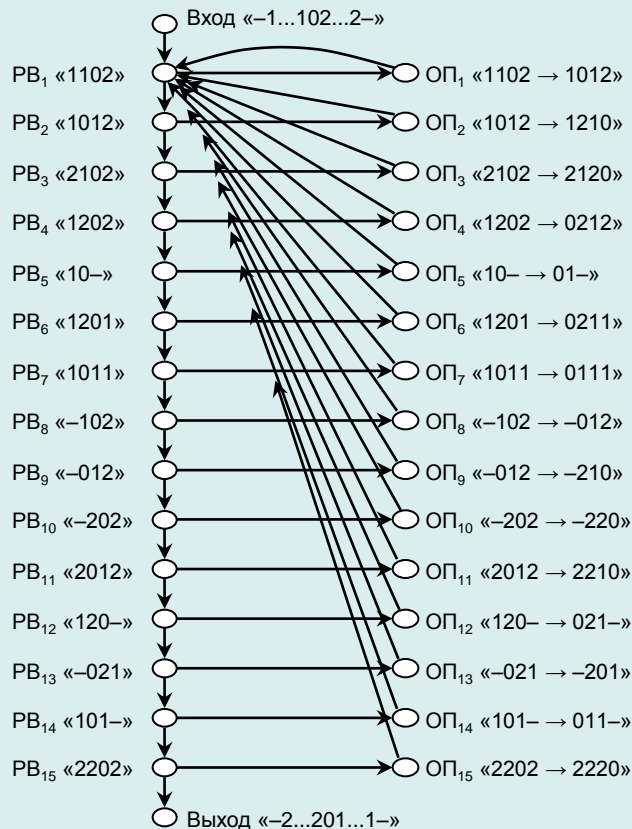


Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
ТАМБОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ



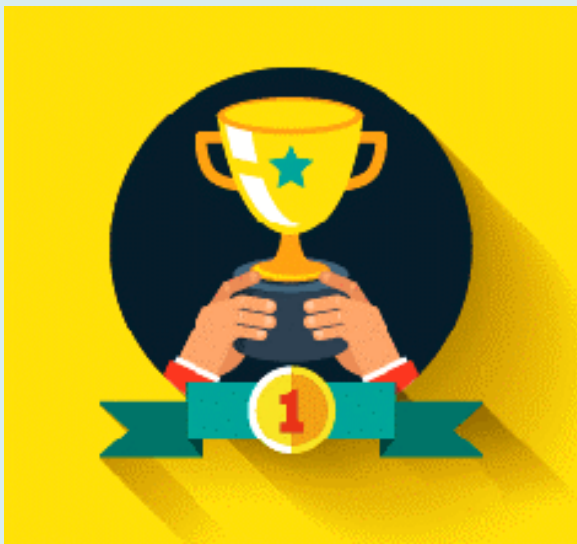
**Нормальный алгоритм Маркова для решения
олимпиадной задачи по информатике «Лягушачьи игры»**



*Автор: Кулаков Юрий Владимирович,
кандидат технических наук, доцент*

Тамбов 2023

Актуальность работы



Насущная необходимость накопления теоретического и практического опыта разработки алгоритмов решения

олимпиадных задач по информатике, а также пропаганда научных знаний и подготовка учащихся к всероссийским и международным олимпиадам по информатике и программированию.

Цель и задачи исследования



Цель исследования: Доказательство алгоритмической разрешимости проблемы достижения требуемой конечной рассадки лягушек в олимпиадной задаче по информатике и программированию «Лягушачьи игры».

Задачи исследования:

- 1) Изучить постановку олимпиадной задачи.
- 2) Исследовать вопрос доказательства алгоритмической разрешимости проблем.
- 3) Сконструировать нормальный алгоритм Маркова для решения олимпиадной задачи «Лягушачьи игры».
- 4) Написать программу на языке Си, реализующую разработанный алгоритм.
- 5) Доказать алгоритмическую разрешимость проблемы достижения требуемой конечной рассадки лягушек в олимпиадной задаче «Лягушачьи игры» путем реализации работы нормального алгоритма Маркова, решающего эту задачу.

