

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

Отчет
по домашней работе

Название: Игра с камешками

Дисциплина: Прикладная теория цифровых автоматов

Студент ИУ6-44Б
(Группа)

(Подпись, дата)

А.М.Гудым
(И.О. Фамилия)

Преподаватель

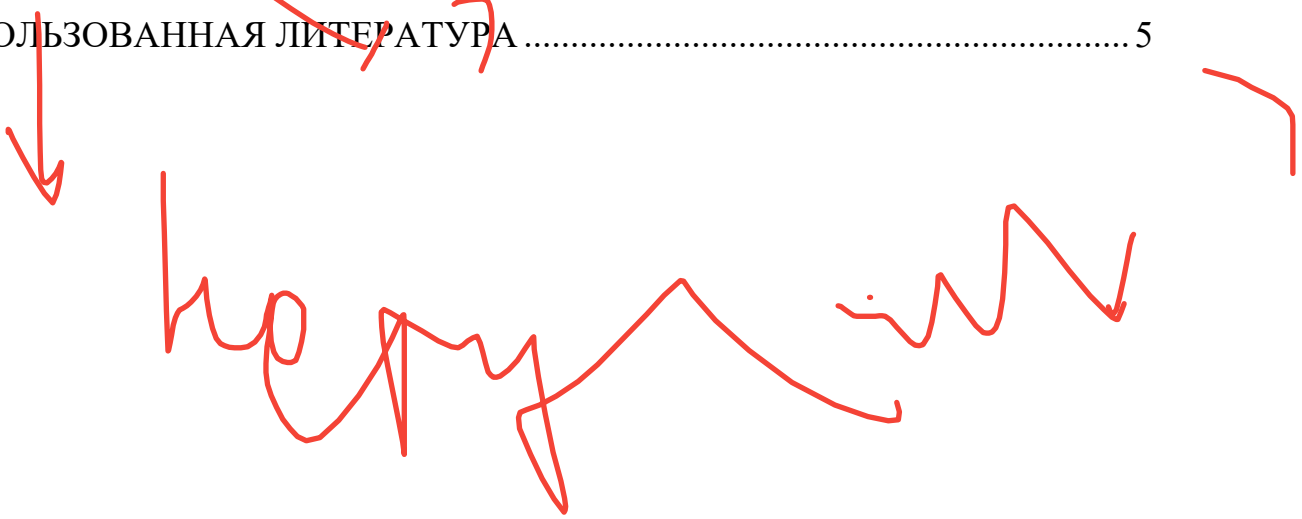
(Подпись, дата)

Ю. И. Бауман
(И.О. Фамилия)

Москва, 2022

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ.....	2
ВВЕДЕНИЕ	3
Спецификация автомата	4
Полученный цифровой автомат	5
Реализация цифрового автомата «Игра с камешками»	5
Листинг программы	5
Тестирование программы	5
ЗАКЛЮЧЕНИЕ	5
ИСПОЛЬЗОВАННАЯ ЛИТЕРАТУРА	5

A large, stylized handwritten signature in red ink is written across the lower half of the page. Two red arrows point downwards from the signature area towards the bottom of the page.

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 (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 соответствие 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

S0 – начальное состояние автомата

A6 – когда в куче лежат 6 камней, и следующий ход игрока А

A5 – когда в куче лежат 5 камней, и следующий ход игрока А

A4 – когда в куче лежит 4 камня, и следующий ход игрока А

A3 – когда в куче лежит 3 камня, и следующий ход игрока А

A2 – когда в куче лежит 2 камня, и следующий ход игрока А

A1 – когда в куче лежит 1 камень, и следующий ход игрока А

A0 – когда в куче не осталось камней и выигрывает В

B7 – когда в куче лежат 7 камней, и следующий ход игрока В

B6 – когда в куче лежат 6 камней, и следующий ход игрока В

B5 – когда в куче лежат 5 камней, и следующий ход игрока В

B4 – когда в куче лежит 4 камня, и следующий ход игрока В

B3 – когда в куче лежит 3 камня, и следующий ход игрока В

B2 – когда в куче лежит 2 камня, и следующий ход игрока В

B1 – когда в куче лежит 1 камень, и следующий ход игрока В

B0 – 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
S_0	B_7	B_6	B_5	$B,7$	$B,6$	$B,5$
A_6	B_5	B_4	B_3	$B,5$	$B,4$	$B,3$
A_5	B_4	B_3	B_2	$B,4$	$B,3$	$B,2$
A_4	B_3	B_2	B_1	$B,3$	$B,2$	$B,1$
A_3	B_2	B_1	B_0	$B,2$	$B,1$	$B,0$
A_2	B_1	B_0	-	$B,1$	$B,0$	-
A_1	B_0	-	-	$B,0$	-	-
A_0	-	-	-	-	-	-
B_7	A_6	A_5	A_4	$A,6$	$A,5$	$A,4$
B_6	A_5	A_4	A_3	$A,5$	$A,4$	$A,3$
B_5	A_4	A_3	A_2	$A,4$	$A,3$	$A,2$
B_4	A_3	A_2	A_1	$A,3$	$A,2$	$A,1$
B_3	A_2	A_1	A_0	$A,2$	$A,1$	$A,0$
B_2	A_1	A_0	-	$A,1$	$A,0$	-
B_1	A_0	-	-	$A,0$	-	-
B_0	-	-	-	-	-	-

