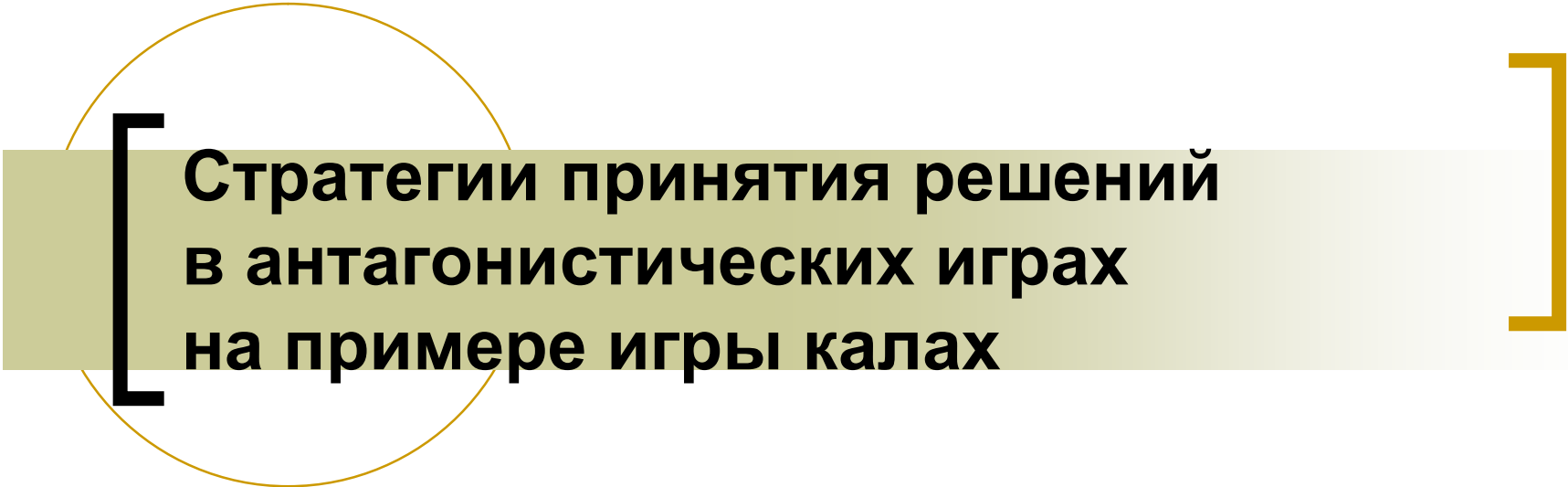




Стратегии принятия решений в антагонистических играх на примере игры калах

Работу выполнил: Борович Андрей
Алексеевич, 6.3 класс;

Руководитель: Серов Юрий Михайлович

[Цели работы]

- Рассмотреть алгоритмы принятия решений в антагонистических играх.
- Создать информационную модель (программу), позволяющую смоделировать и реализовать процесс принятия решения в игре калах.

[Задачи работы]

- Познакомиться с методами программирования антагонистических игр.
- Написать программу для игры в калах, реализующую алгоритм поиска лучшего хода.
- Рассмотреть различные оценочные функции.
- Собрать статистику по игре.

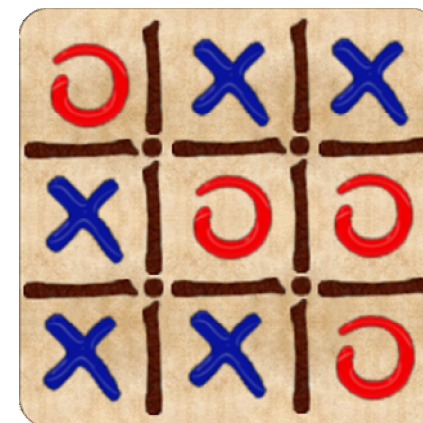
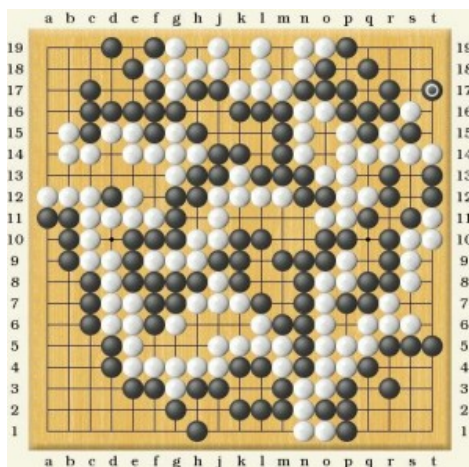
Антагонистические игры

