CS5401 FS2016 Exam 1 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 "cs5401fs2016 exam1". 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 55, but note that the max exam score will be capped at 53 (i.e., there are 2 bonus points, but you can't score more than 100%). You have exactly 75 minutes 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. In an arbitrary unimodal problem there are: [2]
   - (a) no local optima, but multiple global optima *(false as every global optimum is also a local optimum)* [0]
   - (b) **a single local optimum and a single global optimum**
   - (c) multiple local optima but only one global optimum *(false because per definition a unimodal problem has a single local optimum)* [$\frac{1}{2}$]
   - (d) none of the above [0]
2. Mutation has the potential to modify an individual's: [2]
   - (a) genotype [$\frac{1}{2}$]
   - (b) phenotype [$\frac{1}{2}$]
   - (c) alleles [$\frac{1}{2}$]
   - (d) fitness [$\frac{1}{2}$]
   - **(e) all of the above**
   - (f) a, b, and c, but not d [1]
3. A Hamming cliff is: [2]
   - (a) **A pair of binary strings which differ in many of their bits *(i.e., have a large Hamming distance)* exhibit a Hamming cliff**
   - (b) a population of individuals with binary representation whose genotypes are very similar and who all have a large Hamming distance to the global optimum causing premature convergence [$\frac{1}{2}$]
   - (c) a cliff-like plot associated with a binary encoded fitness function belonging to an NP-Complete problem class [0]
   - (d) all of the above [0]
   - (e) none of the above [0]
4. Binary gray code is: [2]
   - (a) a binary coding system specifically developed to represent gray scales in digital images [$\frac{1}{2}$]
   - (b) a binary coding system with consecutive corresponding integers differing by no more than 2 bits [$1\frac{1}{2}$]
   - (c) a binary coding system where the Hamming distance between consecutive corresponding integers is proportional to their magnitude [1]
   - (d) all of the above [0]
   - (e) **none of the above**

5. 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**

6. In an EA which utilizes truncation survival selection: [1]
   (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)* [0]
   (b) the parent selection should be stochastic to decrease the chance of premature convergence
   (c) the parent selection may not be elitist because that would cause premature convergence *(elitist means that the fittest solution is guaranteed to survive which by itself doesn't cause premature convergence)* [1]
   (d) all of the above [1]
   (e) none of the above [0]

7. To increase selective pressure for an EA employing tournament parent selection one can: [1]
   (a) switch from truncation survivor selection (i.e., deterministically replacing the worst individuals) to an elitist stochastic survivor selection [0]
   (b) decrease the tournament size used in parent selection [0]
   (c) both of the above [0]
   (d) none of the above

8. One advantage of implementing survivor selection by employing a so-called reverse k-tournament selection to select who dies is that: [2]
   (a) you guarantee (k)-elitism [1]
   (b) **you guarantee** $(k-1)$**-elitism**
   (c) the probability of surviving is proportional to your fitness rank [0]
   (d) you guarantee 1-elitism (i.e., the fittest individual is guaranteed to survive) [$\frac{1}{2}$]
   (e) none of the above [0]

9. Parameter Control is important in EAs because: [2]
   (a) it may somewhat relieve users from parameter tuning as parameter control may make an EA less sensitive to initial strategy parameter values [1]
   (b) optimal strategy parameter values may change during evolution [1]
   (c) **all of the above**
   (d) none of the above [0]

10. "Blind Parameter Control" is a better name for the class of parameter control mechanisms named "Deterministic Parameter Control" in the textbook because that class: [2]
    (a) includes stochastic mechanisms [1]
    (b) does not use any feedback from the evolutionary process [1]
    (c) **all of the above**
    (d) none of the above [0]
11. 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]
12. Would you expect in general a change in performance of an EA if you change λ but maintain the same total number of fitness evaluations by making a compensatory change in the number of generations? [2]
    (a) no, because the EA's performance depends on its representation (genotype) and its fitness function (environment/problem) which together determine its phenotype (expression of the genotype in a given environment) [0]
    (b) yes, because increasing λ will decrease the generational gap and therefore lead to a better exploitation of the genetic knowledge encoded in the current population [0]
    (c) **yes, because this will change the ratio of exploitation versus exploration**
    (d) no, because as long as the total number of fitness evaluations remains constant, the total number of recombinations as well as the total number of mutations remains per definition constant too and therefore the EA's performance remains unchanged [$\frac{1}{2}$]
    (e) none of the above [0]