Рисунок 1 – Таблица автомата

На рисунке 1 представлен описанный автомат в виде графа

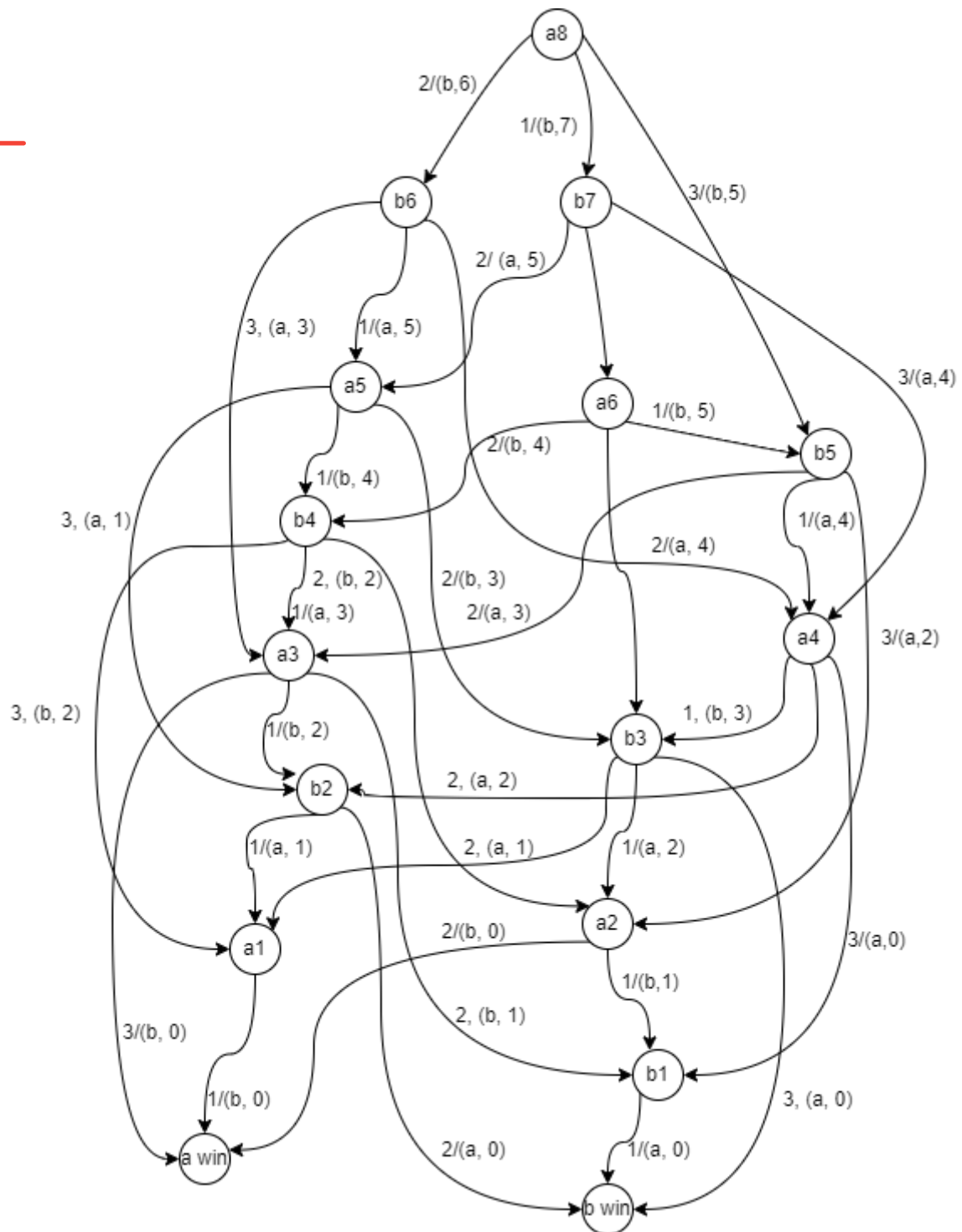


Рисунок 1 – Граф переходов цифрового автомата

РЕАЛИЗАЦИЯ ЦИФРОВОГО АВТОМАТА «ИГРА С КАМЕШКАМИ»

Для реализации описанного цифрового автомата была разработана схема алгоритма, представленная на рисунке 2, и написана программа на языке Python3.

