

Computer Chess Game — project made on LPP classes.

Katarzyna Obrycka Marek Chruźciel

Wrocław, 29 czerwca 2008

1 Introduction

Chess is a well known game played between two players. It's basic rules knows almost everyone, and everyone probably played chess a couple of times. If you're not familiar with chess rules please look at

<http://en.wikipedia.org/wiki/Chess>. Our interest as informatics student were computer chess. We tried to develop an artificial intelligence which plays chess. We also used an ready-to-use graphical interface - XBoard. This report will provide you information about our work, ideas, success and failures.

2 General idea how to play chess.

Computer chess has been analyzed and developed for long time now. We use only well known algorithms. The game is played between two player. Moves are taken alternately. That's why almost every chess programm uses some kind of MiniMax algorithm (<http://en.wikipedia.org/wiki/Minimax>). So does our programm. We use in general the MiniMax with some heuristics. One thing is remebering the whole game graph. Because of that we have to compute the evaluation function and the move generation for particular game state only once. The game state evaluation function is generally based on counting pieces that are left on board. Every single piece of it's type receives some predefined value.

3 Game graph and algorithm

.... To follow

4 Evaluation function

Like mentioned we use counting pieces as base for the function. We also give some high bonus for checking the opposite king, or the mat. Pieces standing in the middle of the board also receive some bonus, as they are 'better standing'.

Values for piece counting :

PAWN_VALUE 100
KNIGHT_VALUE 300
BISHOP_VALUE 325
ROOK_VALUE 500

QUEEN_VALUE 900
KING_VALUE 1000000

5 Rules implementation

All the basic rules are programmed. The following rules are also included - castling, en passant, promotion.

6 Programming work, language.

The whole work is done in C++. We used svn for easier source code management. You can look at the project web site <http://code.google.com/p/iichess/>. There is not much there but you can download the source from the svn repository. We tried to work on-time and we had a couple of meetings to discuss current problems. I guess the choice of C++ language was a little drawback, especially as we had more experience in use of other programming languages.

7 XBoard

As graphical interface for our program we used XBoard, can be found on <http://tim-mann.org/xboard.html>. Documentation about chess engine communication protocol is available on <http://tim-mann.org/xboard/engine-intf.html>. We have implemented only the 'normal' variant of chessgame for it and some of the XBoard features:

1. 'Playother' which allows player to play opponents colour at any moment,
2. 'Usermove' which interrupts computer's thinking and forces move on it.

8 Successes

We developed a decent chess program with a nice XBoard graphical interface. We have done all the rules, talking about this it takes a lot of 'if' and 'switch' instructions to make it. Our code is pretty fine, no mess, comments included. We have also learned a lot.

9 Failures

First of all like in almost every project students make on classes we have had some problems finding free time and some motivation. We had some ideas that we did not managed to finish like - game openings, some work on performance, use of heuristics, pondering. We spent a lot of time finding bugs, that probably could be detected earlier if we did managed and plan better.