Объект, предмет, гипотеза и практическая значимость исследования



Объект исследования: Олимпиадная задача по информатике и программированию «Лягушачьи игры».

Предмет исследования: Возможность достижения требуемой конечной рассадки лягушек в олимпиадной задаче «Лягушачьи игры».

Гипотеза исследования: Проблема достижения требуемой конечной рассадки лягушек в олимпиадной задаче «Лягушачьи игры» алгоритмически разрешима.

Практическая значимость: Приобретение знаний, умений и навыков решения олимпиадных задач по информатике и программированию.

Постановка олимпиадной задачи



(начало)

Всего в тридесятом царстве живет N зеленых и M коричневых лягушек.

Для своей игры они выбирают на болоте $N + M + 1$ кочку, на первые N кочек слева садятся зеленые лягушки, а на последние M – коричневые (т. е. между ними находится одна кочка, на которой никто не сидит).

Зеленые лягушки садятся лицом к коричневым лягушкам, а коричневые – к зеленым.

Лягушки могут двигаться только вперед и не могут разворачиваться.

Постановка олимпиадной задачи (окончание)

На каждом ходе игры одна из лягушек перепрыгивает с той кочки, где она сидит, на свободную кочку. При этом лягушка может прыгнуть на соседнюю кочку вперед, либо перепрыгнуть через одну, если соседняя занята.

Чтобы праздник удался, **зеленые лягушки** должны оказаться на последних кочках, а **коричневые** – на первых.



Напишите программу, которая поможет лягушкам составить план прыжков.

Если же достичь требуемой рассадки лягушек нельзя, выведите одно число минус 1.

