CS5401 FS2015 Exam 2 Key

This is a closed-book, closed-notes exam. The only items you are allowed to use are writing implements. Write your name in the designated spot on the top left of each of the exam pages. If you are caught cheating, you will receive a zero grade for this exam. The max number of points per question is indicated in square brackets after each question. The sum of the max points for all the questions is 50. Note that this exam consists of 20 multiple-choice questions followed by a single multi-part open question. You have exactly 75 minutes to complete this exam. Good luck!

Multiple Choice Questions - circle the letter of your choice on the exam pages

1. There is no recombination in “standard” Evolutionary Programming (EP) because: [2]
   (a) extensive research has shown that the use of recombination is counterproductive in EP [0]
   (b) EP was conceived before the invention of recombination [0]
   (c) each individual in “standard” EP is viewed as the abstraction of a species [0]
   (d) all of the above [0]
   (e) none of the above [0]

2. The phenomenon of bloat in GP occurs most likely because: [2]
   (a) individuals with bigger genomes have a larger chance of survival (also known as “survival of the fattest”) [1]
   (b) the variable length aspect of GP causes a natural tendency for the population to reflect the different possible sizes [2]
   (c) the ratio of alleles to genes in bloated individuals is higher than non-bloated individuals which gives them an evolutionary advantage [0]
   (d) all of the above [2]
   (e) none of the above [0]

3. Koza states that the aim of the fields of artificial intelligence and machine learning is to generate human-competitive results with a high artificial-to-intelligence (AI) ratio where the AI ratio of a problem-solving method means: [2]
   (a) the ratio of automation (generality) to human intelligence (speciality) needed by the problem-solving method to solve a particular problem [1]
   (b) the ratio of that which is delivered by the automated operation of the problem-solving method to the amount of intelligence that is supplied by the human applying the method to a particular problem [0]
   (c) the ratio of artificial intelligence to human intelligence employed by the problem-solving method [0]
   (d) none of the above [0]

4. Koza’s Automatically Defined Functions (ADFs) are: [2]
   (a) the application of GP to automate the creation of functions in computer programs [1]
   (b) the standard method of evolving reusable components in GP [1]
   (c) the use of GP to create functions with a high AI ratio [2]
   (d) none of the above [0]
5. Over-selection is employed in GP because: [2]
   (a) GP typically uses large trees which suffer from bloat [0]
   (b) GP typically uses fitness proportionate selection which suffers from premature convergence \(\frac{1}{7}\)
   (c) GP typically uses large populations which cause excessively high selective pressure \(\frac{1}{2}\)
   (d) all of the above [0]
   (e) none of the above

6. According to the concept of island model EAs in the context of Eldredge and Gould’s theory of punctuated equilibria: [2]
   (a) multiple populations of different species are run in parallel in some kind of communication structure [1]
   (b) after a usually variable number of generations, a number of individuals are selected from each population to be exchanged with others from neighboring populations \(1\frac{1}{7}\)
   (c) during the migration phase, the injection of individuals of potentially high fitness, and with possibly radically different genotypes, facilitates exploration
   (d) the migratory injections interrupt periods of evolutionary stasis by rapid growth when the main population is invaded by individuals from a previously spatially isolated group of individuals of a different species. \(1\frac{1}{2}\)
   (e) all of the above \(1\frac{1}{2}\)
   (f) none of the above [0]

7. In Diffusion Model EAs: [2]
   (a) individuals are modeled by diffusion equations and only panmictic mating is permitted [0]
   (b) the population is conceptually distributed on a grid and mating is restricted to demes
   (c) all of the above \(\frac{1}{2}\)
   (d) none of the above [0]

8. In Crowding: [2]
   (a) new individuals replace similar population members, resulting in the population sharing the niches equally
   (b) the fitness of individuals immediately prior to selection is adjusted according to the number of individuals falling within some prespecified distance of each other \(\frac{1}{2}\)
   (c) individuals share the fitness of similar population members immediately prior to selection, resulting in the number of individuals per niche being dependent on the niche fitness [1]
   (d) none of the above [0]

9. In Multi-Objective EAs employing levels of non-domination, increasing the number of conflicting objectives, generally will: [2]
   (a) not impact the number of levels of non-domination [0]
   (b) increase the number of levels of non-domination [0]
   (c) decrease the number of levels of non-domination
   (d) either increase or decrease the number of levels of non-domination, depending on the amount of selective pressure [0]
10. In Multi-Objective EAs employing levels of non-domination, a decrease in the number of levels, generally will: [2]
   (a) not impact the amount of selective pressure [0]
   (b) increase the amount of selective pressure [0]
   (c) **decrease the amount of selective pressure**
   (d) either increase or decrease the amount of selective pressure, depending on the number of conflicting objectives [0]

11. Which option is best for re-purposing the base Assignment 1d MOEA for the #SAT problem in which the challenge is to determine the total number of unique Boolean variable truth assignments which satisfy a given Boolean formula? [2]
   (a) when a solution is found, store its variable assignment and restart the entire population [1] (While restarts can be employed to search a landscape, niches with large basins of attraction will tend to dominate, making it unlikely to find solutions in niches with small basins of attraction.)
   (b) employ fitness sharing to promote diversity in order for the final population at convergence to hopefully contain all unique solutions [1 ½] (Fitness sharing tends to produce diverse Pareto fronts, but if the population size is smaller than the total number of unique solutions, then the Pareto Optimal Front will not be able to contain them all.)
   (c) no change needs to be made, simply stop when all clauses are satisfied by an individual and take 2 to the power of the number of don’t cares in that individual [1]
   (d) the number of non-dominated individuals in the final converged population (i.e., the Pareto Optimal Front) will per definition be equal to the total number of unique satisfying Boolean variable truth assignments [1 ½]
   (e) **employ fitness sharing and upon convergence, store all non-dominated solutions and restart the entire population. On termination, the union of all stored non-dominated solutions contains the approximate total number of unique Boolean variable truth assignments**