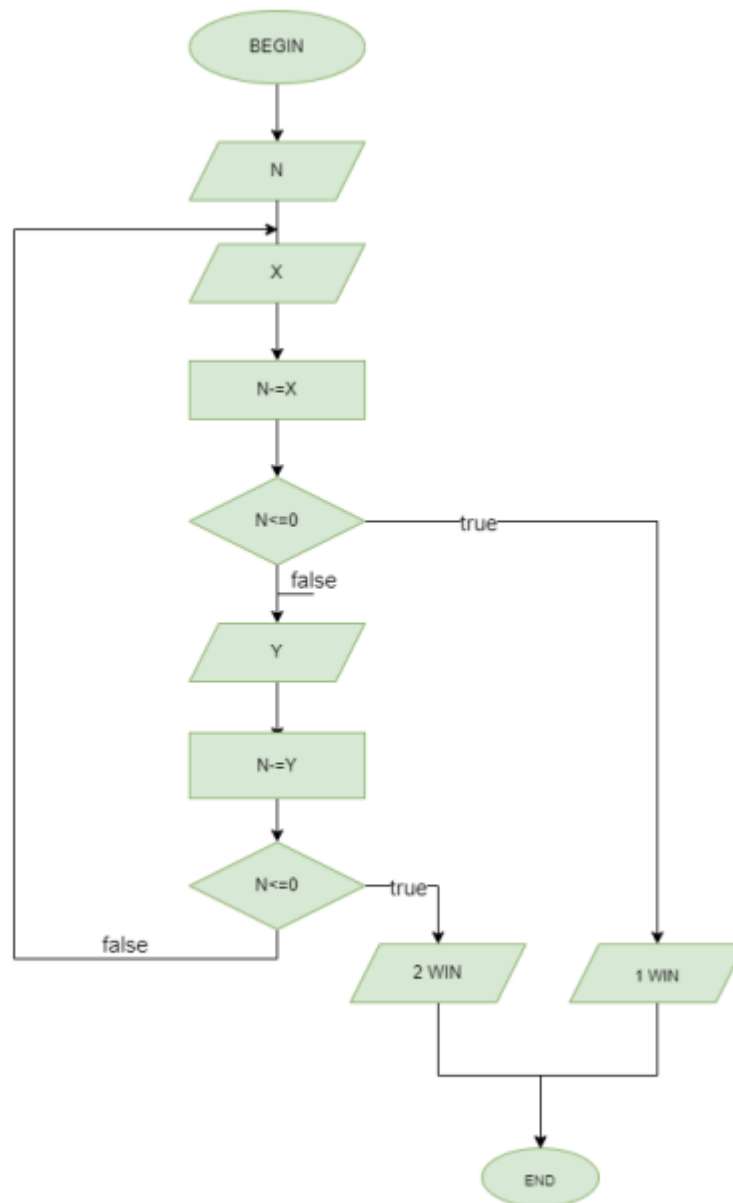


Рисунок 2 – Схема алгоритма

ТЕКСТ ПРОГРАММЫ

```
n = int(input("Укажите кол-во камушков "))
while (n >= 1):
    x = int(input("Игрок 1 возьмёт (1/2/3) "))
    while (x > 3 or x < 1):
        print("Неверный формат, допустимы числа от 1 до 3")
        x = int(input("Игрок 1 возьмёт (1/2/3) "))
    f = n - x
    if (f <= 0):
        print ("Победил первый игрок!")
    n = f
    if (n > 0):
        x = int(input("Игрок 2 возьмёт (1/2/3) "))
        while (x > 3 or x < 1):
            print("Неверный формат, допустимы числа от 1 до 3")
            x = int(input("Игрок 2 возьмёт (1/2/3) "))
        f = n - x
        if (f <= 0):
            print ("Победил второй игрок!")
        n = f
```

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.

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

```
→ avtomaty python3 igra_kamushki.py
Укажите кол-во камушков 8
Игрок 1 возьмёт (1/2/3) 3
Игрок 2 возьмёт (1/2/3) 3
Игрок 1 возьмёт (1/2/3) 2
Победил первый игрок!
```

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

```
→ avtomaty python3 igra_kamushki.py
Укажите кол-во камушков 5
Игрок 1 возьмёт (1/2/3) 4
Неверный формат, допустимы числа от 1 до 3
Игрок 1 возьмёт (1/2/3) 1
Игрок 2 возьмёт (1/2/3) 4
Неверный формат, допустимы числа от 1 до 3
Игрок 2 возьмёт (1/2/3) 3
Игрок 1 возьмёт (1/2/3) 1
Победил первый игрок!
```

Рисунок 4 – Второй пример работы программы

```
→ avtomaty python3 igra_kamushki.py
Укажите кол-во камушков 6
Игрок 1 возьмёт (1/2/3) 3
Игрок 2 возьмёт (1/2/3) 3
Победил второй игрок!
```

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 Python3.
- 3) Закреплены навыки подготовки и оформления отчета по проделанной работе с учетом требований ГОСТ 7.32.

- 2017