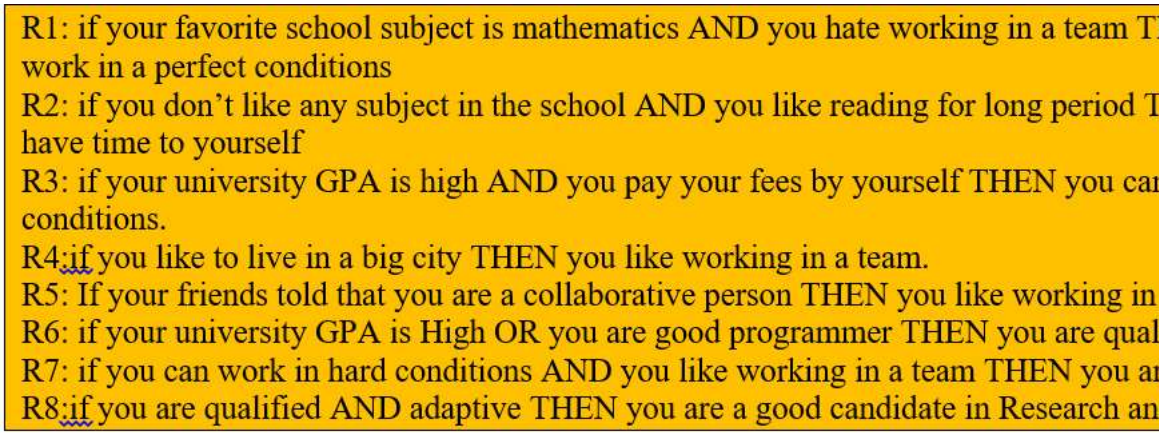
	رهف العابد RAHAF AL- ABED
Started on	Thursday, 22 April 2021, 3:30 PM
State	Finished
Completed on	Thursday, 22 April 2021, 4:30 PM
Time taken	1 hour
Grade	**30.00** out of 30.00 (**100**%)

Question 1

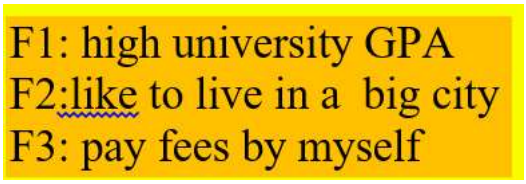
Complete

Mark 7.00 out of 7.00

The following IF-THEN rules are a part from proposed rule-based expert system for job type recommendation :



For the hypothesis "you like working in a team " outline how this could be proved through backward chaining. Assume that current facts include (F1,F2,F3)(7 points)



Goal: you like working in team, R4->fired (achieved), Add F4: you like working in a team

We go through all the rules starting from R1 and look at the THEN part. We reach R4 where the THEN part is our goal. We look at the IF part; "like to live in a big city" is in the facts (F2) so we fire it and it is achieved. We reach to the conclusion of the hypothesis "you like working in a team" being proved.

Comment:

Response history

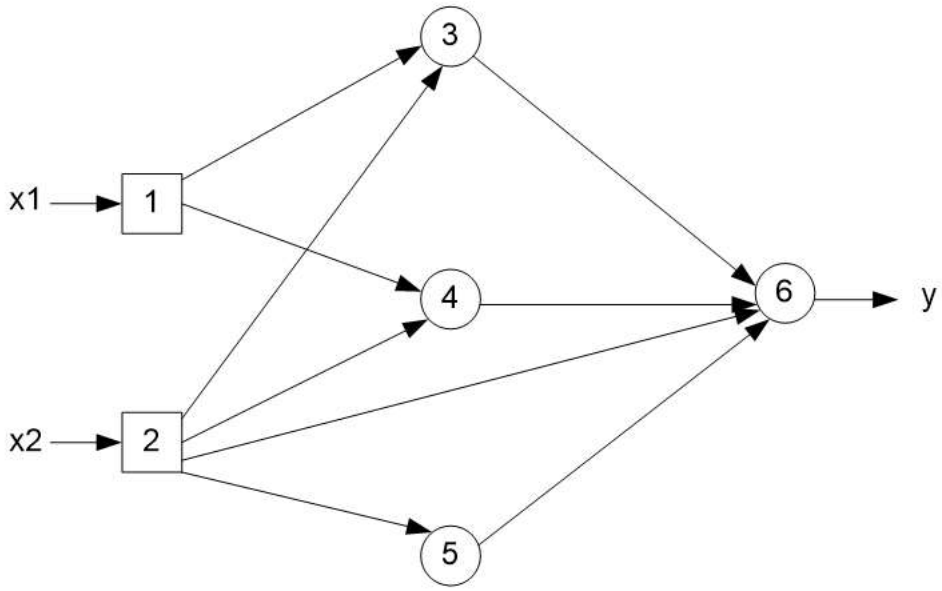
Step	Time	Action	State	Marks
1	22/04/21, 15:30	Started	Not yet answered	
2	22/04/21, 15:40	Saved: Goal: you like working in team, R4->fired (achieved), Add F4: you like working in a team We go through all the rules starting from R1 and look at the THEN part. We reach R4 where the THEN part is our goal. We look at the IF part; "like to live in a big city" is in the facts (F2) so we fire it and it is achieved. We reach to the conclusion of the hypothesis "you like working in a team" being proved.	Answer saved	
3	22/04/21, 16:30	Attempt finished	Complete	
4	**30/04/21, 16:32**	**Manually graded 7 with comment:**	**Complete**	**7.00**

