

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования



НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

Программа, играющая в Английские шашки 12x12

Отчет по курсовой работе

Этап №1

Вариант № 17

по дисциплине

Алгоритмы и структуры данных

РУКОВОДИТЕЛЬ:

Капранов С.Н.

СТУДЕНТ:

Сапожников В.О.

19-ИВТ-3

Работа защищена «___» _____

С оценкой _____

Содержание

1. Текст задачи.....	3
2. Оценочная функция.....	4
2.1. Описание вычисления.....	4
2.2. Пример расчета.....	5
2.3. Программный код.....	7

1. Текст задачи

Написать программу, играющую в Английские шашки на поле 12x12.

Замечание: стандартные Английские шашки играют на поле 8x8, данная модификация играет на поле 12x12 (Размер поля и расстановка шашек аналогичная Канадским шашкам).

Правила игры:

Цель игры — уничтожить все шашки противника или лишить их возможности хода («запереть»).

Игра проводится на доске **12×12** клеток. В начальной позиции у каждого игрока по **30** шашек, расположенных на первых пяти рядах на черных клетках.

Первый ход делают белые шашки (Игрок - человек). Простая шашка ходит по диагонали вперёд на одну клетку.

При достижении любого поля последней горизонтали, простая шашка превращается в дамку. Дамка может ходить на одно поле по диагонали как вперёд так и назад.

Взятие обязательно, если оно возможно. Шашки снимаются с доски лишь после того, как берущая шашка остановилась. При нескольких вариантах взятия игрок выбирает вариант взятия по своему усмотрению, и в выбранном варианте необходимо бить все доступные для взятия шашки. При взятии дамка побьет только через одно поле в любую сторону, а не на любое поле диагонали, как в русских или международных шашках.

2. Оценочная функция

2.1. Описание вычисления

Введем стоимость шашек: обычная шашка стоит 1000 очков, дамка стоит 4000 очков.

Поскольку в алгоритмах минимакса и альфа-бета отсечения компьютер будет пытаться получить максимальное значение от оценочной функции, а в ходе партии играть за «черную» команду, то формула примет вид:

$$f(board) = 1000(bc - wc) + 4000(bk - wk),$$

где bc – кол-во черных простых шашек, wc – кол-во белых простых шашек, bk – кол-во черных дамек, wk – кол-во белых дамек; $board$ – переданное состояние игровой доски.

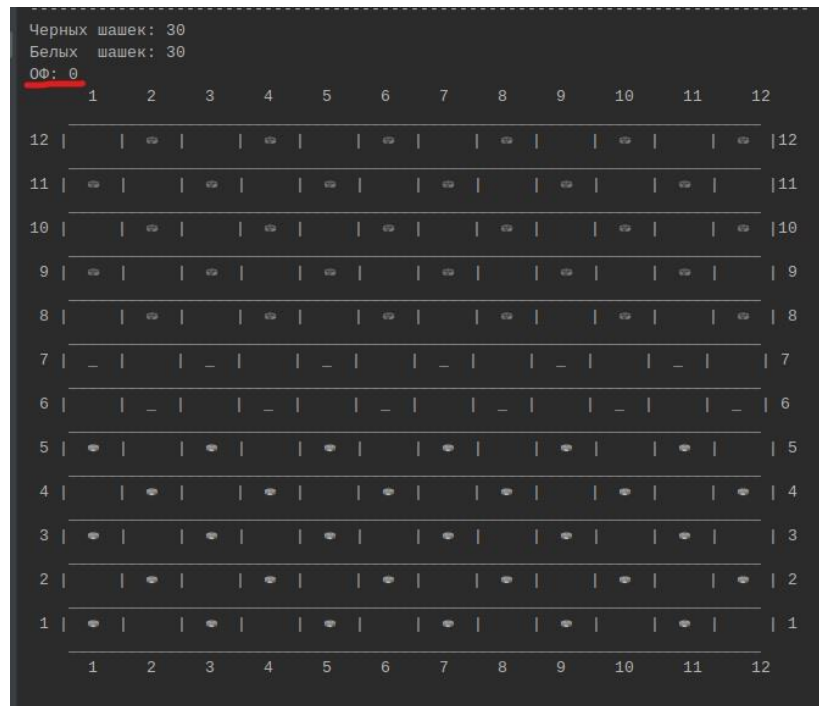
В программной реализации дабы не хранить отдельные значения для кол-ва белых и кол-ва черных дамек, отдельно кол-во обычных и каждый раз вычислять общее кол-во всех шашек каждой команды, было принято решение проводить перебор клеток поля (поле – массив клеток класса `board`), игнорируя недоступные (белые) клетки поля. Тогда оценочная функция примет вид:

