CS5401 FS2018 Final Exam Key

This is a closed-book, closed-notes exam. The only items you are allowed to use are writing implements. Mark each sheet of paper you use with your name and the string "cs5401 fs2018 final exam". If 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 75, but note that the max exam score will be capped at 70 (i.e., there are 5 bonus points but you can't score more than 100%). You have exactly two hours to complete this exam. Keep your answers clear and concise while complete. Good luck!

Multiple Choice Questions - write the letter of your choice on your answer paper

1. Is it possible for a haploid EA to be both pleitropic and polygenetic at the same time: [2]
   - (a) no, because this would require a diploid EA [0]
   - (b) no, because the one excludes the other regardless of whether it's a haploid or a diploid EA [0]
   - (c) yes, because the decoder function can be both surjective and injective at the same time [$\frac{1}{2}$]
   - **(d) yes, because the decoder function can be both not surjective and not injective at the same time**
   - (e) none of the above [0]
2. Genetic drift and natural selection: [2]
   - (a) are different terms for the same concept [0] *(false because they are different concepts)*
   - (b) are different non-related concepts [$\frac{1}{2}$] *(false because while different, they are certainly related)*
   - (c) complement each other because natural selection without genetic drift would select based on phenotypes without regard for genotypes [1] *(bad answer because natural selection selects based on phenotype, whether or not there is genetic drift)*
   - **(d) complement each other because genetic drift without natural selection would result in random search**
3. Fitness proportional selection suffers from the following problems: [2]
   - (a) when fitness values are all very close together, mediocre individuals take over the entire population very quickly, leading to premature convergence [1]
   - (b) outstanding individuals cause the selection pressure to drop because they decrease the number of slots on the virtual roulette wheel from which individuals are selected [$\frac{1}{2}$]
   - (c) transposed versions of the fitness function all behave identically while they represent different problems which we obviously want to be able to differentiate between [$\frac{1}{2}$]
   - (d) all of the above [0]
   - (e) **none of the above**
4. Modern Evolutionary Programming (EP) is practically merging with modern Evolution Strategies (ES) in the aspects of: [2]
   - (a) parent selection [0]
   - (b) self-adaptation of mutation step sizes [1]
   - (c) the order in which mutation variables and strategy parameters are updated [1]
   - (d) answers a, b, and c [$1\frac{1}{2}$]
   - (e) answers a and b [$\frac{1}{2}$]
   - (f) answers a and c [$\frac{1}{2}$]
   - (g) **answers b and c**
   - (h) none of the above [0]

5. The current GP practice of strongly limiting the role of mutation in favor of recombination is because: [2]
   (a) recombination tends to increase genetic diversity in GP, unlike mutation which contrary to in standard EAs which employ a linear representation, has a tendency to destroy critical alleles [0]
   (b) **the generally shared view that in GP, crossover has a large shuffling effect, acting in some sense as a macromutation operator**
   (c) mutation tends to cause excessive bloat in GP, unlike recombination which has a natural parsimony pressure effect [0]
   (d) all of the above [$\frac{1}{2}$]
   (e) none of the above [0]

6. Countermeasures to bloat in GP include: [2]
   (a) increasing mutation rate to maintain genetic diversity [0]
   (b) **increasing parsimony pressure to penalize the fitness of large chromosomes**
   (c) reducing the number of alleles to prevent disproportional tree growth [0]
   (d) all of the above [0]
   (e) none of the above [0]

7. The Pitt and Michigan approaches in Learning Classifier Systems differ in that: [2]
   (a) in the Pitt approach each individual has the option of either representing a single rule or a rule set, while in the Michigan approach each individual represents a single rule and the entire population represents the complete rule set [1]
   (b) in the Pitt approach each individual represents a single rule and the entire population represents the complete rule set, while in the Michigan approach each individual has the option of either representing a single rule or a rule set [$\frac{1}{2}$]
   (c) **in the Pitt approach each individual represents a complete rule set, while in the Michigan approach each individual represents a single rule and the entire population represents the complete rule set**
   (d) in the Pitt approach each individual represents a single rule and the entire population represents the complete rule set, while in the Michigan approach each individual represents a complete rule set [1]
   (e) in the Pitt approach each individual represents a complete rule set, while in the Michigan approach each individual has the option of either representing a single rule or a rule set [1]
   (f) none of the above [0]

8. In Fitness Sharing: [2]
   (a) new individuals replace similar population members, resulting in the population sharing the niches equally [0]
   (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**
   (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. The exacerbation of premature convergence in memetic algorithms is due to: [2]
   (a) limited seeding [$\frac{1}{2}$]
   (b) diversity preserving recombination operators [0]
   (c) non-duplicating selection operators [0]
   (d) Boltzmann selection [0]
   (e) all of the above [0]
   (f) **none of the above**

Regular Questions

10. Explain why it does or why it does not make sense to investigate the Baldwin Effect for a Lamarckian Evolutionary Algorithm approach to solving the Light Up Puzzle. [2]

    This does not make sense, because Lamarckian EAs do not exhibit the Baldwin Effect.

