Fall 2004
Instructor: B. Ross

COSC 3P71 Artificial Intelligence: Assignment 3

Due date: 12:00 noon, Tuesday, November 30.
Late date: 12:00 noon, Friday, December 3. (-25%)
Goal: Genetic algorithms.
Languages: Any programming language of your choice.
Hand in: **(1)** A listing of your source code, and a listing of your program execution for a representative sample of cases. **(2)** Electronic submission of your code and data (see assignment 1). **(3)** A 3-page report with Excel graphs that describe your experiment parameters and results.. Programs marked on style, documentation, and correct execution.

Task: Implement a genetic algorithm! The major modules of the system are:

(a) **Population initializer:** creates a population of size N of randomized individuals.
(b) **Crossover**: given two individuals, this creates two offspring created using crossover.
(c) **Mutator**: given an individual, this creates a mutated individual.
(d) **Evaluation function**: for the problem being used, this takes an individual, and returns a score, where the higher the score, the fitter the individual.
(e) **Genetic algorithm engine**: This is the implementation of a generational GA engine (see attached flowchart).

You can use roulette wheel or tournament selection. If you use roulette, then massaging the fitness scores (eg. squaring the scores) can affect performance.

Make your GA as parameterized as possible. This lets the user refine experiments by setting new genetic parameters (eg. probability that crossover is used, population size, max generations...) .You should implement your system so that you do not need to recompile your application for new experimental parameters. Rather, you should read in the values in some manner at the beginning of a run.

At the end of each generation, the system should print the best chromosome's score, and average population score for that generation. At the end of the run, it should print the best chromosome found during the entire run.

Problem: Super N-Queens

The problem you are to run your system on is the Super N-Queens problem from assignment 1. You can use either Queens or Rooks. However, use the same version that you used in assignment 1, including the same board scores, heuristic scoring function and internal board representation, so that you can compare your results.

You are to do the following 3 experiments:

i) Super N-Queens, 4x4, K=4, hard rule.
ii) Super N-Queens, 8x8, K=8, hard rule.
iii) Classic N-Queens, 8x8.

You should first experiment with different parameters (population sizes, maximum generation, probability of crossover, etc.). Once you have found a good set of parameters, run the same experiment at least 5 different times, using different random number seeds. Collect all the output for those runs.

Report: Include a 3-page (minimum) report that summarizes your experiments. It should include the following:

- Description of experiments.
- Table of parameters used. (Anyone should be able to duplicate your results).
- Table summarizing the best solutions obtained for the 3 experiments, for all 5 runs.
- Indicate the total # individuals processed during a run.
- Also include a comparison between the results of the GA, compared to the search you used in assignment 1. What methodology was superior for this problem?
- Finally, include 2 Excel graphs, which show the performance (best score, average score) of the best run in each experiment. Plot fitness vs generation number.