Department of Computer Science

Artificial Intelligence
COS314

Project 1: Game Trees

Due: Wednesday 28 March, 14:00

For this project, you will use a game tree to implement a *Game of Towers*. Games of Towers are two-player abstract board games that can be played with stacking pieces (e.g., games where it is possible to create 'towers', 'stacks' or 'piles'). A total of 92 such games are listed and described at http://www.di.fc.ul.pt/~jpn/gv/towers.htm.

For this project you are going to have to make a number of choices. You will be given some kind of freedom, but some freedoms do come at a cost. Find below the rules for this project:

1. Select any one of the games listed on the website given above. Be careful, they differ somewhat in complexity. The keys provided with each game in that list will give you an idea of the difficulty level. The more difficult the game, and provided that your program works, the better the marks that you will receive.
2. Decide if a minimax or alpha-beta game tree will be used. More marks will be obtained for the alpha-beta pruning algorithm.
3. For the game that you have chosen, you need to define your own evaluation function. You have to describe this evaluation function in a pdf document. See below.
4. The project is done by each student individually, and all code has to be written by yourself.
5. You may implement the project either in C++, C, Java, or Scala, and must run on Linux. You have to provide a Makefile, ant file, or any other alternative project script that will run on Linux. Describe in the pdf file how the project should be compiled and used.
6. Provide a GUI for your game.
7. Your game should allow for the computer to play against a human, and should also provide for two programs to play against one another, at different user defined ply depths.
8. Although this is not a programming course, you should always provide well-structured, modular, well-written and documented code.
9. Submit your project before the deadline via the online submission system as a compressed tar archive. Provide a pdf file, *project.pdf* in the root file to which your tar ball extracts, wherein you explain how the program should be compiled and used, give your name, surname and student number, and give the game

you have implemented as well as the rules of the game. Also describe your evaluation function in this pdf document.

10. By submitting, it is assumed that you have done, and submitted your own work. If found that the work is not your own, it will not be evaluated.

The project will be evaluated using the folliwing marking scheme:

Aspect		**Mark**
The AI used:		
	Minimax	15
	Alpha-Beta	30
	Evaluation Functions	20
	Any other things done?	10
	Did it work?	40
	Game complexity	20
	Sub-total	**135**
GUI:		
	Text-based	5
	Graphical	20
	Sub-total	**25**
Coding style:		
	Modularity	10
	Documentation	10
	Sub-total	**20**
Total:		**180**