**NAME:**MOHAMED MAHMOD ABO-ELROUS

**ID:**2051710102

**Question 1:**

**MINI**

0

|  |  |  |
| --- | --- | --- |
| **X**  **MAX** | **O** | **-** |
| **O** | **O** | **X** |
| **X** | **-** | **-** |

-1

0

-1

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **X** |
| **O** | **O** | **X** |
| **X** | **-** | **-** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **-** |
| **O** | **O** | **X** |
| **X** | **-** | **X** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **-** |
| **O** | **O** | **X** |
| **X** | **X** | **-** |

**MINI**

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **X** |
| **O** | **O** | **X** |
| **X** | **O** | **-** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **X** |
| **O** | **O** | **X**  **MAX** |
| **X** | **-** | **O** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **O** |
| **O** | **O** | **X** |
| **X** | **X** | **-** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **-** |
| **O** | **O** | **X** |
| **X** | **X** | **O** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **O** |
| **O** | **O** | **X** |
| **X** | **-** | **X** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **-** |
| **O** | **O** | **X** |
| **X** | **O** | **O** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **O** |
| **O** | **O** | **X** |
| **X** | **X** | **X** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **X**  **MINI** |
| **O** | **O** | **X** |
| **X** | **X** | **O** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **O** |
| **O** | **O** | **X** |
| **X** | **X** | **X** |

|  |  |  |
| --- | --- | --- |
| **X** | **O** | **X** |
| **O** | **O** | **X** |
| **X** | **X** | **O** |

0

-1

-1

1

1

0

1

0

1

0

1.D \ Alpha-Beta Pruning: This is an extension of the minimax algorithm that reduces the number of nodes evaluated in the search tree by pruning branches that are guaranteed to be suboptimal. Implementing alpha-beta pruning can significantly reduce the search space and make minimax feasible for deeper and wider search trees.

**Question 2**

***1 ITERATION***

|  |  |
| --- | --- |
| **State** | **6** |
| **G(n)** | **0** |
| **H(n)** | **198** |
| **G(n)+h(n)** | **198** |
| **visit** | **0** |

**2 ITERATION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **State** | **6** | **1** | **3** | **4** |
| **G(n)** | **0** | **397** | **194** | **348** |
| **H(n)** | **198** | **541** | **380** | **381** |
| **G(n)+h(n)** | **198** | **938** | **574** | **729** |
| **visit** | **1** | **0** | **0** | **0** |

**3 ITERATION**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **State** | **6** | **1** | **3** | **4** | **6** | **5** | **4** | **1** |
| **G(n)** | **0** | **397** | **194** | **348** | **388** | **559** | **679** | **605** |
| **H(n)** | **198** | **541** | **380** | **381** | **198** | **161** | **381** | **541** |
| **G(n)+h(n)** | **198** | **938** | **574** | **729** | **586** | **720** | **1060** | **1146** |
| **visit** | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **0** |

**4 ITERATION**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **State** | **6** | **1** | **3** | **4** | **6** | **5** | **4** | **1** | **2** | **3** |
| **G(n)** | **0** | **397** | **194** | **348** | **388** | **559** | **679** | **605** | **720** | **924** |
| **H(n)** | **198** | **541** | **380** | **381** | **198** | **161** | **381** | **541** | **0** | **380** |
| **G(n)+h(n)** | **198** | **938** | **574** | **729** | **586** | **720** | **1060** | **1146** | **720** | **1304** |
| **visit** | **1** | **0** | **1** | **0** | **0** | **1** | **0** | **0** | **0** | **0** |

**C6-C3-C5-C2**

**Question 3:**

***depth-first search:***

|  |  |
| --- | --- |
| **1** | **{1}** |
| **2,3,4** | **{1,2},{1,3},{1,4}** |
| **5,6,3,4** | **{1,2,5},{1,2,6},{1,3},{1,4}** |
| **13,6,3,4** | **{1,2,5,13},{1,2,6},{1,3},{1,4}** |
| **6,3,4** | **{1,2,6},{1,3},{1,4}** |
| **14,15,16,3,4** | **{1,2,6,14},{1,2,6,15},{1,2,6,16},{1,3},{1,4}** |

-The states visited until reaching the target state are 6 states.

-Path {1,2,6,14}.

