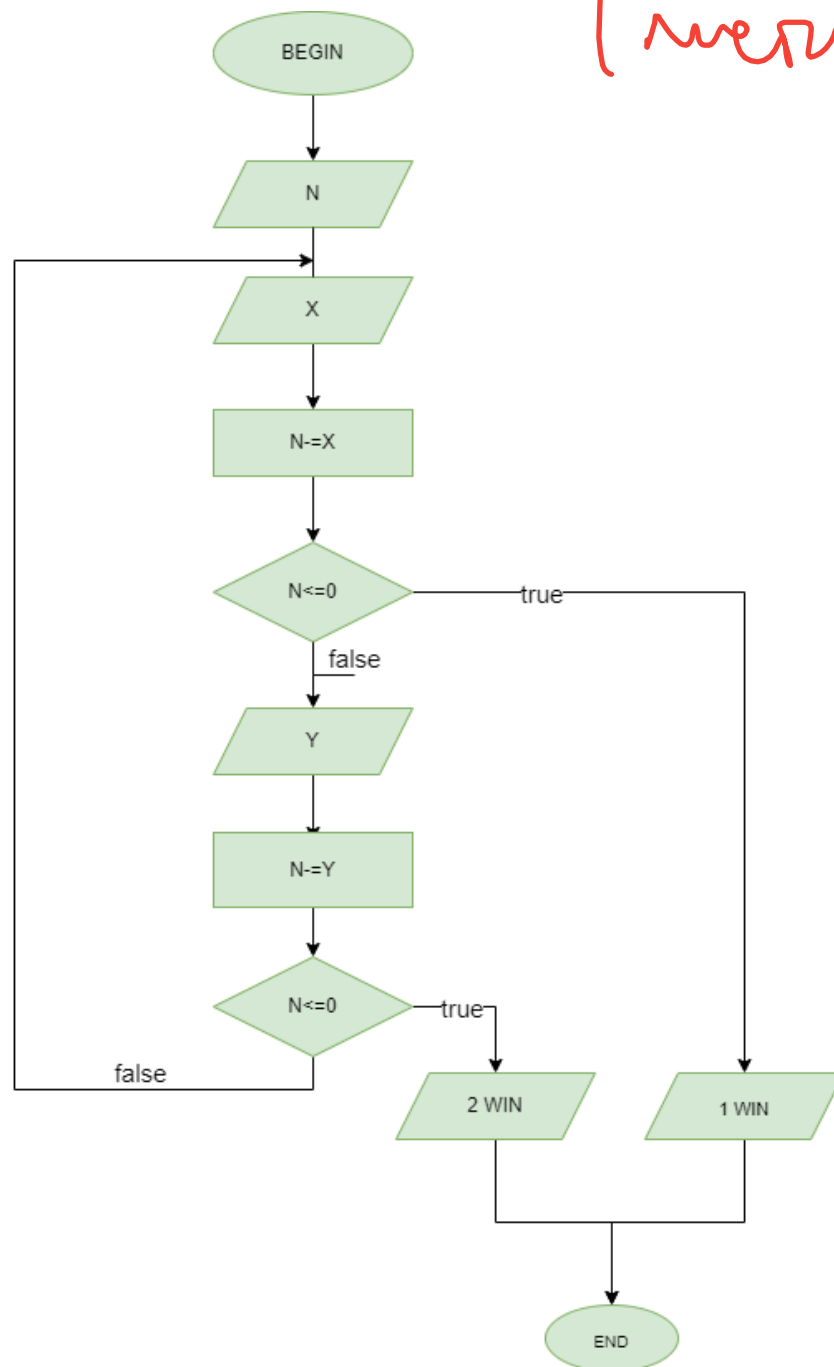


Рисунок 2 – Схема алгоритма

Листинг 1 программы

```
#include <iostream>

using namespace std;

int main()
{
    setlocale(LC_ALL, "Russian");

    int n, x, y;
    cout << "Введите количество камней (n): "; cin >> n;

    while (true) {
        cout << "Игрок 1 возьмёт (1/2/3): "; cin >> x;
        if (x > 3) {
            cout << "Неверное число, введите (1/2/3)" << endl;
            cout << "Игрок 1 возьмёт (1/2/3): "; cin >> x;
        }
        if (n -= x, n <= 0) {
            cout << "Победил игрок 1." << endl; break;
        }

        cout << "Игрок 2 возьмёт (1/2/3): "; cin >> y;
        if (y > 3) {
            cout << "Неверное число, введите (1/2/3)" << endl;
            cout << "Игрок 1 возьмёт (1/2/3): "; cin >> x;
        }
        if (n -= y, n <= 0) {
            cout << "Победил игрок 2." << endl; break;
        }
    }
    return 0;
}
```

Тестирование программы

Для успешного тестирования необходимо проверить все ветви условий, по которым может проходить программа, но так как в нашей задаче таких вариантов множество – проверим основные.

- 1) Представим, что в куче лежит 8 камней, значит вариантов, как игрокам доставать камни – много. Допустим, что первый и второй игроки взяли по 3 камня, следовательно, ожидается, что победа за первым, т. к. ему остаются последние 2 камня.
- 2) Проверим ввод ошибочных значений – как результат будет ожидаться вывод ошибки, и просьба ввести число заново. Далее, проверим работу программы после ввода некорректного значения пользователем: пусть первый игрок возьмет 1 камень, второй – 3, и тогда победа ожидается за первым игроком.
- 3) В конце проверим состояние, что выиграл 2 игрок. Это можно проверить, если при общем количестве камней в куче – 6, первый и второй игроки возьмут по 3 камня, тем самым приведя 2 игрока к победе.

Проведя тестирование получившейся программы, мы убедились, что она работает успешно. Примеры можно увидеть на рисунках 3-5.

```
Консоль отладки Microsoft Visual Studio
Введите количество камней (n): 8
Игрок 1 возьмёт (1/2/3): 3
Игрок 2 возьмёт (1/2/3): 3
Игрок 1 возьмёт (1/2/3): 2
Победил игрок 1.
```

Рисунок 3 – Первый пример работы программы, где побеждает 1-й игрок

На 4 рисунке мы можем видеть, что пользователь сначала вводит ошибочное значение, но программа своевременно его предупреждает, и по итогу удается провести успешную игру.

Рисунок 4 – Второй пример работы программы

Рисунок 5 – Третий пример работы программы, где побеждает 2-й игрок

ЗАКЛЮЧЕНИЕ

- 1) При выполнении домашнего задания был изучен программный способ реализации конечных цифровых автоматов.
- 2) В ходе выполнения данного домашнего задания был спроектирован, реализован и проверен конечный автомат для «Игры в камешки». Программная реализация автомата была выполнена на языке C++.
- 3) Закреплены навыки подготовки и оформления отчета по проделанной работе с учетом требований ГОСТ 7.32. — 2917

ИСПОЛЬЗОВАННАЯ ЛИТЕРАТУРА

1. Савельев А.Я Прикладная теория цифровых автоматов. М. Высшая школа, 1987.
2. Вавилов Е.Н. Портной Г.Д. Синтез схем электронных цифровых машин. М. Советское радио, 1963.
3. Поспелов Д.А. Логические методы анализа и синтеза схем. М. Энергия, 1974