12. Dawkin’s concept of a “meme” is: [2]
   (a) the addition of a learning phase to the evolutionary cycle [1 ½]
   (b) a unit of biological transmission [1 ½]
   (c) **a unit of cultural transmission**
   (d) a process of imitation [1 ½]
   (e) all of the above [1 ½]
   (f) none of the above [0]

13. “Intelligent” initialization in a memetic algorithm can be performed by: [2]
   (a) Seeding [1 ½]
   (b) Selective Initialization [1 ½]
   (c) Locally Optimized Random Initialization [1 ½]
   (d) Mass Mutation [1 ½]
   (e) **all of the above**
   (f) none of the above [0]

   (a) improved EA performance is obtained through local search [1]
   (b) **acquired traits are passed on genetically**
   (c) improved EA performance is obtained through the Baldwin effect [0]
   (d) all of the above [1 ½]
   (e) none of the above [0]
15. The Baldwin Effect is: [2]
   (a) improved EA performance obtained by applying local search prior to fitness calculation [2]
   (b) improved EA performance obtained by applying local search after fitness calculation [1]
   (c) improved EA performance obtained by combining local search with Lamarckian evolution [1/2]
   (d) none of the above [0]

16. A Coevolutionary Algorithm (CoEA) is an EA: [2]
   (a) where the fitness of each individual depends on one or more individuals from a different species [1]
   (b) with exactly two populations [0]
   (c) with two or more populations [0]
   (d) where the fitness of each individual depends on one or more other individuals [2]
   (e) none of the above [0]

17. A Competitive Coevolutionary Algorithm is a CoEA: [2]
   (a) with two or more competing populations [1/2]
   (b) where each individual competes with one or more individuals in the competing population [1]
   (c) where individuals compete with each other to gain fitness at each others expense [1]
   (d) all of the above [1]
   (e) none of the above [0]

18. Disengagement in a two-population competitive CoEA occurs when: [2]
   (a) the individuals in both populations stop competing and start collaborating [0]
   (b) both populations get stuck in local optimums leading to a loss of search gradient [0]
   (c) one population gets stuck in a local optimum and the other population stops evolving because of a loss of evolutionary pressure [1]
   (d) all of the above [0]
   (e) none of the above

19. Mediocre stability in a competitive CoEA occurs when: [2]
   (a) the convergence of the system is not very stable [0]
   (b) the system stabilizes in a suboptimal equilibrium [2]
   (c) both of the above [1]
   (d) cycling causes instability in the system [0]
   (e) none of the above [0]

20. The Iterated Prisoner’s Dilemma problem: [2]
   (a) is technically not a competitive coevolution problem because it is a single population problem [0]
   (b) is technically not a competitive coevolution problem because it is a single species problem [0]
   (c) both of the above [0]
   (d) none of the above
Regular Questions - write your answer under the question on the exam page

21. Say you want to maximize the number of satisfied clauses in MAXSAT, while simultaneously maximizing the robustness of your solution by minimizing the number of specified boolean variables. You execute a multi-objective EA and the final population contains the solutions listed in the following table, where higher satisfaction indicates a larger number of satisfied clauses and higher robustness indicates fewer specified boolean variables:

<table>
<thead>
<tr>
<th>ID</th>
<th>Satisfaction</th>
<th>Robustness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td>8</td>
</tr>
</tbody>
</table>

(a) Plot the above table and use dotted lines to indicate the area of domination for each element. [2]

(21a)

(b) List for each element which elements it dominates; indicate elements with their IDs. [2]

<table>
<thead>
<tr>
<th>ID</th>
<th>Dominates</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>1,4,7,9</td>
</tr>
<tr>
<td>3</td>
<td>None</td>
</tr>
<tr>
<td>4</td>
<td>7,9</td>
</tr>
<tr>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>6</td>
<td>1,5,7,9</td>
</tr>
<tr>
<td>7</td>
<td>None</td>
</tr>
<tr>
<td>8</td>
<td>1,2,4,5,6,7,9</td>
</tr>
<tr>
<td>9</td>
<td>None</td>
</tr>
<tr>
<td>10</td>
<td>1,2,3,4,5,7,9</td>
</tr>
</tbody>
</table>
(c) Show the population distributed over non-dominated levels like some multi-objective EAs employ, after each addition of an element, starting with element 1 and ending with element 10 increasing the element number one at a time; indicate elements with their IDs. So you need to show ten different population distributions, the first one consisting of a single element, and the last one consisting of ten elements. [6]

After adding element 1:

Level 1: 1

After adding element 2:

Level 1: 2
Level 2: 1

After adding element 3:

Level 1: 2,3
Level 2: 1

After adding element 4:

Level 1: 2,3
Level 2: 1,4

After adding element 5:

Level 1: 2,3,5
Level 2: 1,4

After adding element 6:

Level 1: 2,3,6
Level 2: 1,4,5

After adding element 7:

Level 1: 2,3,6
Level 2: 1,4,5
Level 3: 7

After adding element 8:

Level 1: 3,8
Level 2: 2,6
Level 3: 1,4,5
Level 4: 7

After adding element 9:

Level 1: 3,8
Level 2: 2,6
Level 3: 1,4,5
Level 4: 7,9

After adding element 10:

Level 1: 8,10
Level 2: 2,3,6
Level 3: 1,4,5
Level 4: 7,9