- это игры с участием двух игроков, выигрыши которых противоположны

10^{171}

10^{43}

<512

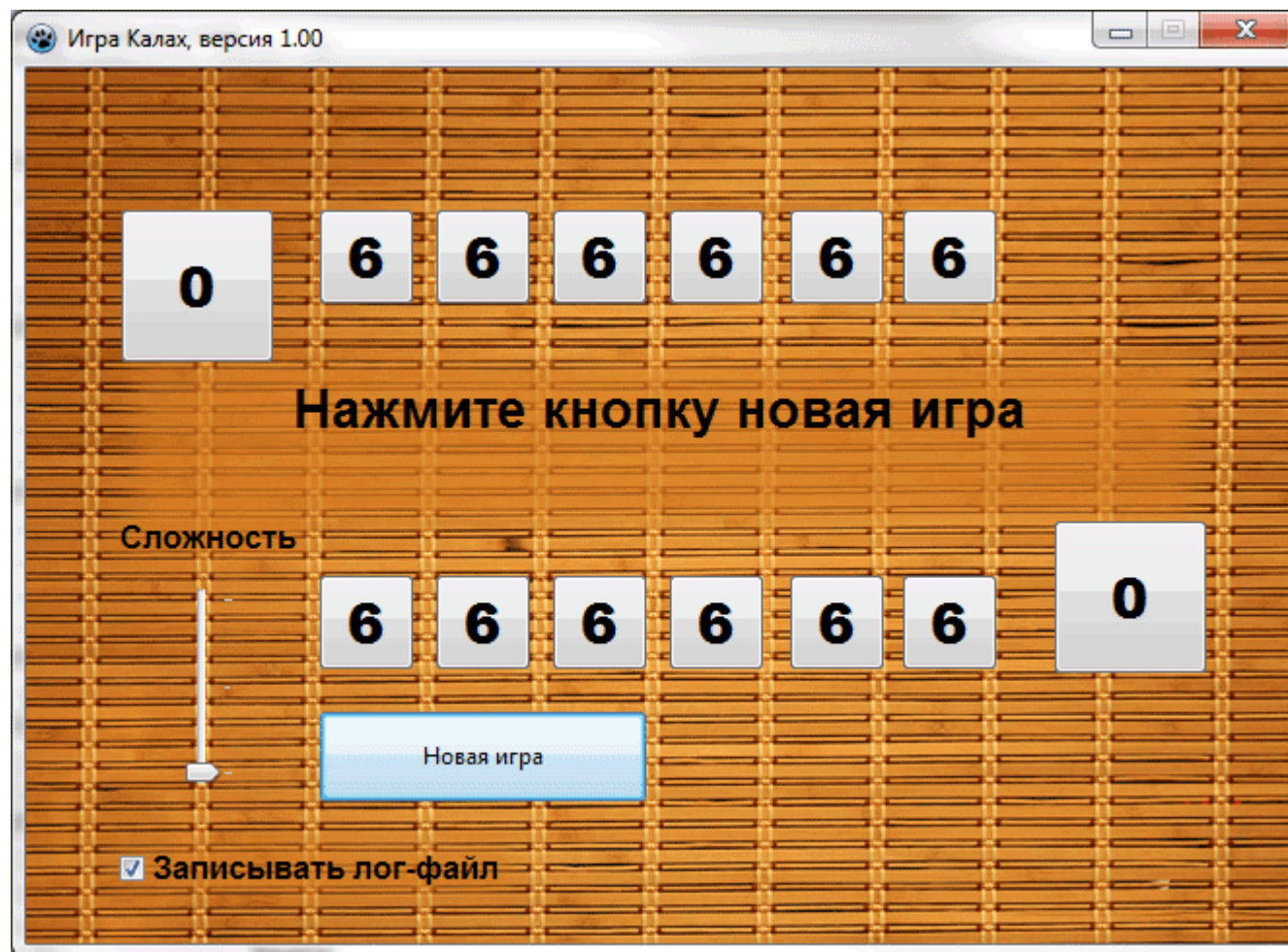


[Правила калаха]

- Игрок берёт все камни из одной своей лунки и раскладывает по одному в последующие лунки против часовой стрелки.



