Term Project

This semester you will implement a game called "Numbrix". You can play a version of this game at http://www.parade.com/2596/marilynvossavant/play-todays-numbrix/ This game is played on a square grid (the size of the grid can vary). To refer to positions on the grid, we use the following numbering convention: each position is referenced by two numbers: its column and its row. In the gird below, the lower left corner (where 58 is located) is position (1,1), the lower right corner (where 5 is located) is (1,8), and the upper left corner (where 45 is located) is (8,1).

45	44	39	38	23	22	19	18
46							17
47							16
48							15
63							14
64							3
59							4
58	57	56	55	8	7	6	5

A game starts with various positions filled with numbers. The objective of this game is to place numbers into the open cells so they make a path in numerical order, 1 through the last number. Note: the number 1 and the last number are NOT guaranteed to be on the initial board. You can add numbers horizontally or vertically in any direction. Diagonal paths are not allowed. The solution to the above problem is given below:

45	44	39	38	23	22	19	18
46	43	40	37	24	21	20	17
47	42	41	36	25	26	27	16
48	49	50	35	30	29	28	15
63	62	51	34	31	12	13	14
64	61	52	33	32	11	2	3
59	60	53	54	9	10	1	4
58	57	56	55	8	7	6	5

CAP5535 students are required to use Lisp in developing their solution. CAP 4621 students can use the language of their choice (though a 10% bonus will be given for using Lisp).

Project Time Table

The following are important dates when project reports or code submissions are required.

Oct. 18 Initial submission/“demonstration” of project (submission)

This submission will show that your program can play this game with no intelligence. Your program should request and read the name of a file containing a set of data describing the size and the starting numbers on the board, should allow a human to specify moves (moves will be specified as three numbers separated by blanks with no other punctuation), should make the move and display the new game board, should recognize when a game is completed, should display all moves made in playing the game, and should ask if another game is desired (repeating this process if required). Any other added features ("bells and whistles") which you feel should be part of the user interface should be demonstrated here (for example, displaying a set of rules to the game is *highly* recommended). Each student will turn in a zipped copy of their program on this date with a “read me” file describing how to execute your program. The main routine for your program should be called “numbrix”.

Oct. 21 Written report (submission)

A PDF or Word file of your written report should be turned in on this date (1-2 pages in length). This report will detail what intelligence you intend to implement in your program. Note: you might not be able (due to time until the end of the semester, time limits for a move, and space within the machine) to fully implement all of this. Your report should therefore be structured to contain the following two sections:

1. This is what I will definitely implement, and
2. This is what I hope to implement provided I don't have any major difficulties.

Nov. 1 Search Implemented (NO submission)

On this date I encourage you to have a search program implemented that, as a minimum, performs a blind depth-first search of the problem filling in and changing values on the board until a solution is generated. No submission of code is required on this date!! This is merely a road marker for you. From this date until the final submission, you can concentrate on implementing your intelligence.

Nov. 25 All projects must be fully implemented and running for final submission (submission)

This submission will be used to show what you have as a final program. You will be graded on the program playing a legal game (not making illegal moves), any changes which have been made since the last “demonstration”, and how well the program plays. Your program should allow humans to play manually or the computer to play automatically. During automatic play, your program should display the board after each move, and should print a list of moves made after the game is completed with the total CPU time required to generate that solution. You are to submit a zipped version of your program and, again, a read-me file describing how to load your program and any other directions required to execute your program.

Nov. 25 – Dec. 4 Tournament (submission)

I will post a series of data sets that you will be required to run your program against. You must provide the move sequence generated by each of these data sets in your final project report. You should also report the execution (CPU) time taken to generate each solution. Each solution must be found in no more than 1 CPU minute (I reserve the right to change this value).

We will verify your results by testing your program ourselves on one or more of the tournament data sets to verify your results.

Dec. 4 Final written report (submission)

This report will detail the entire project. As a minimum this report should include:

1. A description of the game which was implemented.
2. A description of the implementation approach which was taken.
3. A description of all programs written and all major variables used.
4. A flowchart of the program.
5. A calling hierarchy or UML diagram showing the various methods/functions and how the various they call each other.
6. A description of what intelligence was implemented and what was not. Discuss why the non-implemented intelligence was not implemented.
7. A discussion of what you would do differently if you were able to start over.
8. A complete code listing.
9. The move sequences, time used, and number of beads remaining that was generated by your program for each tournament data set.

Grading

The grading of the project will be according to the following scale:

Initial Submission 10%
Intelligence Report 15%
Play against tournament data sets 25%: 15% for playing all data sets (regardless of whether you successfully complete the data set) and reporting your results and 10% for the intelligence of play.
Final Submission 15%
Final Report 35%