Cairo University
Faculty of Computers and Artificial Intelligence
Midterm Exam

Department: Computer Science
Course Name: Soft Computing **Date: 7/12/2020**
Course Code: CS464 **Duration: 1 hour**
Instructor(s): Sabah Sayed **Pages: 4**
Name: .... **ID: ....** **Total Marks: /150**

Answer all questions

Question 1: ***Genetic Algorithm [100 marks]***

Assume we have the function $f(x) = x^3 - 60 * x^2 + 900 * x + 100$
where x is constrained to [0 .. 63]. We want to maximize f(x) (**the optimal is x=10**).
Using a binary representation, x can be represented using **6** binary digits.

a) Given the following four chromosomes give the values for x and f(x).
1 marks for each x 2 marks for each f(x) → 12 marks

Chromosome	Binary String	X	f(x)
C_1	011100	28	212
C_2	001111	15	3475
C_3	010111	23	1227
C_4	000100	4	2804

b) Apply Roulette Wheel Selection, What is the selection probability for chromosomes in (a)?
2 marks for total f(x) 2.5 marks for each selection probability → 12 marks

Chromosome	Binary String	f(x)	selection probability
C_1	011100	212	212/ 7718 = 0.027
C_2	001111	3475	3475/ 7718 = 0.45
C_3	010111	1227	1227/ 7718 = 0.158
C_4	000100	2804	2804/ 7718 = 0.363
		Total : 7718	

c) Apply uniform crossover on C_2 and C_4 according to the template BAABAB.
Parent A(C2): 001111 Offspring A: 001110 → 5 marks
Parent B(C4): 000100 Offspring B: 000101 → 5 marks

d) Apply two points crossover on C_1 and C_3 where the crossover points are 1,4.
Parent A(C1): 0|111|00 Offspring A: 010100 → 5 marks
Parent B(C3): 0|101|11 Offspring B: 011111 → 5 marks

e) Show the population after applying the Generational replacement strategy, Has the overall fitness improved? Show how? → 21 marks

Yes improved 4 marks

1 marks for each binary string

1 marks for each X

1 marks for each current generation f(x)

5 marks for Total fitness or Avg fitness or Max fitness (it is enough to calculate the total fitness only or the max fitness only or the average fitness only)

Chromosome	**Binary String**	**X**	**Current generation f(x)**	**Previous generation f(x)**
C_1	010100	20	2100	212
C_2	001110	14	3684	3475
C_3	011111	31	131	1227
C_4	000101	5	3225	2804
			Total : 9140 Avg fitness: 2285 Max fitness: 3684	Total : 7718 Avg fitness: 1929.5 Max fitness: 3475

f) Assume the initial population was x={17, 21, 4, 28}, Using one-point crossover, what is the probability of finding the optimal solution? Explain your reasons. → 15 marks

The probability is zero 5 marks

If we look at the values in binary we get

x	**Binary**
17	010001
21	010101
4	000100
28	011100

We know that the optimal solution is x = 10 which, in binary is 001010.
You can see that we need a 1 in positions 2 and 4 (counting from the). In the initial population there is no individual with a 1 in position 2. This means that no matter how many times we apply single point crossover we will never be able to find the optimal solution. (10 marks for the reason)

g) In the Island GA, One extra operator is added. What is this operator? Mention the four properties should be specified in its policy? → 20 marks

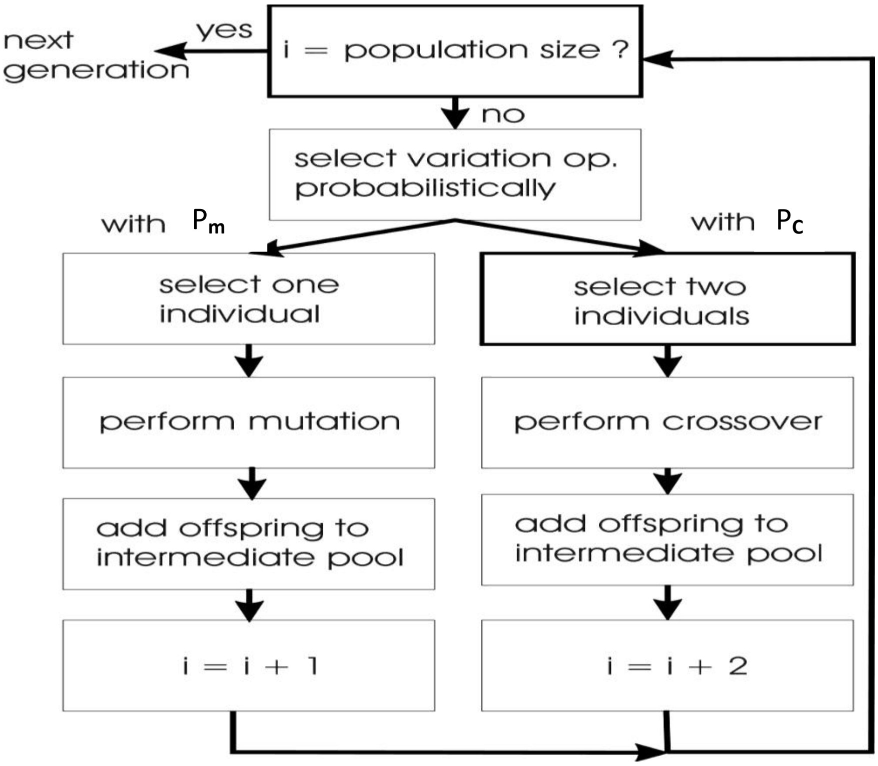
Migration 4 marks

Migration policies specify: 4 marks for each point

1- A communications topology, which determines the migration paths between islands
2- A migration rate, which determines the frequency of migration
3- A selection mechanism, to decide which individuals will migrate
4- A replacement strategy, to decide which individual of the destination island will be replaced