[Внешний вид программы]



[Уровни сложности]

- Уровень 1. Компьютер притворяется «болваном» и делает ход из первой непустой лунки.
- Уровень 2. Компьютер оценивает свой ход и ответ соперника.
- Уровень 3. Компьютер оценивает свои ходы и ответы соперника на 4 хода вперёд.

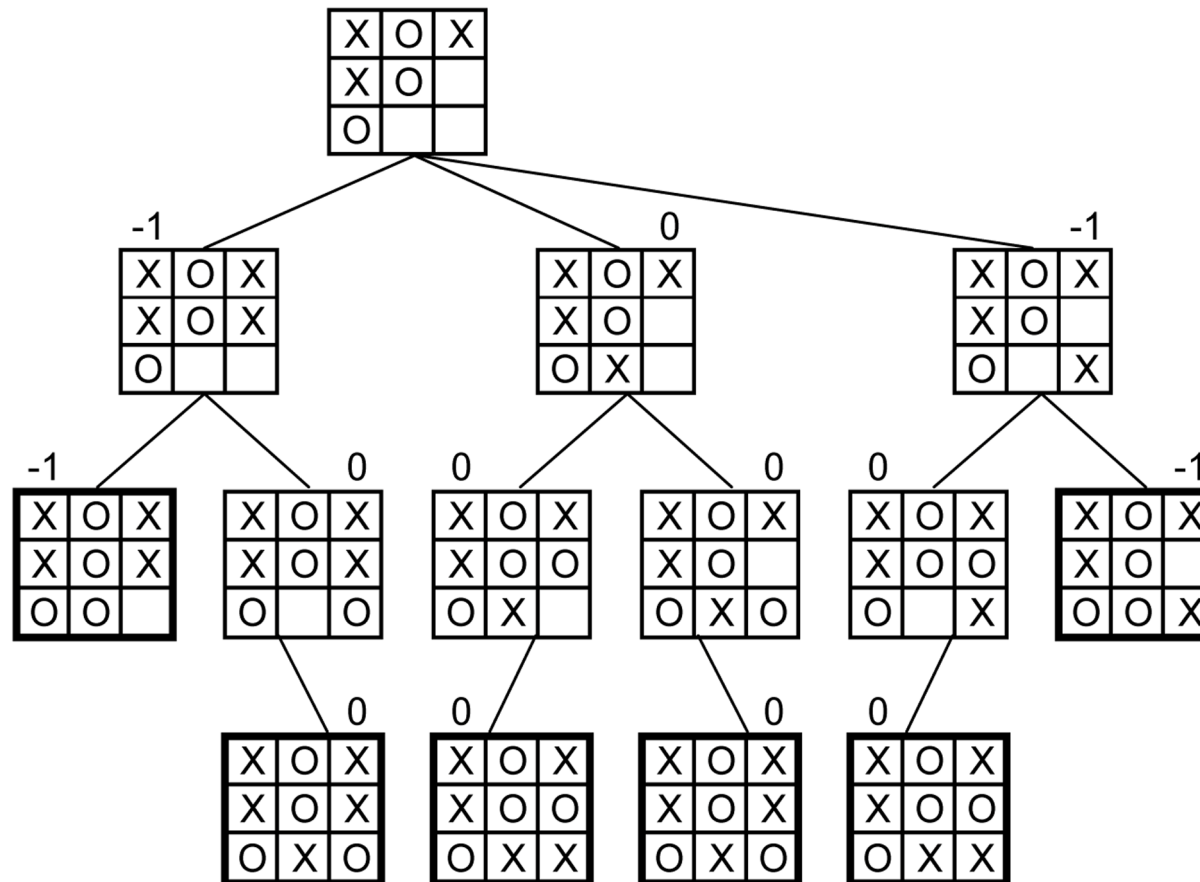


Стратегии поиска хода

Два типа стратегий (по К.Шеннону):

