**CSE422**

**Artificial Intelligence**

**Lab**

8- Queen

* 8 X 8 2d board.
* Queens can be allocated that no queen can be attacked by another queen horizontally, vertically or diagonally

Task

1. Write a fitness function which checks the fitness of a board by checking the number of non-attacking pair of queens.   Hint: Maximum number of non-attacking pairs of queens can be (8\*7)/2. Input: board, Output: a number telling the fitness of the board
2. Write a Crossover function. A crossover function will take two boards, an index number as input and return two new boards.
3. Write a Mutation function.
4. Create a population of randomly generated boards.
5. Randomly select two members of the population.
6. Randomly generate an index number.
7. Call crossover function using the above two as input.
8. Call fitness function for the new boards from the output.
9. Call mutation function if necessary.
10. Add new members to a new population set if appropriate.
11. Run 5 to 10 until the new population set is large enough.
12. Select a few members from the old population to add to the new population set.
13. Run 1- 12 until a board with highest fitness value is created.

**Department of Computer Science and Engineering**

|  |  |
| --- | --- |
| **Course Code:CSE422** |  |
| **Course Name: Artificial Intelligence** | **Prerequisite:**CSE111, CSE221 |

**Lab 03**

**Genetic Algorithm**

1. **Lab Overview:**

The students will solve N-Queen problem using python programming and visualizing the evolution performance.

1. **Learning Objective:**
2. Introducing the 4-Queen problem
3. Solution of 4-Queen problem in Backtracking approach
4. Demerits of Backtracking approach
5. Introducing 8-Queen problem
6. Discussion on Genetic Algorithm
7. Solution of 8-Queen problem using GA
8. **Lesson Fit:**

There is pre-requisite to this lab: CSE111, CSE221. You should have intensive Programming Knowledge and capability to understand algorithms.

1. **Acceptance and Evaluation**

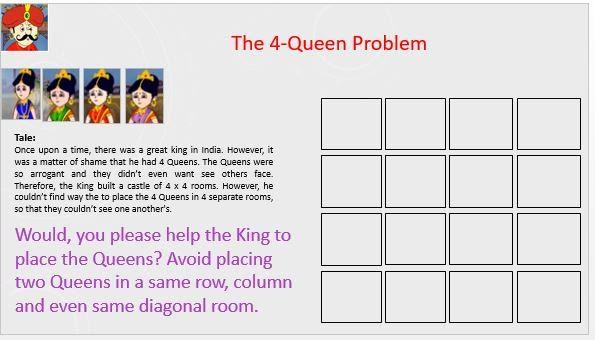
Students will show the output using different datasets and python code. They will be marked according to their lab performance. The main evaluation criteria will be based on project report and demonstration.

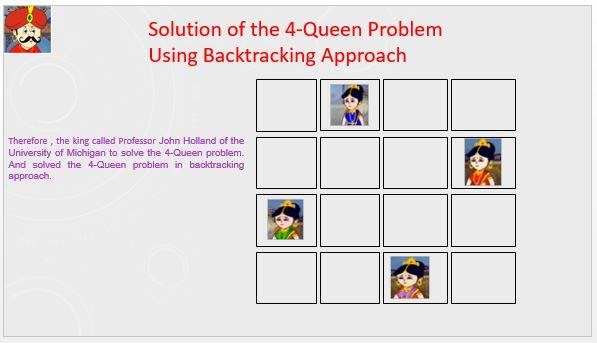
1. **Learning Outcome:**

After this lab, the students will be able to:

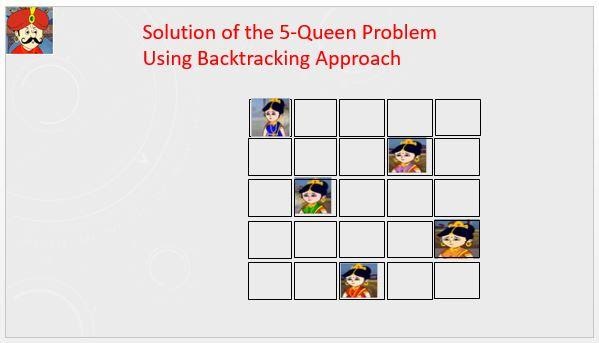
1. Demerits to solve N-Queen problem using Backtracking approach.
2. Solve the N-Queen problem using Genetic Algorithm
3. **Activity Detail**

* **Hour: 1.0 - 2.0**

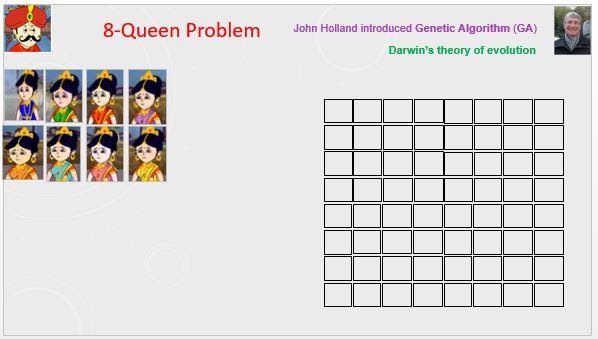


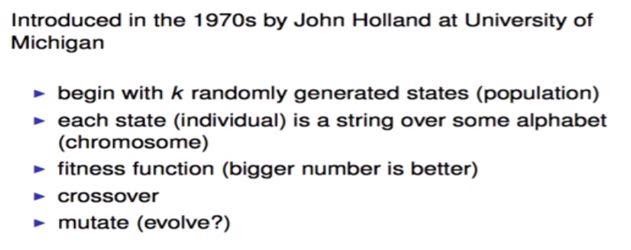


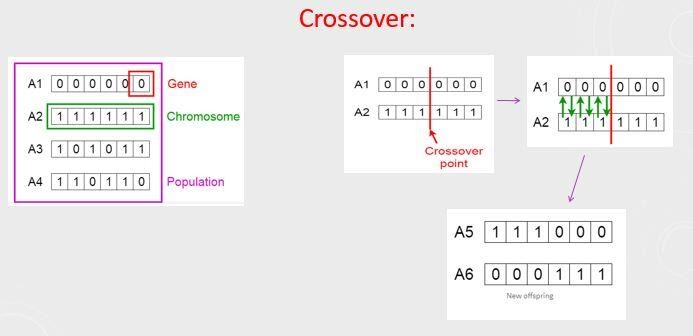


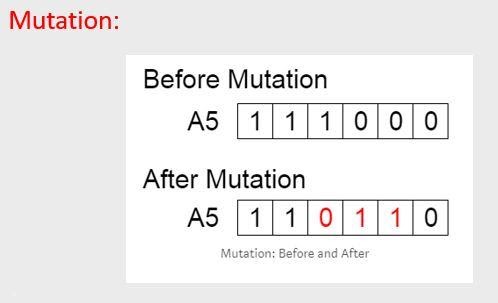


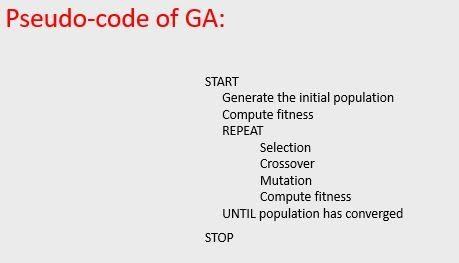


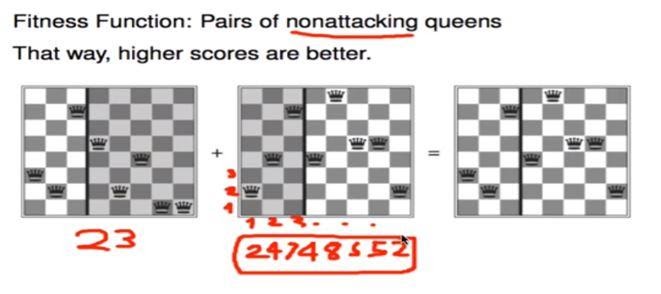


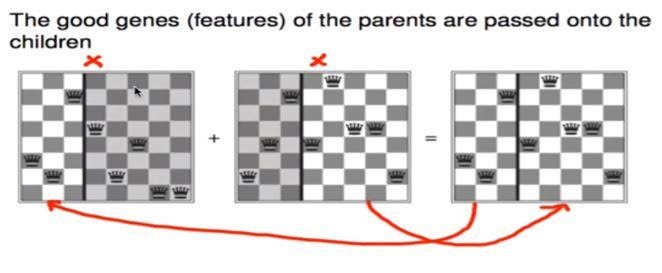


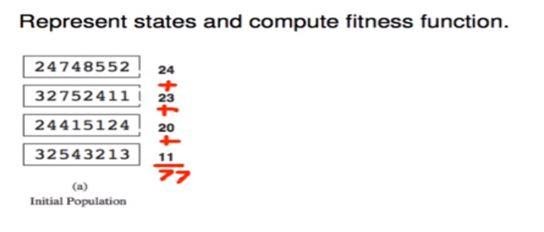


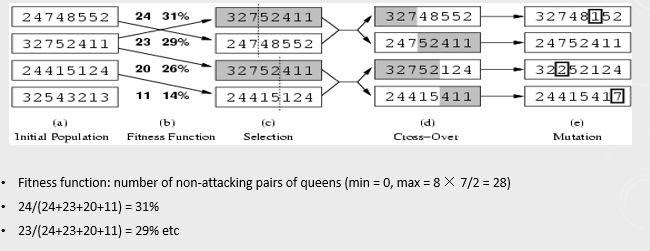


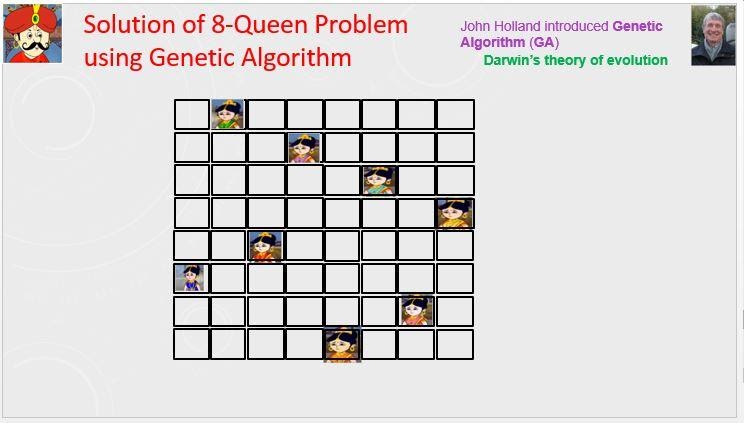


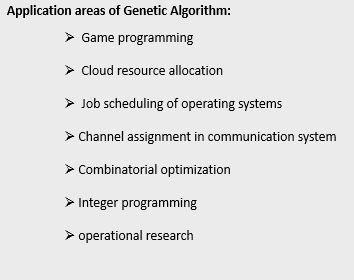












* **Hour: 2.0-3.0**

(It is Not a Group Task, Try Individually)

**Marks: 10**                                                                   **Time: 50 minutes**

**Task 1**: Implement N-Queen problem using Genetic Algorithm in python programming.

**Task 2**: Visualize the evolution through plotting the changes of fitness values, and the variances of fitness values for convergence.

Hints:  Take help from Prateek Joshi’s Book chapter 8, you can follow Covariance Matrix Adaptation Evolution Strategy (CMA-ES).

Evaluation Process (VIVA and Written answers): You have to explain your program and show your work to the Lab Instructor. Instructor may ask you some questions to evaluate your knowledge and expertise level.

---End---

Page  of