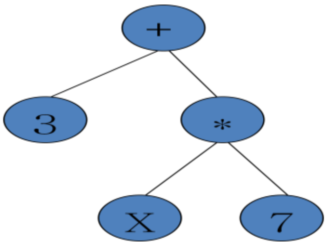
<u>Question 2:</u> ***Genetic Programming [20 marks]***

a) *[10 marks]* Write a basic flowchart for Genetic Programming.



b) *[10 marks]* A program uses genetic programming for solving a problem where Function Set = { + , - , /, * , %} and Terminal Set = {X, Y, Z, Integers}

i. Show an example individual in the population.
Or any individual using function and terminal sets elements



5 marks

ii. What is the genotype space in this problem?
{ + , - , /, * , %, X, Y, Z, Integers} 5 marks

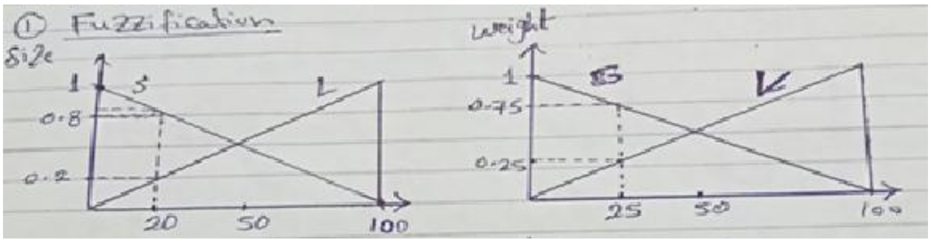
<u>Question 3:</u> ***Fuzzy Logic [30 marks]***

Consider a problem with two input variables, **size** and **weight**, and one output variable, **quality,** with the following fuzzy sets:
size: small **S** {0, 0, 100}, large **L** {0, 100, 100} in range [0 .. 100]
weight: light **G** {0, 0, 100}, Heavy **V** {0, 100, 100} in range [0 .. 100]
quality: bad **B** {0, 0, 5}, medium **M** {0, 5, 10}, good **G** {5, 10, 10} in range [0 .. 10]
<u>The rule base:</u>

R1: if size is S and weight is G then quality is B
R2: if size is S and weight is V then quality is M
R3: if size is L and weight is G then quality is M
R4: if size is L and weight is V then quality is G

Find the crisp value of quality given size= 20 and weight =25

Step 1: Fuzzification: (total 10 marks)



[2 marks]

size = 20
small fuzzy set: Line 1: point1 (0,1) , point2 (100,0)
slope (a) = -1/100
y = -x/100 + b → b =1
Line1 eqn.: y = -x/100 + 1 [2 marks]
substitute by size = 20 → μ_S(size=20) = 0.8 [1 mark]

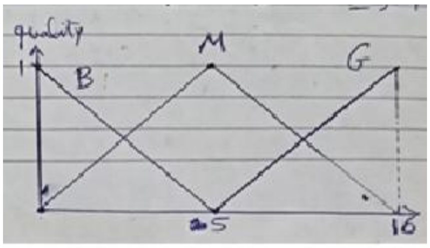
large fuzzy set: Line 1: point1 (0,0) , point2 (100,1)
slope = 1/100
y = x/100 + b → b =0
Line2 eqn.: y = x/100 [2 marks]
substitute by size = 20 → μ_L(size=20) = 0.2 [1 mark]

weight = 2.5
same fuzzy sets, so same line1 & line2 equations
Line1 eqn.: y = -x/100 + 1
substitute by weight = 25 → μ_G(weight=25) = 0.75 [1 mark]
Line2 eqn.: y = x/100
substitute by weight = 25 → μ_V(weight=25) = 0.25 [1 mark]

Step 2: Inference: [total 10 mark]

R1: if (0.8 Λ 0.75) → 0.75 [2 mark] μ_B(quality) [0.5 mark]
R2: if (0.8 Λ 0.25) → 0.25 [2 mark] μ_m(quality) [0.5 mark]
R3: if (0.2 Λ 0.75) → 0.2 [2 mark] μ_m(quality) [0.5 mark]
R4: if (0.2 Λ 0.25) → 0.2 [2 mark] μ_G(quality) [0.5 mark]

Step 3: Defuzzification: [total 10 mark]



[2 mark]

centroid(bad) = (0+0+5)/3 = 1.67 [1 mark]
centroid(medium) = (0+5+10)/3 = 5 [1 mark]
centroid(good) = (5+10+10)/3 = 8.33 [1 mark]

Note: it is also valid if student used the 0.2 only or 0.25 only for μ_m in defuzzification step

Z* = (0.75*1.67 + 0.25*5+ 0.2*5 + 0.2*8.33)/(0.75+0.25+0.2+0.2) [4 mark -> 1 mark for each term in numerator and I mark for denominator]

= 1.2525 + 1.25 + 1 + 1.666 / 1.4

= 3.69 ≃ 3.7 [1 mark]