- Тип А – перебор всех возможных ходов на фиксированную глубину и использование в конце оценочной функции.
 - MiniMax
 - NegaMax
- Тип В – выборочное расширение определённых ветвей, используя накопленные знания игры, чтобы «подрезать» ненужные ветви дерева решений.
 - Alpha-beta отсечение
 - Метод Монте-Карло

[Дерево решений]



[Оценка позиции]

Наиболее простая оценочная функция для калаха:

$$\text{evaluate} = K_h - K_c$$

В моей программе используется более точная оценочная функция такого вида:

$$\text{evaluate} = (K_h - K_c) * 100 + N_h - N_c,$$

где K_h и K_c – количество камней в калахе игрока и программы; N_h и N_c – сумма камней в лунках игрока и программы.

Статистика

	Игры	Победы игроков	Процент побед
Уровень 1	17	14	82%
Уровень 2	19	8	42%
Уровень 3	24	4	17%

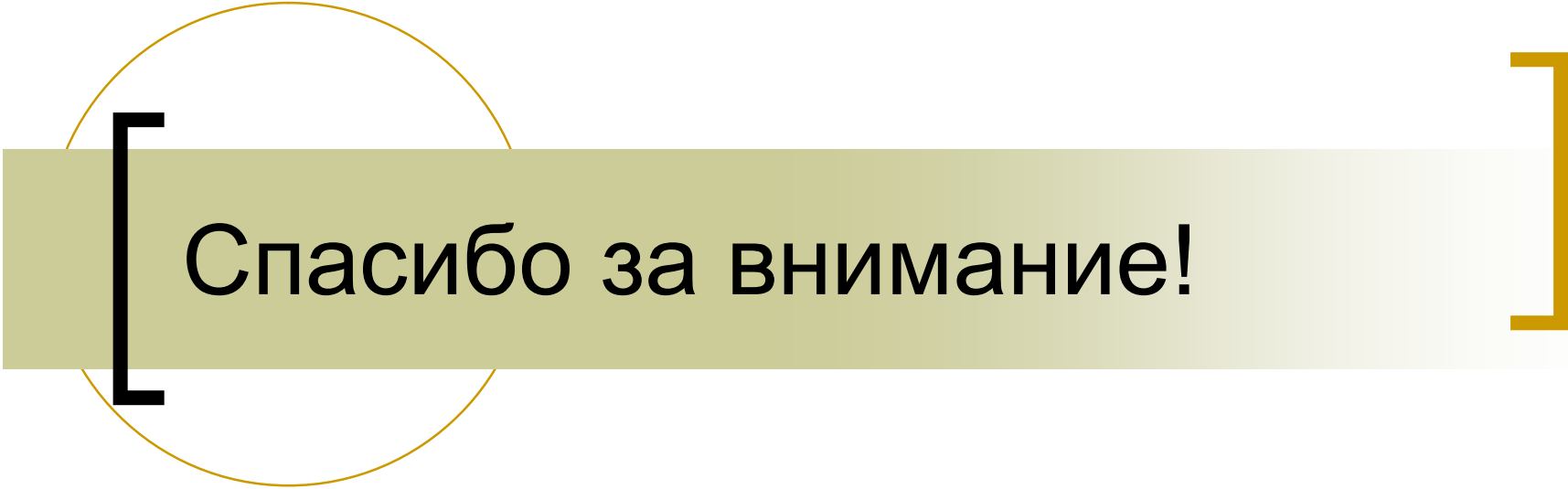
На сложном уровне из 24 партий программа выиграла 20, то есть доля выигрышей составляет 83%.

[Демонстрация]

- Kalah.exe

Выводы

- Простые алгоритмы (учитывающие только текущую позицию) не могут в большинстве случаев побеждать сильного соперника-человека.
- Созданная мной программа (уровень 3) выигрывала у людей с высшим образованием в 83% случаев. Программа является довольно сильным «игроком», обыгрывающим взрослого человека.
- Алгоритмы поиска лучшего хода, опирающиеся на полный перебор дерева позиций вглубь, не дают хорошего быстродействия и требуют дополнительных алгоритмов (alpha-beta отсечение, метод Монте-Карло).



Спасибо за внимание!

Работу выполнил: Борович Андрей
Алексеевич, 6.3 класс;

Сайт: <http://andrey.borevich.ru>

E-mail: borevich.andrey@gmail.com