***breadth-first search*:**

|  |  |
| --- | --- |
| **1** | **{1}** |
| **2,3,4** | **{1,2},{1,3},{1,4}** |
| **3,4,5,6** | **{1,3},{1,4},{1,2,5},{1,2,6}** |
| **4,5,6,7,8** | **{1,4},{1,2,5},{1,2,6}{1,3,7},{1,3,8}** |
| **5,6,7,8,9,10,11,12** | **{1,2,5},{1,2,6}{1,3,7},{1,3,8}{1,4,9},{1,4,10},{1,4,11},{1,4,12}** |
| **6,7,8,9,10,11,12,13** | **{1,2,6}{1,3,7},{1,3,8}{1,4,9},{1,4,10},{1,4,11},{1,4,12},{1,2,5,13}** |
| **7,8,9,10,11,12,13,14,15,16** | **{1,3,7},{1,3,8}{1,4,9},{1,4,10},{1,4,11},{1,4,12},{1,2,5,13}{1,2,6,14},{1,2,6,15}{1,2,6,16}** |
| **8,9,10,11,12,13,14,15,16,17** | **{1,3,8}{1,4,9},{1,4,10},{1,4,11},{1,4,12},{1,2,5,13}{1,2,6,14},{1,2,6,15}{1,2,6,16},{1,3,7,17}** |
| **9,10,11,12,13,14,15,16,17** | **{1,4,9},{1,4,10},{1,4,11},{1,4,12},{1,2,5,13}{1,2,6,14},{1,2,6,15}{1,2,6,16},{1,3,7,17}** |
| **10,11,12,13,14,15,16,17,18** | **{1,4,10},{1,4,11},{1,4,12},{1,2,5,13}{1,2,6,14},{1,2,6,15}{1,2,6,16},{1,3,7,17},{1,4,9,18}** |
| **11,12,13,14,15,16,17,18,19,20,21** | **{1,4,11},{1,4,12},{1,2,5,13}{1,2,6,14},{1,2,6,15}{1,2,6,16},{1,3,7,17},{1,4,9,18},{1,4,10,19},{1,4,10,20},{1,4,10,21}** |
| **12,13,14,15,16,17,18,19,20,21,22,23,24** | **{1,4,12},{1,2,5,13}{1,2,6,14},{1,2,6,15}{1,2,6,16},{1,3,7,17},{1,4,9,18},{1,4,10,19},{1,4,10,20},{1,4,10,21},{1,4,11,22},{1,4,11,23},{1,4,11,24}** |
| **13,14,15,16,17,18,19,20,21,22,23,24,25,26,27** | **{1,2,5,13}{1,2,6,14},{1,2,6,15}{1,2,6,16},{1,3,7,17},{1,4,9,18},{1,4,10,19},{1,4,10,20},{1,4,10,21},{1,4,11,22},{1,4,11,23},{1,4,11,24},{1,4,12,25},{1,4,12,26},{1,4,12,27},** |
| **14,15,16,17,18,19,20,21,22,23,24,25,26,27** | **{1,2,6,14},{1,2,6,15}{1,2,6,16},{1,3,7,17},{1,4,9,18},{1,4,10,19},{1,4,10,20},{1,4,10,21},{1,4,11,22},{1,4,11,23},{1,4,11,24},{1,4,12,25},{1,4,12,26},{1,4,12,27},** |

**-The states visited until reaching the target state are 14 states.**

**-Path {1,2,6,14}.**

C-Breadth-First Search (BFS) and Depth-First Search (DFS) are essential algorithms for navigating tree or graph structures, each with distinct characteristics:

Traversal Approach:

BFS explores the graph level by level, while DFS delves as deep as possible along each branch before backtracking.

Data Structures:

BFS employs a queue, while DFS utilizes a stack or recursion.

Node Visiting Order:

BFS visits nodes in the order they were discovered, prioritizing closer nodes.

DFS may visit nodes in any order, emphasizing depth-first exploration.

Space Complexity:

BFS generally requires more memory due to its queue storage.

DFS typically demands less memory, particularly with recursive implementations.

Completeness and Efficiency:

BFS guarantees finding the shortest path in unweighted graphs.

DFS does not ensure the shortest path and may encounter infinite loops in cyclic graphs without cycle detection.

Applications:

BFS is useful for shortest path finding, network routing, and social network analysis.

DFS is applied in topological sorting, puzzle-solving, cycle detection, and searches where any path suffices.

In essence, BFS suits scenarios demanding shortest paths or specific node visiting orders, while DFS is ideal for deep exploration and cycle detection. The choice between them hinges on problem specifics and objectives.