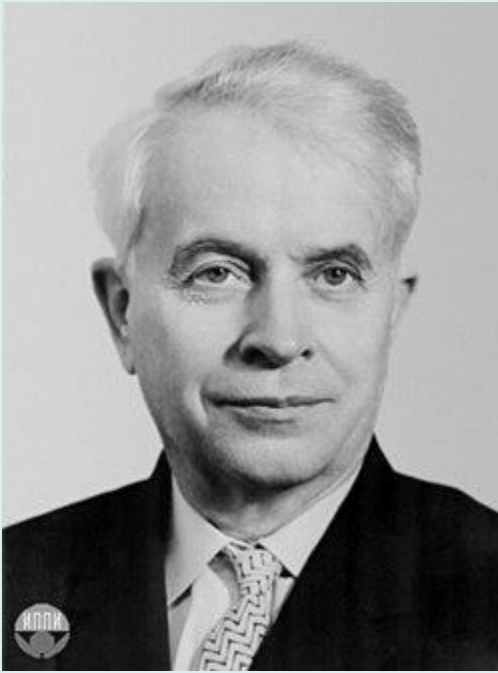
Игра для случая $N = M = 3$



ИНТУИТИВНОЕ (НАИВНОЕ) ОПРЕДЕЛЕНИЕ АЛГОРИТМА



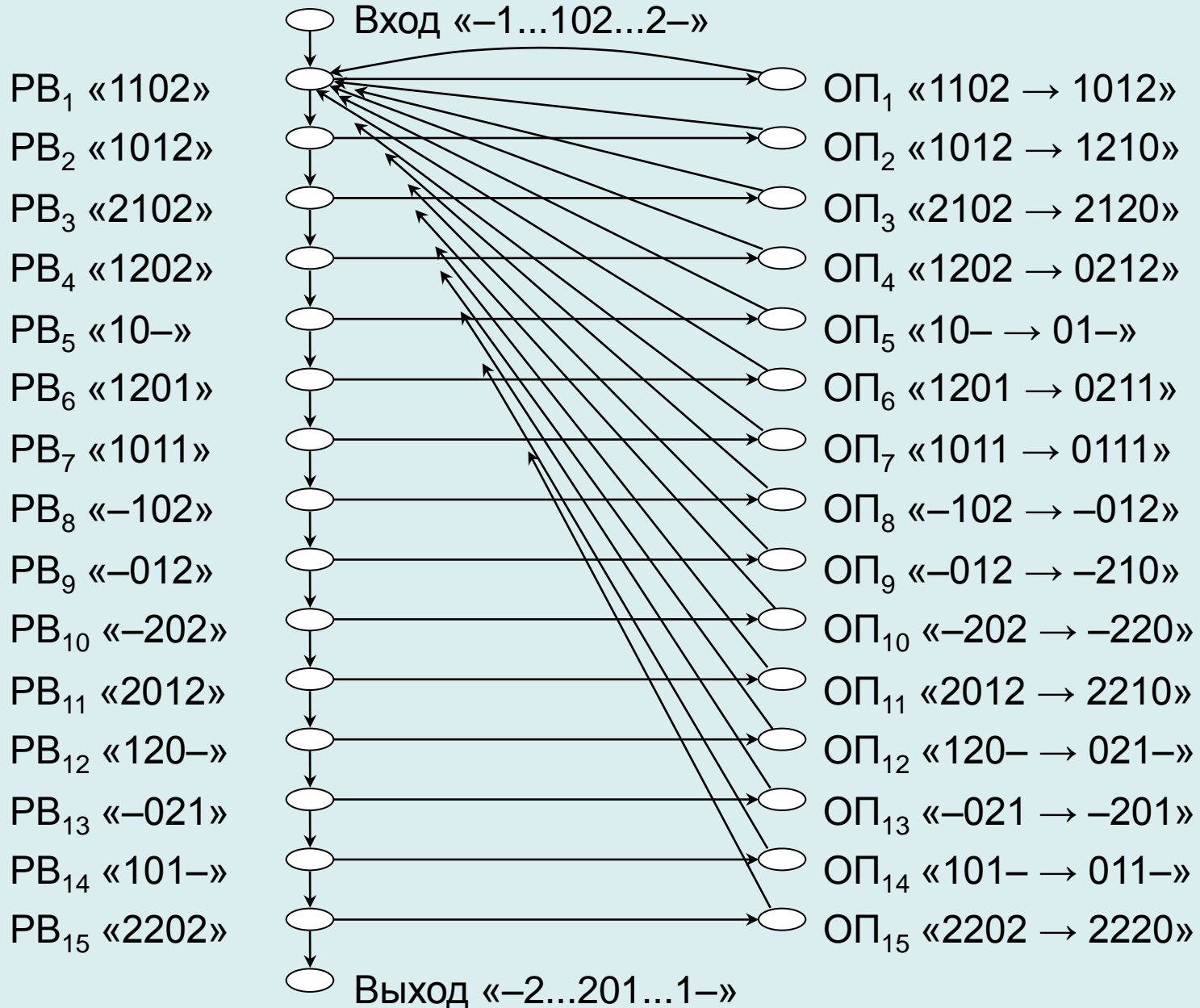
ПЕРЕФРАЗИРОВАННЫЙ ТЕЗИС ТЮРИНГА



*Для любого алгоритма,
понимаемого в интуитивном
смысле, можно построить
нормальный алгоритм Маркова,
функционирование которого
эквивалентно этому алгоритму.
(конец 1940-х годов)*

Некоторая проблема является *алгоритмически разрешимой*, если существует алгоритм (нормальный алгоритм Маркова) для ее решения, и *алгоритмически неразрешимой* – в противном случае.

ГРАФ-СХЕМА НОРМАЛЬНОГО АЛГОРИТМА МАРКОВА



ПЕРВЫЙ ПРЫЖОК (ЗЕЛЁНОЙ ЛЯГУШКИ НА СВОБОДНУЮ КОЧКУ)

Исходная рассадка лягушек



преобразуется в первую рассадку



PV_1 «1102»

1) «-1110222-» (OP_1 «1102 → 1012») «-1101222-»

ВТОРОЙ ПРЫЖОК (КОРИЧНЕВОЙ ЛЯГУШКИ ЧЕРЕЗ ЗЕЛЁНУЮ)

Первая рассадка лягушек



преобразуется во вторую рассадку



$PВ_1$ «1102», $PВ_2$ «1012»

2) «-1101222-» ($ОП_2$ «1012 → 1210») «-1121022-»