$$f(board) = \sum_{i=0}^{11} \sum_{j=i \% 2}^{11} x,$$

где x меняется по закону: $x = \begin{cases} 1000, & \text{если в клетке черная шашка} \\ -1000, & \text{если в клетке белая шашка} \\ 4000, & \text{если в клетке черная дамка} \\ -4000, & \text{если в клетке белая дамка} \end{cases}$

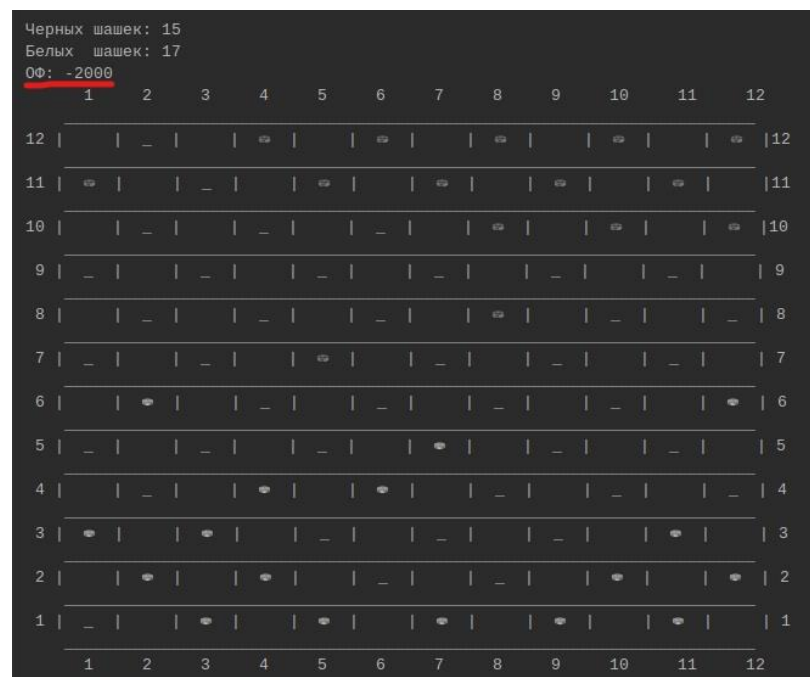
Замечание: данный алгоритм оценочной функции показал свою состоятельность в демоверсии алгоритма минимакса в ряде тестирований.

2.2. Пример расчета



Скриншот 1.

