GAGP Tutorial 4 (week 7)
Genetic Programming

1. Genetic programming (GP) is an evolutionary technique which attempts to evolve programs fit for some purpose. Describe a typical GP system: explain how programs are represented in the system; give examples of the genetic operators applied; and state the main steps of the evolutionary algorithm indicating where there are design choices to make.

2. Express the following functions in Lisp notation, using only $+ - */$ as non-terminals and $x, 0, 1, 2, 3, \ldots$ as terminals.

    (a) $y = 3x + 2$
    (b) $y = 5x^4 - 2x^2$
    (c) $y = -0.25x^3 + 3.5$

3. Which of the functions in question 1 can you represent using only x and 1 as terminals?

4. What fitness function can GP use for solving symbolic regression problems? Can you think of any alternatives? How much domain specific knowledge about the problem is encoded in this fitness function?

5. Do schemata and building blocks exist in Genetic Programming populations?

6. A GP system is employed to evolve a controller for a mobile robot. The fitness function evaluates the robot performance starting from 50 initial positions. In a long series of tests the system is observed to produce a satisfactory controller in 70% of runs.

The system designer decides to improve the GO system by speeding it up, and reduces the number of fitness cases to 25. The system now produces a satisfactory controller in only 50% of runs. Is this an improvement?

(Hint: consider the amount of work the system must do to produce, evnetually, a satisfactory controller.)