11. How are conflicting rules in the action set of a Learning Classifier System resolved? [2]

    This is not applicable because per definition all the rules in the action set advocate the same action.

12. Alice is writing an EA to solve the binary knapsack constraint satisfaction problem. Given the following constraint handling approaches:

    (a) Ignore the constraints under the motto: all is well that ends well.

    (b) Upon generating an infeasible solution, immediately kill it and generate a new solution; repeat this step until a feasible solution is generated.

    (c) Employ a penalty function that reduces the fitness of infeasible solutions, preferably so that the fitness is reduced in proportion to the number of constraints violated, or to the distance from the feasible region.

    (d) Employ a repair function that takes infeasible solutions and "repairs" them by transforming them into a related feasible solution, typically as close as possible to the infeasible one.

    (e) Employ a closed feasible solution space which guarantees that the initial population consists of feasible solutions only and all evolutionary operations on feasible solutions are guaranteed to result in feasible solutions. Typically a combination of custom representation, initialization, recombination, and mutation is employed to achieve this.

    (f) Employ a decoder function that maps genotype space to phenotype space such that the phenotypes are guaranteed to be feasible even when the genotypes are infeasible. Typically this involves mapping multiple different genotypes to the same phenotype.

    Which of these six constraint handling approaches do you recommend Alice employs? Explain your answer! [5]

    There are three cases:

    Case 1 *If the sum of the item costs is smaller or equal to the constraint value, then use the first approach where the constraints are simply ignored.*

    Case 2 *If Alice knows that the ratio of invalid to total solutions is extremely low, for instance if the sum of the item costs barely exceeds the constraint value, then use the second approach where invalid solutions are immediately discarded and use either stochastic survival or a mutation with for instance a Gaussian distributed mutation rate to guarantee global optimum reachability.*

    Case 3 *Otherwise use a high quality decoder function which will guarantee valid solutions while imposing no limitations on the search of the genotype space.*

13. Is the genotypic encoding for the Assignment 2 Series of Pac-Man vs. the Ghosts pleitropic, polygenetic, both, or neither? Explain your answer! [3]

    It is pleitropic and polygenetic, because one gene (function or terminal node in GP tree) can impact multiple phenotypic traits (controller actions in the form of GP tree outputs) which means the genotypic encoding is pleitropic, and one phenotypic trait (controller action) can depend on multiple genes (function or terminal nodes in GP tree) which means the genotypic encoding is polygenetic.

14. Is the genotype-phenotype decoding function for the Assignment 2 Series of Pac-Man vs. the Ghosts surjective, injective, both, or neither? Explain your answer! [3]

    It is surjective but not injective, because all controllers are valid genotypes (surjective), but there exist controllers than can be encoded by multiple distinct genotypes, for instance by swapping two constant terminals being fed into a summation function (not injective).

15. Is the phenotype to fitness mapping for the Assignment 2 Series of Pac-Man vs. the Ghosts surjective, injective, both, or neither? Explain your answer! [3]

    It is surjective but not injective, because potentially all controllers can be represented, and therefore all valid fitness values obtained (surjective), but there exist distinct controllers which obtain the same fitness, for instance given a symmetric scenario they follow a reverse direction strategy (not injective).

16. Given the following two parents with permutation representation:
    $p1 = (475318692)$
    $p2 = (524836971)$

    (a) Compute the first offspring with Cycle Crossover. [4]
    Cycle 1: 4-5, Cycle 2: 7-2-1-3-8-6-9
    Construction of first offspring by scanning parents from left to right, starting at parent 1 and alternating parents:

    i. Add cycle 1 from parent 1: $4 \cdot 5 \cdots\cdots$
    ii. Add cycle 2 from parent 2: 425836971

    (b) Compute the first offspring with PMX, using crossover points between the 2nd and 3rd loci and between the 6th and 7th loci. [5]

    i. $\cdot\cdot 5318 \cdot \cdot\cdot$
    ii. $4 \cdot 5318 \cdot \cdot\cdot$
    iii. $4 \cdot 5318 \cdot \cdot 6$
    iv. 425318976

    (c) Compute the first offspring with Edge Crossover, except that for each random choice you instead select the lowest element. [10]

Original Edge Table:

Element	Edges	Element	Edges
1	3,8,7,5	6	8,9+,3
2	9,4+,5	7	4,5,9,1
3	5,1,8,6	8	1,6,4,3
4	2+,7,8	9	6+,2,7
5	7,3,1,2		

Construction Table:

Element selected	Reason	Partial result
1	Lowest	1
3	Equal list size, so lowest	13
5	Equal list size, so lowest	135
2	Equal list size, so lowest	1352
4	Common edge	13524
7	Equal list size, so lowest	135247
9	Only element	1352479
6	Only element	13524796
8	Last element	**135247968**