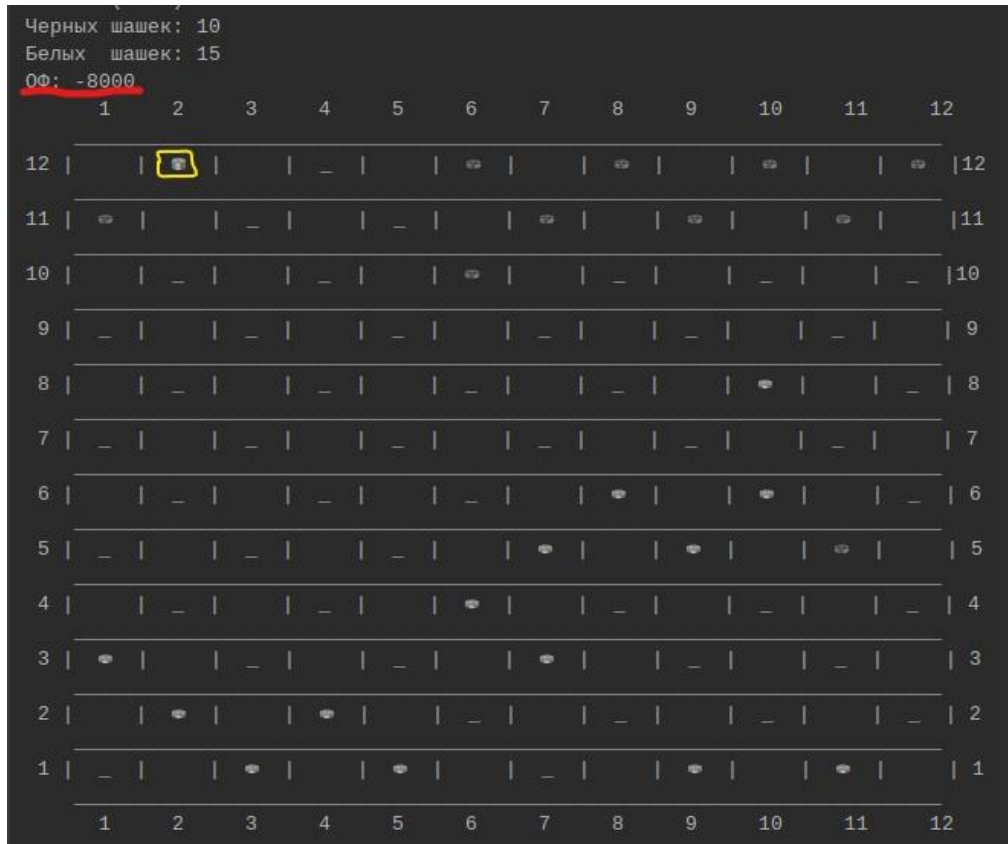
Начальное положение доски (скриншот 1), по 30 обычных шашек с каждой стороны. Условия игроков равнозначны, значение, возвращаемое оценочной функцией равно 0.



Скриншот 2.

На данном поле (скриншот 2) кол-во белых шашек(игрок) на 2 больше кол-во черных(компьютер). Значение, возвращаемое оценочной функцией:

$$1000(15 - 17) + 0 = -2 * 1000 = -2000$$



Скриншот 3.

В данном случае (скриншот 3) кол-во белых (игрок) превосходит кол-во черных (компьютер) на 5 шашек, но среди белых так же присутствует дамка (отмечена желтым цветом). Значение, возвращаемое оценочной функцией:

$$1000(10 - 14) + 4000(0 - 1) = -4 * 1000 + (-1) * 4000 = -8000$$

2.3. Программный код

MoveEvaluator.java

```
/**
 * Класс, в котором реализованы все методы
 * для оценки ходов.
 * */
public class MoveEvaluator
{
    /**
     * Введем некоторую стоимость для шашек.
     * */
    public final int POINT_KING          = 4000;
    public final int POINT_NORMAL        = 1000;

    /**
     * Оценочная функция
     * */
    public int fieldAssessment(Board board)
    {
        int value = 0;

        // проход по доске.
        for (int r = 0; r < Board.rows; r++)
        {
            // проверка только доступных для хода столбцов
            for (int c = (r % 2 == 0) ? 0 : 1; c < Board.cols;
                c += 2)
            {
                CellContents entry = board.cell[r][c];

                //Поскольку компьютер играет за черных
                //и пытается получить максимальное значение от
                //оценочной функции, ведется следующая
                //расстановка знаков
                if (entry == CellContents.white)
                {
                    value -= POINT_NORMAL;
                }
                else if (entry == CellContents.whiteKing)
                {
                    value -= POINT_KING;
                }
                else if
                (entry == CellContents.black)
                {
                    value += POINT_NORMAL;
                }
                else if (entry == CellContents.blackKing)
                {
                    value += POINT_KING;
                }
            }
        }
    }
}
```

```
        return value;
    }
```

Объект класса Board (доска) содержит поле-двумерный массив размерности 12x12 значений перечисляемого типа CellContents, который отвечает за содержимое клеток. По ходу игры значение элементов массива (содержимое клеток) будет меняться.

Код переселяемого типа CellContents

```
/**
 * Последовательность, отвечающая за
 * содержание клеток.
 * */
enum CellContents
{
    inaccessible,    //недоступная клетка
    empty,           //пустая доступная клетка
    white,           //белая шашка
    whiteKing,       //белая дамка
    black,           //черная шашка
    blackKing        //черная дамка
}
```