The following are some regular questions:

13. (a) What is the binary gray code for the standard binary number 0001101100? [2]
    0001011010
    (b) What is the standard binary number encoded by the binary gray code 000000101? [2]
    000000110
14. Given the following two parents with permutation representation:
    $p1 = (546139278)$
    $p2 = (342957816)$
    (a) Compute the first offspring with Cycle Crossover. [4]
    Cycle 1: 5-3, Cycle 2: 4, Cycle 3: 6-2-8, Cycle 4: 1-9-7
    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: $5 \cdots 3 \cdots \cdot$
        ii. Add cycle 2 from parent 2: $54 \cdot \cdot 3 \cdots \cdot$
        iii. Add cycle 3 from parent 1: $546 \cdot 3 \cdot 2 \cdot 8$
        iv. Add cycle 4 from parent 2: **546937218**

(b) Compute the first offspring with PMX, using crossover points between the 1st and 2nd loci and between the 5th and 6th loci. [5]

i. ·4613 · · · ·
ii. ·4613 · · · 2
iii. ·4613 · ·92
iv. 54613 · ·92
v. **546137892**

(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	6+,3,8	6	4,1+,3
2	9+,7,4	7	2,8+,5
3	1,9,6,4	8	7+,5,1
4	5,6,3,2	9	3,2+,5
5	8,4,9,7		

Construction Table:

Element selected	Reason	Partial result
1	Lowest	1
6	Common edge	16
3	Shortest list size	163
4	Equal list size, so lowest	1634
2	Shortest list size	16342
9	Common edge	163429
5	Only element	1634295
7	Equal list size, so lowest	16342957
8	Last element	**163429578**

Edge Table After Step 1:

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

Edge Table After Step 2:

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

Edge Table After Step 3:

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

Edge Table After Step 4:

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

Edge Table After Step 5:

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

Edge Table After Step 6:

Element	Edges	Element	Edges
		7	8+,5
		8	7+,5
		9	5
5	8,7		

Edge Table After Step 7:

Element	Edges	Element	Edges
		7	8+
		8	7+
5	8,7		

(d) Compute the first offspring with Order Crossover, using crossover points between the 4th and 5th loci and between the 7th and 8th loci. [3]

i. Child 1: $\cdot\cdot\cdot\cdot 392 \cdot\cdot$

ii. Child 1: **457839216**

15. Show how one can configure an EA to be functionally equivalent to Simulated Annealing (SA) by copying and filling out the provided table on your answer paper, or, if you believe it is inherently impossible, explain why. [5]

For your convenience, here is some SA pseudo-code:

function SimulatedAnnealing($problem, schedule$) **returns** a solution state
  $current \leftarrow MakeNode(InitialState[problem])$
  for $t \leftarrow 1$ to ∞ **do**
    $T \leftarrow schedule[t]$
    if $T = 0$ **then**
      return current
    end if
    $next \leftarrow$ a randomly selected successor of $current$
    $\Delta E \leftarrow Value[next] - Value[current]$
    if $\Delta E > 0$ **then**
      $current \leftarrow next$
    else
      $current \leftarrow next$ with probability $e^{\Delta E/T}$
    end if
  end for

An EA can be configured to be functionally equivalent to SA as follows:

μ	*1*
λ	*1*
Initialization	*the single pop member is uniform randomly selected* *t=1*
Parent selection	*the lone population member is always selected to be the parent*
Recombination rate	*0%*
Mutation rate	*100%*
Mutation	*offspring = random successor of the parent*
Survival selection type	$(\mu + \lambda)$
Survival selection	*fitness F[] is defined to be equal to Value[]* *T=schedule[t]* *IF* $F[offspring] - F[parent] > 0$ *THEN offspring survives* *ELSE offspring survives with probability* $e^{\frac{F[offspring]-F[parent]}{T}}$ *t++*
Termination condition	*T==0*