**Question 4:**

the total cost=204



At node 3 the ant has 2 choices:

From C3 to C1 the heuristic value is 0.01 and its pheromone value is 35.35 hence P=0.473

From C3 to C2 the heuristic value is 0.01 and its pheromone value is 44.43 hence P=0.526

At node 2 the ant has 2 choices:

From C2 to C1 the heuristic value is 0.01 and its pheromone value is 2.06 hence P=0.134

From C2 to C0 the heuristic value is 0.06 and its pheromone value is 47.34 hence P=0.865

At node 0 the ant has 1 choices:

From C0 to C1 the heuristic value is 0.02 and its pheromone value is 46.55 hence P=1

1. pheromone=(1-ρ)pheromone  
   1-0.25=0.75

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **C0** | **C1** | **C2** | **C3** |
| C0 | 0.00 | 34.91 | 35.50 | 0.00 |
| C1 | 34.91 | 0.00 | 1.54 | 26.51 |
| C2 | 35.50 | 1.54 | 0.00 | 33.32 |
| C3 | 0.00 | 26.51 | 33.32 | 0.00 |

1. The route links pheromone will be enhanced by Δ=Q/L

Δ The updated Pheromone matrix:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **C0** | **C1** | **C2** | **C3** |
| C0 | 0.00 | 34.97 | 35.50 | 0.00 |
| C1 | 34.91 | 0.00 | 1.54 | 26.57 |
| C2 | 35.56 | 1.54 | 0.00 | 33.32 |
| C3 | 0.00 | 26.51 | 33.38 | 0.00 |

**Question 5**

V1=V1+α1(P1-X1)+α2(Pg-X1)

=+0.9

=+0.9

=

=

X1=X+V1

X1**=  
F(X1)=**5x11+2x22+6x33+9x45+1x57

= -861.88

**F(P1)=**5P11+2P22+6P33+9P45+1P57

= 6

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V2=V2+α1(P2-X2)+α2(Pg-X2)

=+0.9

=+0.9

=

=

X2=X+V1

X2**=  
F(X2)=**5x11+2x22+6x33+9x45+1x57

= -6205.88

**F(P2)=**5P11+2P22+6P33+9P45+1P57

= 236-

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V3=V3+α1(P3-X3)+α2(Pg-X3)

=+0.9

=+0.9

=

=

X3=X+V1

X3**=  
F(X3)=**5x11+2x22+6x33+9x45+1x57

= 75.4

**F(P3)=**5P11+2P22+6P33+9P45+1P57

= 279-

V4=V4+α1(P4-X4)+α2(Pg-X4)

=+0.9

=+0.9

=

=

X4=X+V1

X4**=  
F(X4)=**5x11+2x22+6x33+9x45+1x57

= 68.97-

**F(P4)=**5P11+2P22+6P33+9P45+1P57

= 28-

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V5=V5+α1(P5-X5)+α2(Pg-X5)

=+0.9

=+0.9

=

=

X5=X+V1

X5**=  
F(X5)=**5x11+2x22+6x33+9x45+1x57

= 365.88-

**F(P5)=**5P11+2P22+6P33+9P45+1P57

= 6-

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V6=V6+α1(P6-X6)+α2(Pg-X6)