Question 2

Complete

Mark 8.00 out of 8.00

Given is the following neural network with sigmoid activation function in all layers . (8points)



Given w36 =0.1/ w46=0.2/ w56=0.7 and the error gradient in y6= 0.3

y3=0.5 , learning rate = 0.1 x1 = 1,x2= -1

Calculate

The update in the Wight of w13

error gradient in y3 = y3 * (1-y3) * error gradient in y6 * w36 = 0.5(1-0.5)(0.3)(0.1) = 0.0075

delta w13 = x1 * learning rate * error gradient in y3 = 1*0.1*0.0075 = 0.00075

updated w13 = w13 + delta w13 = w13 + 0.00075

Comment:

Response history

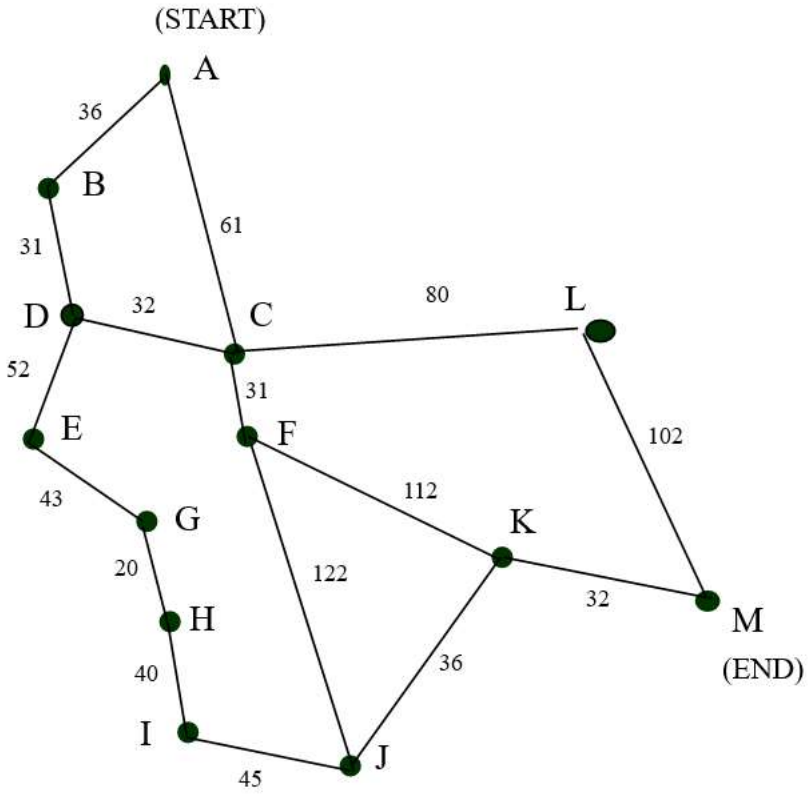
Step	Time	Action	State	Marks
1	22/04/21, 15:30	Started	Not yet answered	
2	22/04/21, 15:52	Saved: error gradient in y3 = y3 * (1-y3) * error gradient in y6 * w36 = 0.5(1-0.5)(0.3)(0.1) = 0.0075 delta w13 = x1 * learning rate * error gradient in y3 = 1*0.1*0.0075 = 0.00075 updated w13 = w13 + delta w13 = + 0.00075 =	Answer saved	
3	22/04/21, 16:30	Saved: error gradient in y3 = y3 * (1-y3) * error gradient in y6 * w36 = 0.5(1-0.5)(0.3)(0.1) = 0.0075 delta w13 = x1 * learning rate * error gradient in y3 = 1*0.1*0.0075 = 0.00075 updated w13 = w13 + delta w13 = w13 + 0.00075	Answer saved	
4	22/04/21, 16:30	Attempt finished	Complete	
5	**30/04/21, 16:18**	**Manually graded 8 with comment:**	**Complete**	**8.00**

Question 3

Complete

Mark 9.00 out of 9.00

Consider the following map.