ОСТАЛЬНЫЕ ПРЫЖКИ ЛЯГУШЕК

- 3) «-11**21022**-» (ОП₃ «**2102** → **2120**») «-11**21202**-»
- 4) «-11**21202**-» (ОП₄ «**1202** → **0212**») «-11**20212**-»
- 5) «-1**120212**-» (ОП₄ «**1202** → **0212**») «-1**021212**-»
- 6) «-**1021212**-» (ОП₈ «**-102** → **-012**») «-**0121212**-»
- 7) «-**0121212**-» (ОП₉ «**-012** → **-210**») «-**2101212**-»
- 8) «-**2101212**-» (ОП₂ «**1012** → **1210**») «-**2121012**-»
- 9) «-**2121012**-» (ОП₂ «**1012** → **1210**») «-**2121210**-»
- 10) «-**2121210**-» (ОП₅ «**10-** → **01-**») «-**2121201**-»
- 11) «-**2121201**-» (ОП₆ «**1201** → **0211**») «-**2120211**-»
- 12) «-**2120211**-» (ОП₄ «**1202** → **0212**») «-**2021211**-»
- 13) «-**2021211**-» (ОП₁₀ «**-202** → **-220**») «-**2201211**-»
- 14) «-**2201211**-» (ОП₁₁ «**2012** → **2210**») «-**2221011**-»
- 15) «-**2221011**-» (ОП₇ «**1011** → **0111**») «-**2220111**-»

ОПЕРАТОРЫ ПОДСТАНОВКИ, ИСПОЛЬЗУЕМЫЕ ПРИ РЕШЕНИИ ЗАДАЧИ РАЗЛИЧНОЙ РАЗМЕРНОСТИ

Оператор подстановки	Размерность задачи								
	1×1	1×2	1×3	2×1	2×2	2×3	3×1	3×2	3×3
ОП ₁	0	0	0	1	1	1	1	1	1
ОП ₂	0	0	0	1	1	1	1	1	1
ОП ₃	0	1	1	0	1	1	0	1	1
ОП ₄	0	0	1	0	1	1	0	1	1
ОП ₅	1	0	1	1	0	1	1	0	1
ОП ₆	0	0	0	1	0	1	1	1	1
ОП ₇	0	0	0	0	0	0	1	0	1
ОП ₈	1	1	1	0	0	0	1	1	1
ОП ₉	1	1	1	0	0	0	1	1	1
ОП ₁₀	0	1	1	0	0	0	0	1	1
ОП ₁₁	0	0	1	0	1	1	0	0	1
ОП ₁₂	0	1	0	0	1	0	0	1	0
ОП ₁₃	0	0	0	1	1	1	0	0	0
ОП ₁₄	0	0	0	0	1	0	0	1	0
ОП ₁₅	0	0	0	0	0	1	0	0	0

СКРИНШОТЫ РАБОТЫ ПРОГРАММЫ, РЕАЛИЗУЮЩЕЙ НОРМАЛЬНЫЙ АЛГОРИТМ МАРКОВА (НАЧАЛО)

```
Borland C++ for DOS
Frogs starting position: -102-
The solution of the problem:
-102- <OP8 "-102=>-012">
-012- <OP9 "-012=>-210">
-210- <OP5 "10-=>01-">
-201-
Press any key ... _
```

```
Borland C++ for DOS
Frogs starting position: -1102-
The solution of the problem:
-1102- <OP1 "1102=>1012">
-1012- <OP2 "1012=>1210">
-1210- <OP5 "10-=>01-">
-1201- <OP6 "1201=>0211">
-0211- <OP13 "-021=>-201">
-2011-
Press any key ... _
```

```
Borland C++ for DOS
Frogs starting position: -1022-
The solution of the problem:
-1022- <OP8 "-102=>-012">
-0122- <OP9 "-012=>-210">
-2102- <OP3 "2102=>2120">
-2120- <OP12 "120-=>021-">
-2021- <OP10 "-202=>-220">
-2201-
Press any key ...
```

```
Borland C++ for DOS
Frogs starting position: -11022-
The solution of the problem:
-11022- <OP1 "1102=>1012">
-10122- <OP2 "1012=>1210">
-12102- <OP3 "2102=>2120">
-12120- <OP12 "120-=>021-">
-12021- <OP4 "1202=>0212">
-02121- <OP13 "-021=>-201">
-20121- <OP11 "2012=>2210">
-22101- <OP14 "101-=>011-">
-22011-
Press any key ...
```

```
Borland C++ for DOS
Frogs starting position: -10222-
The solution of the problem:
-10222- <OP8 "-102=>-012">
-01222- <OP9 "-012=>-210">
-21022- <OP3 "2102=>2120">
-21202- <OP4 "1202=>0212">
-20212- <OP10 "-202=>-220">
-22012- <OP11 "2012=>2210">
-22210- <OP5 "10-=>01-">
-22201-
Press any key ...
```