=+0.9

=+0.9

=

=

X6=X+V1

X5**=  
F(X6)=**5x11+2x22+6x33+9x45+1x57

= 11.22-

**F(P6)=**5P11+2P22+6P33+9P45+1P57

= 139-

**Update Global best**

**f(p) < f(g)**

**g p**

p1 =

**f(p1) =**-861.88

p2=

**f(p2)**= -6205.88

P3= 

**f(p3)**= 279-

P4==

**f(p4)=-68.97**

**P5=**=

**f(p5)=**365.88-

P6=

**f(p6)=** 139-

f(gbest)=5x11+2x22+6x33+9x45+1x57

=-279

f(p1) < f(gbest) True

f(p2) < f(gbest) True

f(p3) < f(gbest) False

f(p4) < f(gbest) False

f(p5) < f(gbest) True

f(p6) < f(gbest) False

We pick minimum from p1 and p2,p5

Gbest p2

Minimize function then gbest =

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **X1** | **X2** | **X3** | **X4** | **X5** | **X6** |
| 2.5 | 2.1 | 3.9 | 0.2 | 1.6 | -0.4 |
| 0.3 | 0 | 1.8 | -3.2 | -1.1 | -0.8 |
| 0.9 | 1.6 | 2.2 | -1.8 | -0.9 | -1.9 |
| -2.5 | -3.7 | -1.1 | -1.1 | -2.1 | 1.3 |
| -0.6 | 0 | 0.7 | -0.5 | 0.9 | -0.8 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **V1** | **V2** | **V3** | **V4** | **V5** | **V6** |
| 2.5 | 1.1 | 4.9 | -1.8 | 0.6 | -1.4 |
| -0.7 | 2 | 3.8 | -4.2 | -3.1 | -0.8 |
| 0.9 | 0.6 | 1.2 | -3.8 | -0.9 | -2.9 |
| -4.5 | -2.7 | -1.1 | -2.1 | -4.1 | 0.3 |
| -1.6 | 2 | 0.7 | -1.5 | -0.1 | 0.2 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **P1** | **P2** | **P3** | **P4** | **P5** | **P6** |
| 2.5 | 2.1 | 2 | 0.2 | 1.6 | 1 |
| 0.3 | 0 | 0 | 3.2- | 1.1- | 0 |
| 0.9 | 1.6 | 0 | 1.8- | 0.9- | 1 |
| -2.5 | -3.7 | -2 | 1.1- | 2.1- | 0 |
| -0.6 | 0 | -1 | 0.5- | 0.9 | 1 |

global best is:=

**Next iteration**

V1=V1+α1(P1-X1)+α2(Pg-X1)

=+0.9

=+0.9

=

=

X1=X+V1

X1**=  
F(X1)=**5x11+2x22+6x33+9x45+1x57

=-194435.6

**F(P1)=**5P11+2P22+6P33+9P45+1P57

= -861.88

Update Pbest1

f(x) < f(p)

p x

P 1=



V2=V2+α1(P2-X2)+α2(Pg-X2)

=+0.9

=+0.9

=

=

X2=X+V2

X2**=  
F(X2)=**5x11+2x22+6x33+9x45+1x57

=-96420.87

**F(P2)=**5P11+2P22+6P33+9P45+1P57

= -6205.88

Update Pbest1

f(x) < f(p)

p x

P 1=



V3=V3+α1(P3-X3)+α2(Pg-X3)

=+0.9

=+0.9

=

=

X3=X+V3

X3**=  
F(X3)=**5x11+2x22+6x33+9x45+1x57

=-6974.97

**F(P3)=**5P11+2P22+6P33+9P45+1P57

= -279

Update Pbest1

f(x) < f(p)

p x

P 1=



V4=V4+α1(P4-X4)+α2(Pg-X4)

=+0.9

=+0.9

=

=

X4=X+V4

X4**=  
F(X4)=**5x11+2x22+6x33+9x45+1x57

=-9726.58

**F(P4)=**5P11+2P22+6P33+9P45+1P57

= -68.97

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V5=V5+α1(P5-X5)+α2(Pg-X5)

=+0.9

=+0.9

=

=

X5=X+V5

X5**=  
F(X5)=**5x11+2x22+6x33+9x45+1x57

=-119733.59

**F(P5)=**5P11+2P22+6P33+9P45+1P57

= -365.88

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V6=V6+α1(P6-X6)+α2(Pg-X5)

=+0.9

=+0.9

=

=

X6=X+V6

X6**=  
F(X6)=**5x11+2x22+6x33+9x45+1x57

=-16.23

**F(P6)=**5P11+2P22+6P33+9P45+1P57

= 12

Update Pbest1

f(x) < f(p)

p x

P 1=

=

Update Global best

f(p) < f(g)