Using the A* algorithm work out a route from town A to town M. Use the following cost functions.

In your answer provide the following:

1. State the order in which the nodes were expanded (3 marks)

2. State the route that is taken, and give the total cost (3 mark)

3. Explain the relationship between the A* algorithm and the Uniform Cost Search algorithm. (3marks)

G(n) = The cost of each move as the distance between each town (shown on map).

H(n) =heuristic to the target are given in the table below.

H(n):

A	223		E	165		I	100		M	0
B	222		F	136		J	60			
C	166		G	122		K	32			
D	192		H	111		L	102			

1.A

227AC, 258AB

228 ACF, 243 ACL, 258 AB, 285 ACD

236 ACFK, 243 ACL, 258 AB, 285 ACD

236ACFKM, 243 ACL, 258 AB, 285 ACD

Stop

2. Route: A->C->F->K->M

Total cost: 61+31+112+32 = 236

3. The uniform cost search algorithm sorts the queue based on lowest accumulated cost. The A* algorithm includes the uniform cost search algorithm but the extended version. The extended version of the uniform cost algorithm sorts based on lowest (cost + heuristic) value, which the A* algorithm uses to sort the entire queue.

Comment:

Response history

Step	Time	Action	State	Marks
1	22/04/21, 15:30	Started	Not yet answered	
2	22/04/21, 16:15	Saved: 1.A 227AC, 258AB 228 ACF, 243 ACL, 258 AB, 285 ACD 236 ACFK, 243 ACL, 258 AB, 285 ACD 236ACFKM, 243 ACL, 258 AB, 285 ACD Stop 2. Route: A->C->F->K->M Total cost: 61+31+112+32 = 236 3. The uniform cost search algorithm sorts the queue based on lowest accumulated cost. The A* algorithm includes the uniform cost search algorithm but the extended version. The extended version of the uniform cost algorithm sorts based on lowest (cost + heuristic) value, which the A* algorithm uses to sort the entire queue.	Answer saved	
3	22/04/21, 16:30	Attempt finished	Complete	
4	**30/04/21, 16:04**	**Manually graded 9 with comment:**	**Complete**	**9.00**

Question 4

Complete

Mark 6.00 out of 6.00

1. Suppose it is desired to find the maximum of the function of two variables:

$$f(x,y) = -(x - x^3 - y^3)\, e^{-y^2}$$

Where parameters *x* and *y* vary between -8 and 8.

- Using genetic algorithm to solve the above problem give :

1. **one possible chromosome (3 points)**

2. **fitness function** for your chromosomes, explain how you chose your fitness function. (3 point)

Parameters for x and y: between -8 and 8 not inclusive => 15 values

First we need to encode: represent the x and y values each containing 8 bits

one possible chromosome: 11111001|00001000

if we have another possible chromosome we can crossover randomly and get new offsprings

Fitness function: substitute the x and y values in the function itself. Since we want the maximum, the higher the value of the output, the better the fitness. Then we can use the roulette wheel where we calculate the ratio of the individual chromosome fitness to the population total fitness. This ratio helped determine the chromosome chance of being selected. The average fitness improves.

Iterate either until the desired number of generations or until value is found

Comment:

Response history

Step	Time	Action	State	Marks
1	22/04/21, 15:30	Started	Not yet answered	
2	22/04/21, 16:28	Saved: Parameters for x and y: between -8 and 8 not inclusive => 15 values First we need to encode: represent the x and y values each containing 8 bits one possible chromosome: 11111001|00001000 if we have another possible chromosome we can crossover and get new offsprings Fitness function: substitute the x and y values in the function itself. Since we want the maximum, the higher the value of the output, the better the fitness. Then we can use the roulette wheel where we calculate the ratio of the individual chromosome fitness to the population total fitness. This ratio helped determine the chromosome chance of being selected. The average fitness improves. Iterate either until the desired number of generations or until value is found	Answer saved	
3	22/04/21, 16:30	Saved: Parameters for x and y: between -8 and 8 not inclusive => 15 values First we need to encode: represent the x and y values each containing 8 bits one possible chromosome: 11111001|00001000 if we have another possible chromosome we can crossover randomly and get new offsprings Fitness function: substitute the x and y values in the function itself. Since we want the maximum, the higher the value of the output, the better the fitness. Then we can use the roulette wheel where we calculate the ratio of the individual chromosome fitness to the population total fitness. This ratio helped determine the chromosome chance of being selected. The average fitness improves. Iterate either until the desired number of generations or until value is found	Answer saved	
4	22/04/21, 16:30	Attempt finished	Complete	
5	**30/04/21, 16:41**	**Manually graded 6 with comment:**	**Complete**	**6.00**