СКРИНШОТЫ РАБОТЫ ПРОГРАММЫ, РЕАЛИЗУЮЩЕЙ НОРМАЛЬНЫЙ АЛГОРИТМ МАРКОВА (ОКОНЧАНИЕ)

```
Borland C++ for DOS
Frogs starting position: -110222-
The solution of the problem:
-110222- <OP1 "1102=>1012">
-101222- <OP2 "1012=>1210">
-121022- <OP3 "2102=>2120">
-121202- <OP4 "1202=>0212">
-120212- <OP4 "1202=>0212">
-021212- <OP13 "-021=>-201">
-201212- <OP11 "2012=>2210">
-221012- <OP2 "1012=>1210">
-221210- <OP5 "10-=>01-">
-221201- <OP6 "1201=>0211">
-220211- <OP15 "2202=>2220">
-222011-
Press any key ...
```

```
Borland C++ for DOS
Frogs starting position: -111022-
The solution of the problem:
-111022- <OP1 "1102=>1012">
-110122- <OP2 "1012=>1210">
-112102- <OP3 "2102=>2120">
-112120- <OP12 "120-=>021-">
-112021- <OP4 "1202=>0212">
-102121- <OP8 "-102=>-012">
-012121- <OP9 "-012=>-210">
-210121- <OP2 "1012=>1210">
-212101- <OP14 "101-=>011-">
-212011- <OP6 "1201=>0211">
-202111- <OP10 "-202=>-220">
-220111-
Press any key ...
```

```
Borland C++ for DOS
Frogs starting position: -11102-
The solution of the problem:
-11102- <OP1 "1102=>1012">
-11012- <OP2 "1012=>1210">
-11210- <OP5 "10-=>01-">
-11201- <OP6 "1201=>0211">
-10211- <OP8 "-102=>-012">
-01211- <OP9 "-012=>-210">
-21011- <OP7 "1011=>0111">
-20111-
Press any key ...
```

```
Borland C++ for DOS
Frogs starting position: -1110222-
The solution of the problem:
-1110222- <OP1 "1102=>1012">
-1101222- <OP2 "1012=>1210">
-1121022- <OP3 "2102=>2120">
-1121202- <OP4 "1202=>0212">
-1120212- <OP4 "1202=>0212">
-1021212- <OP8 "-102=>-012">
-0121212- <OP9 "-012=>-210">
-2101212- <OP2 "1012=>1210">
-2121012- <OP2 "1012=>1210">
-2121210- <OP5 "10-=>01-">
-2121201- <OP6 "1201=>0211">
-2120211- <OP4 "1202=>0212">
-2021211- <OP10 "-202=>-220">
-2201211- <OP11 "2012=>2210">
-2221011- <OP7 "1011=>0111">
-2220111-
Press any key ...
```


ВЫВОДЫ



- 1) Изучена постановка олимпиадной задачи «Лягушачьи игры».
- 2) Исследован вопрос алгоритмической разрешимости проблем.

- 3) Сконструирован нормальный алгоритм Маркова для решения олимпиадной задачи «Лягушачьи игры».
- 4) Написана программа на языке Си, реализующая разработанный алгоритм.
- 5) Доказана алгоритмическая разрешимость проблемы достижения требуемой конечной рассадки лягушек в олимпиадной задаче «Лягушачьи игры» путем реализации работы нормального алгоритма Маркова, решающего эту задачу.

СПАСИБО ЗА ВНИМАНИЕ!