g p

P1=

f(p1) ==-194435.64

P2= 

f(p2) ==-96420.87

P3=

f(p3) ==-6974.97

P4==

f(p4) ==-9726.58

P5==

f(p5) =-119733.59

P6==

f(p6) ==-16.23

f(gbest)=5x11+2x22+6x33+9x45+1x57

=6205.88

f(p1) < f(gbest) True

f(p2) < f(gbest) True

f(p3) < f(gbest) True

f(p4) < f(gbest) True

f(p5) < f(gbest) True

f(p6) < f(gbest) False

We pick minimum from p1 and p2,p3,p4,p5

Gbest p1

Minimize function then gbest =

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **V1** | **v2** | **v3** | **v4** | **v5** | **v6** |
| 2.38 | 1.1 | 2.65 | -1.23 | 0.75 | 0.61 |
| -0.79 | 2 | 1.64 | -3.24 | -2.77 | 0.16 |
| 1.11 | 0.6 | -0.96 | -2.78 | -0.15 | 0.76 |
| -4.86 | -2.7 | -2.69 | -2.88 | -4.58 | -2.37 |
| -1.42 | 2 | -1.04 | -1.35 | -0.37 | 2.06 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **X1** | **X2** | **X3** | **X4** | **X5** | **X6** |
| 4.88 | 3.2 | 5.55 | -1.03 | 2.35 | 0.21 |
| -0.49 | 2 | 3.44 | -6.44 | -3.87 | -0.64 |
| 2.01 | -2.2 | 1.24 | -4.58 | -1.05 | -1.14 |
| -7.36 | -6.4 | -3.79 | -3.98 | -6.68 | -1.07 |
| -2.02 | 2 | -0.34 | -1.85 | 0.53 | 1.26 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **P1** | **P2** | **P3** | **P4** | **P5** | **P6** |
| 4.88 | 3.2 | 5.55 | -1.03 | 2.35 | 0.21 |
| -0.49 | 2 | 3.44 | -6.44 | -3.87 | -0.64 |
| 2.01 | -2.2 | 1.24 | -4.58 | -1.05 | -1.14 |
| -7.36 | -6.4 | -3.79 | -3.98 | -6.68 | -1.07 |
| -2.02 | 2 | -0.34 | -1.85 | 0.53 | 1.26 |

Minimize function then gbest =

V1=V1+α1(P1-X1)+α2(Pg-X1)

=+0.9

=+0.9

=

=

X1=X+V1

X1**=  
F(X1)=**5x11+2x22+6x33+9x45+1x57

=-2455103.52

**F(P1)=**5P11+2P22+6P33+9P45+1P57

= -194435.6

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V2=V2+α1(P2-X2)+α2(Pg-X2)

=+0.9

=+0.9

=

=

X2=X+V2

X1**=f  
F(X2)=**5x11+2x22+6x33+9x45+1x57

= -652282.69

**F(P2)=**5P11+2P22+6P33+9P45+1P57

=-96420.87

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V3=V3+α1(P3-X3)+α2(Pg-X3)

=+0.9

=+0.9

=

=

X3=X+V3

X3**=  
F(X3)=**5x11+2x22+6x33+9x45+1x57

= 222266.60

**F(P3)=**5P11+2P22+6P33+9P45+1P57

=-6974.97

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V4=V4+α1(P4-X4)+α2(Pg-X4)

=+0.9

=+0.9

=

=

X4=X+V4

X4**=  
F(X4)=**5x11+2x22+6x33+9x45+1x57

= -276619.31

**F(P4)=**5P11+2P22+6P33+9P45+1P57

=-9726.58

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V5=V5+α1(P5-X5)+α2(Pg-X5)

=+0.9

=+0.9

=

=

X5=X+V5

X5**=  
F(X5)=**5x11+2x22+6x33+9x45+1x57

= -1489467.62

**F(P5)=**5P11+2P22+6P33+9P45+1P57

=-119733.59

Update Pbest1

f(x) < f(p)

p x

P 1=

=

V6=V6+α1(P6-X6)+α2(Pg-X5)

=+0.9

=+0.9

=

=

X6=X+V6

X6**=  
F(X6)=**5x11+2x22+6x33+9x45+1x57

=27110.47

**F(P6)=**5x11+2x22+6x33+9x45+1x57

=-16.23

Update Global best

f(p) < f(g)

g p

P1==

f(p1) =-2455103.52

P2==

f(p2) =-652282.69

P3=

f(p3) =-222266.60

P4=

f(p4) =-276619.31

P5=

f(p5) =-1489467.62

P6==

f(p6) ==-16.23

f(gbest)=5x11+2x22+6x33+9x45+1x57

=-194435.6

f(p1) < f(gbest) True

f(p2) < f(gbest) True

f(p3) < f(gbest) True

f(p4) < f(gbest) True

f(p5) < f(gbest) True

f(p6) < f(gbest) False

We pick minimum from p1 and p2,p3,p4,p5

Gbest p1

Minimize function then gbest =

B\ In scenarios with a large and complex solution space, particle swarm optimization (PSO) may face challenges in achieving convergence due to the risk of premature convergence and difficulties in adequately exploring the search landscape. However, with appropriate parameter tuning, initialization strategies, and diversity maintenance mechanisms, PSO can still be effective in navigating such spaces. Its ability to balance exploration and exploitation makes it suitable for tackling optimization tasks using deep local optimization. Moreover, integrating problem-specific knowledge and constraint-handling techniques can enhance the convergence behavior of PSO, enabling it to find high-quality solutions within reasonable computational resources.