

# Homework 3: ILPs and Game Search

*Due: 5:00 PM, Mar 2, 2009*

## Contents

<b>1</b>	<b><math>\alpha\beta</math> Pruning</b>	<b>1</b>
<b>2</b>	<b>Phong</b>	<b>2</b>
<b>3</b>	<b>Maximax</b>	<b>2</b>
<b>4</b>	<b>Checkers</b>	<b>2</b>
<b>5</b>	<b>Chess</b>	<b>3</b>

## Objectives

By the end of this homework, you will know:

1. the minimax algorithm and  $\alpha\beta$  pruning

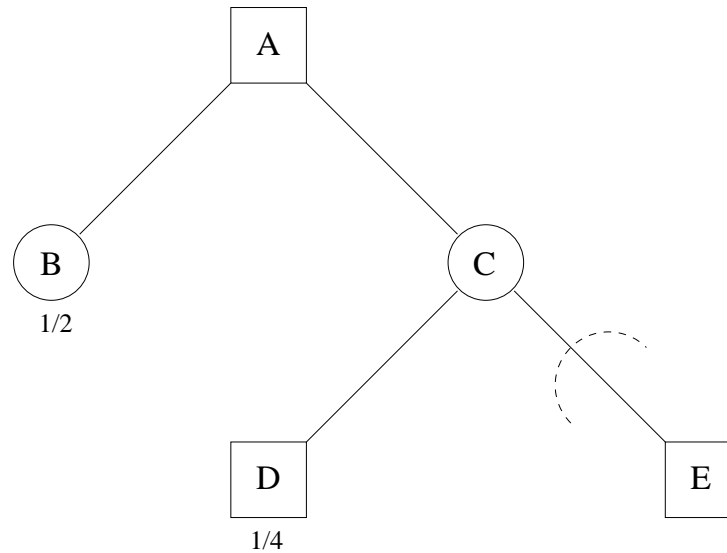
By the end of this homework, you will be able to:

1. express optimization problems as ILPs

## Practice

### 1 $\alpha\beta$ Pruning

Trace  $\alpha\beta$ PRUNING on the following abbreviated game:



## Problems

### 2 Phong

Consider the following two-player, alternating move, zero-sum game. The game begins with one pile of seven coins. At each player's turn, s/he picks one pile of coins and divides it into two piles such that the number of coins in the first pile is a multiple of the number of coins in the second. For example, a pile of four coins may be divided into piles of one and three coins, or two piles of two coins. The game is over when the largest pile contains only two coins. The losing player is s/he that makes the last move.

Draw the game tree for this game. Compute the minimax value of the game. Indicate which nodes in your game tree (if any) can be pruned via  $\alpha\beta$  pruning.

### 3 Maximax

Consider the labeling of 2 player game trees not with a single value, but with a pair of values: *e.g.*,  $(x, -x)$ . The first (second) value denotes the *payoffs* to player 1 (2). Extend this labeling and the MINIMAX algorithm to handle  $N$  player games.

### 4 Checkers

Checkers is a 2 player board game where the objective is to capture all of one's opponents pieces. There are two types of pieces, checkers, which can move only in the forward direction, and kings, which can move both forwards and backwards. Give a (sensible) heuristic evaluation function  $e$  for the game of checkers in terms of the number of checkers  $c$  and the number of kings  $k$  on the board for each of red and black.

## 5 Chess

**Task:** The  $n \times m$ -queens problem is “played” on an  $n \times m$  sized rectangular chess board. Given such a board, for arbitrary  $n$  and  $m$ , write an ILP to answer the following question: *what is the maximum number of queens that can be placed on the board such none is attacking any other?* Two sample configurations of queens on  $6 \times 9$  boards appear in Figure 1—one legal and one illegal.

							Q	
	Q							
					Q			
								Q
				Q				
Q								

	Q							
		Q						
							Q	
Q		Q						
								Q

Figure 1:  $6 \times 9$ -Queens. LHS: A valid configuration. RHS: An invalid configuration.

For those of you who are not familiar with the legal moves of chess pieces, a queen can move as many cells as she likes in any direction she likes—along a row, along a column, or along a diagonal. At the end of this writeup, there is a pointer to a resource with short descriptions of the legal moves in chess, which should provide all of the information about chess that is necessary to complete this assignment. Feel free to see a TA on hours if you have questions.

Read on before you start working on this problem. The second part of the problem builds on the first, and you may want to keep that in mind while working.

**Task:** A second problem we would you like for you to express as an ILP is the following: *what is the maximum number of points that can be scored by placing queens, rooks, bishops, and knights on an  $n \times m$  chess board such that none is attacking any other, where the pieces are valued at  $q, r, b$ , and  $k$ , respectively?* Typically,  $q = 9$ ,  $r = 5$ ,  $b = 3$ , and  $k = 3$ ; however, your formulation should be general enough to handle arbitrary point values as inputs to the problem.

**Tip:** Here are some questions to think about while formulating these problems as integer programs:

- What function is being optimized (i.e. maximized or minimized)?
- What are the decision variables in this problem?
- What are the constraints on the decision variables?  
e.g., Are they integer-valued or real-valued?
- How can the rules governing the movement of pieces be formulated as constraints?

**Extra Credit:** Write an ILP to solve the *Knight’s Tour Problem*, a variant of the traveling salesperson problem on a chess board: *can a knight visit all the squares on a chess board exactly once and return to its original position?*