# COSC 5P71 Genetic Programming - Term Test

B. Ross

Name: _____ Student number: _____

*There are 6 questions on 7 pages totaling 91 marks. No aids are permitted. Please use the back of pages if you need more space for your answers.*

**1. [24]** Multiple choice [+2 correct; 0 empty; -0.5 wrong guess; one free wrong guess permitted] Select the single best answer for each question.

1. Applications like the ant trail and Boolean circuits benefit with the use of ADF's because:

a. they have multiple objectives in their search space
b. they have single hills in their search space
c. they have inherent modular regularities that can be exploited
d. they naturally promote closure

2. An evolutionary computation technique that uses a single population:

a. coevolution
b. ADF
c. steady state
d. none of the above

3. GP systems often include a user parameter, "probability of internal crossover". This is required because:

a. otherwise leaf nodes might be selected too often
b. it prevents run-time errors (e.g. divide by zero)
c. it reduces program bloat
d. none of the above

4. Which of the following is a characteristic of "high selection pressure":

a. genetic diversity will be maximized
b. genetic diversity may be sacrificed
c. will result with a tournament size of 2
d. not possible with Roulette wheel selection

5. When considering computational complexity issues, evolutionary computation is suitable for discovering:

a. optimal solutions to NP-complete problems
b. solutions to virtually any search problem conceivable
c. reasonable solutions to NP-complete problems
d. dangerous solutions to silly problems

6. ADF is an acronym for:

a. automatically defined function
b. absolute depth factor
c. a defined-once function
d. acceptably deep formula

7. Which of the following is an appropriate use of genetic programming:

a. To discover the best move to make in a computer chess program.
b. To quickly search a binary tree for a target key value.
c. To evolve a “brain” for a computer game opponent.
d. To optimize the numeric coefficients for a given polynomial formula.

8. Island-model parallelism can be effective because:

a. It can exploit multiple CPU’s.
b. The islands help preserve genetic diversity in the population.
c. NP-complete problems become polynomial in complexity
d. all of the above

9. Which is an advantage of co-evolution:

a. It naturally supports multi-objective problems.
b. It can be implemented in an island-model GA.
c. It permits powerful, dynamic fitness evaluation.
d. Premature convergence is not possible.

10. The main step of GP that is most effectively accelerated with graphics cards processors:

a. computing Roulette wheel areas
b. crossover
c. solution testing
d. fitness evaluation

11. Cellular encoding involves:

a. generating graph structures suitable for neural networks
b. generating  a new node in parallel with a given node
c. starting with a basic embryo object
d. all of the above

12. One advantage of using sigma scaling of the population fitness is:

a. it improves the performance of tournament selection
b. it produces a constant selective pressure throughout the run
c. it distributes the fitness award proportionally to the diversity of behaviour
d. it is a way to handle high-dimensional many-objective problems.

**2. [15]** Briefly define the following terms, and their relevance in genetic programming.

(a) Pareto domination

(b) elitism

(c) linear genetic programming

(d) strong typing

(e) ephemeral random constant

**3. [12]** Write pseudo-code for the “grow” random tree generation algorithm. Also give an illustration of it working for a simple example problem. Include in your example a step-by-step explanation of all the steps.

**4. [12]** Discuss ways in which genetic programming can benefit with the use of parallelism, by identifying and discussing 3 examples of parallelism possible in GP. Also discuss how different kinds of parallelism affect speed (implementation) vs solution quality.

**5. [10+4+2=16]** Consider the following population of programs, and their associated objective vectors (3 objectives in total). The numbers in each objective are to be minimized (0 is the optimal score):

| # | Obj 1 | Obj 2 | Obj 3 | Pareto rank | Rank1 | Rank2 | Rank3 | Sum of ranks |
|---|---|---|---|---|---|---|---|---|
| 1 | 33 | 0 | 43 | | | | | |
| 2 | 0 | 47 | 43 | | | | | |
| 3 | 78 | 62 | 0 | | | | | |
| 4 | 43 | 19 | 20 | | | | | |
| 5 | 55 | 55 | 89 | | | | | |

a) Fill in the rest of the table for Pareto Ranks and Sum of Ranks. (Rank 1 is the rank of Obj 1 relative to the population, etc.).

b) Define “normalized sum of ranks”. How would it be applied to the above table of scores? (You do not need to actually compute it... just describe how it would be calculated).

c) Describe circumstances in which using a normalized sum of ranks is worth consideration.

**6. [6+6=12]** (a) Describe the No Free Lunch argument. Discuss the main idea behind the proof. What are the practical results of this theorem. In other words, does it mean that evolutionary computation is futile to ever use?

(b) Discuss the computational complexity aspect of evolutionary algorithms such as GP. Does GP solve NP-complete problems? If so, how? And if not, why is GP used at all?

***** The End *****