

COMP 5660/6660 Fall 2024 Final Exam Key

This is a closed-book, closed-notes exam. The sum of the max points for all the questions is 126, but note that the max exam score will be capped at 118 (i.e., there are 8 bonus points, but you can't score more than 100%). You have exactly 120 minutes to complete this exam. Keep your answers clear and concise while complete. Good luck!

1. Mutation has the potential to modify an individual's: [4 pts]

- (a) genotype
- (b) phenotype
- (c) alleles
- (d) fitness

Select one of:

- a [1]
- b [1]
- c [1]
- d [1]
- a and b [2]
- a and c [2]
- a and d [2]
- b and c [2]
- b and d [2]
- a, b, and c [3]
- a, b, and d [3]
- a, c, and d [3]
- b, c, and d [3]
- **a, b, c, and d**
- none of a, b, c, nor d [0]

2. In an EA which utilizes truncation survival selection: [4 pts]

- (a) the chance of premature convergence is lower than other elitist EAs (*false because truncation survival has the highest selective pressure of all the regular elitist survival mechanisms*)
- (b) the parent selection must not be elitist because that would cause premature convergence (*elitist means that the fittest solution is guaranteed to survive so doesn't apply to parent selection and also doesn't necessarily cause premature convergence*)
- (c) the parent selection should be stochastic to decrease the chance of premature convergence

- a [1]
- b [1]
- **c**
- a and b [0]
- a and c [2]
- b and c [2]
- a, b, and c [1]
- none of a, b, nor c [0]

3. Each amino acid can be coded for by several codons in your DNA — histidine, for example, is coded for by CAT and CAC. All 20 amino acids can be produced by *at least* two unique sets of codons, and all possible codon trios code for *something*. Therefore, the mapping from DNA codon trio to amino acids is: [4 pts]

- (a) injective
- (b) surjective
- (c) bijective

Select one of:

- a [0]
- **b**; This answer is correct, as many DNA sequences can map to the same amino acid sequence (so not injective and hence not bijective), and all possible amino acid sequences are inherently mapped to by numerous DNA sequences (so at least by one and hence surjective).
- c [2]
- none of a, b, nor c [0]

4. Genetic drift is caused by: [4 pts]

- (a) finite population size
- (b) uncorrelated self-adaptive mutation
- (c) panmictic mixing
- (d) stochastic survival selection

Select one of:

- a [2]
- b [0]
- c [2]
- d [0]
- a and b [1]
- **a and c**
- a and d [1]
- b and c [1]
- b and d [0]
- c and d [1]
- a, b, and c [3]
- a, b, and d [1]
- a, c, and d [3]
- b, c, and d [1]
- a, b, c, and d [2]
- none of a, b, c, nor d [0]

5. The ramped half-and-half method is the most common technique in GP for: [4 pts]

- (a) initialization
- (b) parent selection
- (c) survival selection
- (d) termination

Select one of:

- **a**
- b [0]
- c [0]
- d [0]
- none of a, b, c, nor d [0]

6. In Diffusion Model EAs: [4 pts]

- (a) individuals are modeled by diffusion equations and only panmictic mating is permitted
- (b) individuals are modeled by diffusion equations and mating is restricted to demes
- (c) the population is conceptually distributed on a grid and only panmictic mating is permitted
- (d) the population is conceptually distributed on a grid and mating is restricted to demes

Select one of:

- a [0]
- b [2]
- c [2]
- **d**
- a and c [1]
- b and d [2]
- none of a, b, c, nor d [0]

7. A hyper-heuristic is: [4 pts]

- (a) a metaheuristic which searches algorithm space employing algorithmic primitives
- (b) a type of Genetic Programming to automate the design of algorithms employing a Turing complete set of primitives (*it can be a different type of metaheuristic than GP and typically should use higher order primitives than a Turing complete set to avoid an infeasibly large search space*)
- (c) a type of EA which employs algorithmic primitives extracted typically from existing algorithms to automate the design of algorithms (*it can be a different type of metaheuristic than an EA*)

Select one of:

- **a**
- b [2]
- c [3]
- a and b [3]
- a and c [3]
- b and c [2]
- a, b, and c [2]

- none of a, b, nor c [0]

8. Hyper-heuristics are particularly well suited for: [4 pts]

- (a) Sequential EAs
- (b) Synchronous Parallel EAs
- (c) Asynchronous Parallel EAs (**because hyper-heuristics are computationally expensive (so are particularly well suited for parallel computing) and tend to exhibit heterogeneous execution times (so synchrony may be expected to cause excessive idling)**)

Select one of:

- a [1]
- b [2]
- c
- a and b [1]
- a and c [1]
- b and c [2]
- a, b, and c [1]
- none of a, b, nor c [0]

9. Pittsburgh-style LCS: [4 pts]

- (a) predates but is similar to GP in that each individual represents a complete model mapping input to output spaces
- (b) each gene typically represents a rule
- (c) tends to outperform Michigan-style LCS given sufficient computing resources and effective parsimony methods
- (d) suffers from bloat similar to GP

Select one of:

- a [1]
- b [1]
- c [1]
- d [1]
- a and b [2]
- a and c [2]
- a and d [2]
- b and c [2]
- b and d [2]
- c and d [2]
- a, b, and c [3]
- a, b, and d [3]
- a, c, and d [3]
- b, c, and d [3]
- **a, b, c, and d**
- none of a, b, c, nor d [0]

10. “Intelligent” initialization in a memetic algorithm can be performed by: [4 pts]

- (a) Seeding
- (b) Selective Initialization
- (c) Locally Optimized Random Initialization
- (d) Mass Mutation

Select one of:

- a [1]
- b [1]
- c [1]
- d [1]
- a and b [2]
- a and c [2]
- a and d [2]
- b and c [2]
- b and d [2]
- c and d [2]
- a, b, and c [3]
- a, b, and d [3]
- a, c, and d [3]
- b, c, and d [3]
- **a, b, c, and d**
- none of a, b, c, nor d [0]

11. On a computer system with 400 computing cores and given a population size of 200 and an offspring size of 300, employing an Asynchronous Parallel EA (APEA) for evolving GP controllers for Pac-Man: [4 pts]

- (a) may be expected to reduce run-time versus a Synchronous Parallel EA (SPEA) because a SPEA cannot utilize more cores than the offspring size while an APEA can
- (b) may be expected to increase run-time versus a SPEA because an APEA cannot utilize more cores than the population size while a SPEA can
- (c) may be expected to reduce run-time versus a SPEA because a SPEA has to wait for the longest evaluation to complete while an APEA can exploit the heterogeneous evaluation times common to GP

Select one of:

- a [2]
- b [0]
- c [2]
- a and b [1]
- **a and c**
- b and c [1]
- a, b, and c [2]
- none of a, b, nor c [0]

12. Disengagement in a two-population competitive CoEA occurs when: [4 pts]

- (a) the individuals in both populations stop competing and start collaborating
- (b) both populations get stuck in local optimums leading to a loss of search gradient
- (c) one population gets stuck in a local optimum and the other population stops evolving because of a loss of evolutionary pressure

Select one of:

- a [0]
- b [0]
- c [2]
- a and b [0]
- a and c [1]
- b and c [1]
- all of a, b, and c [0]
- **none of a, b, nor c**

13. In neuro-evolution – evolving artificial neural networks (ANNs) –: [4 pts]

- (a) the high computational cost of evolution can be amortized over sufficient repeated uses of the resulting ANN
- (b) the high computational cost of evolution is incurred a priori, with the resulting ANN run in production/real-time
- (c) only the hyperparameters (e.g., learning rate, number of layers) can be evolved, not the weights (which the ANN has to learn) and topology/structure (which a human has to design)
- (d) Lamarckian evolution is not supported because the genotype-to-phenotype mapping is per definition highly complex **that's true for generative neuro-evolution, but not in general**

Select one of:

- a [2]
- b [2]
- c [0]
- d [1]
- **a and b**
- a and c [1]
- a and d [2]
- b and c [1]
- b and d [2]
- c and d [0]
- a, b, and c [3]
- a, b, and d [3]
- a, c, and d [1]
- b, c, and d [1]
- all of a, b, c, and d [2]
- none of a, b, c, nor d [0]

14. Some of the advantages of Interactive EAs are: [4 pts]

- (a) handling situations with no clear fitness function
- (b) handling situations with changeable objectives and preferences
- (c) improved search ability through the user changing their guiding principle
- (d) increased exploration and diversity through direct user manipulation of the system

Select one of:

- a [1]
- b [1]
- c [1]
- d [1]
- a and b [2]
- a and c [2]
- a and d [2]
- b and c [2]
- b and d [2]
- c and d [2]
- a, b, and c [3]
- a, b, and d [3]
- a, c, and d [3]
- b, c, and d [3]
- **a, b, c, and d**
- none of a, b, c, nor d [0]

15. To battle the “human bottleneck” in interactive evolution, one can employ: [4 pts]

- (a) surrogate fitness functions
- (b) small population sizes
- (c) multi-objective EAs for problems that exhibit a mixture of quantitative and qualitative aspects
- (d) crowdsourcing

Select one of:

- a [1]
- b [1]
- c [1]
- d [1]
- a and b [2]
- a and c [2]
- a and d [2]
- b and c [2]
- b and d [2]
- c and d [2]
- a, b, and c [3]
- a, b, and d [3]
- a, c, and d [3]
- b, c, and d [3]
- **a, b, c, and d**
- none of a, b, c, nor d [0]

16. In the automated design of EAs, encoding the evolutionary process as a directed graph is preferable compared to Koza-style GP trees, because it: [4 pts]

- (a) is easier to implement
- (b) has a larger space of representable EAs
- (c) can represent “traditional” evolutionary cycles, while trees cannot
- (d) is a more natural representation for evolutionary cycles

Select one of:

- a [0]
- b [2]
- c [0]
- d [2]
- a and b [1]
- a and c [0]
- a and d [1]
- b and c [1]
- **b and d**
- c and d [1]
- a, b, and c [1]
- a, b, and d [3]
- a, c, and d [1]
- b, c, and d [3]
- a, b, c, and d [2]
- none of a, b, c, nor d [0]

17. Say for the cutting stock problem, you want to simultaneously minimize stock length and width. You execute a multi-objective EA and the final population contains the solutions listed in the following table, where higher length efficiency indicates shorter stock length, and higher width efficiency indicates shorter stock width:

ID	Length efficiency	Width Efficiency
1	7	4
2	2	10
3	8	5
4	1	7
5	10	4
6	2	7
7	10	1
8	1	10
9	9	6
10	3	5

- (a) List for each element which elements it dominates; indicate elements with their IDs. [4 pts]

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

- (b) 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. [12 pts]

After adding element 1:

Level 1: 1

After adding element 2:

Level 1: 1,2

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: 2,3,5

Level 2: 1,6

Level 3: 4

After adding element 7:

Level 1: 2,3,5

Level 2: 1,6,7

Level 3: 4

After adding element 8:

Level 1: 2,3,5

Level 2: 1,6,7,8

Level 3: 4

After adding element 9:

Level 1: 2,5,9

Level 2: 3,6,7,8

Level 3: 1,4

After adding element 10:

Level 1: 2,5,9

Level 2: 3,6,7,8

Level 3: 1,4,10

18. Given the following two parents with permutation representation:

$p1 = (792834615)$

$p2 = (315426978)$

compute the first offspring with Order Crossover, using crossover points between the 2nd and 3rd loci and between the 7th and 8th loci. Show your offspring construction steps. [4 pts]

(a) $\dots 28346 \dots$

(b) 592834671

19. Given the following two parents with permutation representation:

$p1 = (792834615)$

$p2 = (315426978)$

compute the first offspring with Cycle Crossover. Show first the cycles you've identified and then the construction of the offspring. [6 pts]

Cycle 1: 7-3-2-5-8-4-6-9-1

Construction of first offspring by scanning parents from left to right, starting at parent 1 and alternating parents:

(a) Add cycle 1 from parent 1: 792834615

20. Given the following two parents with permutation representation:

$p1 = (792834615)$

$p2 = (315426978)$

compute the first offspring with PMX, using crossover points between the 2nd and 3rd loci and between the 7th and 8th loci. Show your offspring construction steps. [8 pts]

(a) $\dots 28346 \dots$

(b) $5 \cdot 28346 \dots$

(c) $5 \cdot 28346 \cdot 9$

(d) 512834679

21. Given the following parents with permutation representation:

$p1 = (792834615)$

$p2 = (315426978)$

compute the first offspring with Edge Crossover, except that for each random choice you instead select

the lowest element. Show how you arrived at your answer by filling the following templates: [12 pts]

Edge Table:

Element	Edges
---------	-------

Construction Table:

Element Selected	Reason Selected	Partial Result
------------------	-----------------	----------------

Original Edge Table:

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

Construction Table:

Element selected	Reason	Partial result
1	Lowest	1
5	Common edge	1,5
7	Shortest list size	1,5,7
9	Common edge	1,5,7,9
6	Shortest list size	1,5,7,9,6
2	Equal list size, so lowest	1,5,7,9,6,2
4	Equal list size, so lowest	1,5,7,9,6,2,4
3	Only element	1,5,7,9,6,2,4,3
8	Last element	1,5,7,9,6,2,4,3,8

Edge Table After Step 1:

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

Edge Table After Step 2:

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

Edge Table After Step 3:

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

Edge Table After Step 4:

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

Edge Table After Step 5:

Element	Edges	Element	Edges
		6	4,2
2	8,4		
3	8+,4	8	2,3+
4	3,2		

Edge Table After Step 6:

Element	Edges	Element	Edges
2	8,4		
3	8+,4	8	3+
4	3		

Edge Table After Step 7:

Element	Edges	Element	Edges
3	8+	8	3+
4	3		

Edge Table After Step 8:

Element	Edges	Element	Edges
3	8+	8	

22. The n -bit multiplexer function consists of k address bits a_i followed by 2^k data bits d_j where $n = k + 2^k$ and the function is defined as $a_{k-1}, \dots, a_1, a_0, d_{2^k-1}, \dots, d_1, d_0$. Assume a Michigan-style Learning Classifier System (LCS) to solve a 6-bit multiplexer problem with the following rule set:

Rule 1: 101### : 1 → 60

Rule 2: 0#1110 : 1 → 50

Rule 3: #01##0 : 0 → 40

Rule 4: ##1##0 : 1 → 30

Rule 5: #11010 : 1 → 20

Rule 6: 10#0#0 : 0 → 10

If the input string 100010 is presented to this LCS:

- (a) which rules will the match set consist of? [2 pts]

Rule 6

- (b) which rules will the action set consist of and what action will the LCS execute? Show how you computed this. [4 pts]

Group them by advocated action and compute predicted action payoff:

Action 0: Rule 6: Mean predicted action payoff: 10/1=10

(Action 1: no matching rules) Highest predicted payoff action: Action 0

Action set: Rule 6

LCS executes Action 0

23. Given the following bit strings v_1 through v_5 and schema S

$v_1 = (11101110111101)$ $fitness(v_1) = 0.3$

$v_2 = (10110010001101)$ $fitness(v_2) = 0.1$

$v_3 = (00001010011010)$ $fitness(v_3) = 1.0$

$v_4 = (01001110111001)$ $fitness(v_4) = 1.9$

$v_5 = (11001011110101)$ $fitness(v_5) = 1.7$

$S = (00000001111111)$

- (a) Compute the *order* of S . [2 pts]

14

- (b) Compute the *defining length* of S and show your computation. [2 pts]

14-1=13

- (c) Compute the fitness of S and justify your answer. [2 pts]

Undefined because S doesn't match any of the given strings.

- (d) Do you expect the number of strings matching S to increase or decrease in subsequent generations? Explain your answer! [4 pts]

Because S currently doesn't match any strings and eventually may match strings after sufficient recombination and mutation has taken place, the number of strings matching S is expected to eventually increase.