Edge Table After Step 1:

Element	Edges	Element	Edges
1	3,8,7,5	6	8,9+,3
2	9,4+,5	7	4,5,9
3	5,8,6	8	6,4,3
4	2+,7,8	9	6+,2,7
5	7,3,2		

Edge Table After Step 2:

Element	Edges	Element	Edges
		6	8,9+
2	9,4+,5	7	4,5,9
3	5,8,6	8	6,4
4	2+,7,8	9	6+,2,7
5	7,2		

Edge Table After Step 3:

Element	Edges	Element	Edges
		6	8,9+
2	9,4+	7	4,9
		8	6,4
4	2+7,8	9	6+,2,7
5	7,2		

Edge Table After Step 4:

Element	Edges	Element	Edges
		6	8,9+
2	9,4+	7	4,9
		8	6,4
4	7,8	9	6+,7

Edge Table After Step 5:

Element	Edges	Element	Edges
		6	8,9+
		7	9
		8	6
4	7,8	9	6+,7

Edge Table After Step 6:

Element	Edges	Element	Edges
		6	8,9+
		7	9
		8	6
		9	6+

Edge Table After Step 7:

Element	Edges	Element	Edges
		6	8
		8	6
		9	6+

(d) Compute the first offspring with Order Crossover, using crossover points between the 3rd and 4th loci and between the 7th and 8th loci. [3]

i. Child 1: · · ·3186 · ·

ii. Child 1: **249318675**

17. Assuming a simple genetic algorithm whose global optimum has a fitness of 100.0 and given the following bit strings v_1 through v_5 and schema S
$v_1 = (01010110011001)$ $fitness(v_1) = 88.0$
$v_2 = (01110110001001)$ $fitness(v_2) = 1.0$
$v_3 = (01110110111001)$ $fitness(v_3) = 1.0$
$v_4 = (11110110011000)$ $fitness(v_4) = 1.0$
$v_5 = (11110110011001)$ $fitness(v_5) = 2.0$
$S = (01 * 10 * 10 * * * 001)$

(a) Compute the *order* of S. [1]
9

(b) Compute the *defining length* of S and show your computation. [1]
14-1=13

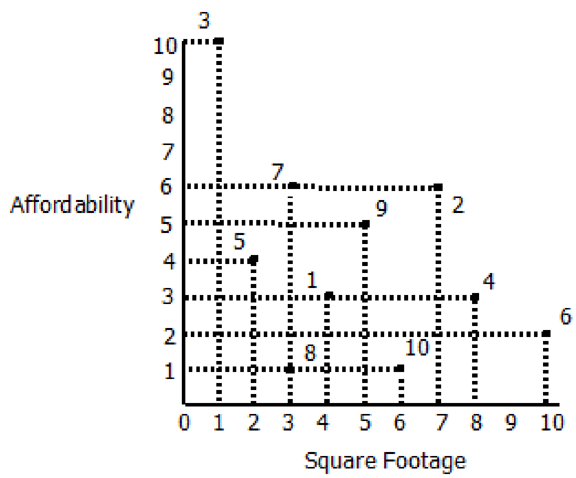
(c) Compute the fitness of S and show your computation. [1]
$\frac{88.0+1.0+1.0}{3} = 30.0$

(d) Do you expect the number of strings matching S to increase or decrease in subsequent generations? Explain your answer! [4]
The average population fitness is $\frac{88.0+1.0+1.0+1.0+2.0}{5} = \frac{93}{5} = 18\frac{3}{5}$. *While S has a higher fitness than the population as a whole, it's low compared to the global optimum and its high order and very high defining length make its survival unlikely, particularly because the Hamming distance between it and low fitness individuals v_2 and v_3 is only 2, so even a single or double bit mutation which still matches it may turn out to be of low fitness. Therefore, it may be expected that the number of strings matching S will decrease in subsequent generations.*

18. Say you want to purchase a new house and care most about maximizing square footage and minimizing price. You collect square footage data and pricing on ten different houses and then you normalize both the square footage data and the pricing which results in the following table, where higher square footage numbers indicate greater square footage and higher pricing numbers indicate better affordability (so lower price):

ID	Square footage	Price
1	4	3
2	7	6
3	1	10
4	8	3
5	2	4
6	10	2
7	3	6
8	3	1
9	5	5
10	6	1

(a) Plot the above table using dotted lines to indicate the area of domination for each element, with square footage on the horizontal axis and affordability on the vertical axis. [2]



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

ID	Dominates
1	8
2	1,5,7,8,9,10
3	None
4	1,8,10
5	None
6	8,10
7	5,8
8	None
9	1,5,8
10	8

(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:** 2,3,4
- **Level 2:** 1

After adding element 5:

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

After adding element 6:

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

After adding element 7:

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

After adding element 8:

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

After adding element 9:

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

After adding element 10:

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