

OPEN ENDED LAB
CS-323 ARTIFICIAL INTELLIGENCE
TE CS BATCH 2022
FALL SEMESTER 2024

PROBLEM DEFINITION:

A **genetic algorithm (GA)** is a metaheuristic inspired by the process of natural selection that belongs to the larger class of evolutionary algorithms (EA). Genetic algorithms are commonly used to generate high-quality solutions to optimization and search problems via biologically inspired operators such as selection, crossover, and mutation. Some examples of GA applications include optimizing decision trees for better performance, solving sudoku puzzles, hyperparameter optimization, and causal inference.

Simulate the genetic algorithm using a simple example to understand how it works.

Produce the Python code of the genetic algorithm.

Discover a suitable computing problem and apply the coded genetic algorithm to solve it.

The task will be graded for CLO-3: **Practice** with algorithms for widely used computing operations, C3, PLO-3 using the rubric at the end of this file.

INSTRUCTIONS:

1. Students must explain the complete working of the genetic algorithm along with the example in the report. Also, they must describe the problem they have selected to be solved using genetic algorithm.
2. Students must come up with novel simulation and application to score good marks.
3. Students can make groups of **at most three** students for this assignment.
4. Students are required to deliver the Python project and a well formatted project report (of not more than 6 pages).
5. Submission must be made before 24th Nov 2024.
6. The report will be submitted in Google Classroom.
7. The student making the submission will mention his groupmates (names and roll numbers) in private comments. Other students will mention the name and roll number of the submitting group member in private comments.

DEPARTMENT OF COMPUTER & INFORMATION SYSTEMS ENGINEERING**Course Code: CS-323****Course Title: Artificial Intelligence****Open Ended Lab****TE Batch 2022, Fall Semester 2024****Grading Rubric****Group Members:**

| Student No. | Name | Roll No. |
|-------------|------|----------|
| S1 | | |
| S2 | | |
| S3 | | |

| CRITERIA AND SCALES | | | | Marks Obtained | | |
|--|--|--|---|----------------|----|----|
| | | | | S1 | S2 | S3 |
| Criterion 1: Has the student appropriately simulated the working of the genetic algorithm? | | | | | | |
| 0 | 1 | 2 | - | | | |
| The explanation is too basic. | The algorithm is explained well with an example. | The explanation is much more comprehensive. | | | | |
| Criterion 2: How well is the student's understanding of the genetic algorithm? | | | | | | |
| 0 | 1 | 2 | 3 | | | |
| The student has no understanding. | The student has a basic understanding. | The student has a good understanding. | The student has an excellent understanding. | | | |
| Criterion 3: How good is the programming implementation? | | | | | | |
| 0 | 1 | 2 | 3 | | | |
| The project could not be implemented. | The project has been implemented partially. | The project has been implemented completely but can be improved. | The project has been implemented completely and impressively. | | | |
| Criterion 4: How good is the selected application? | | | | | | |
| 0 | 1 | 2 | - | | | |
| The chosen application is too simple. | The application is fit to be chosen. | The application is different and impressive. | | | | |
| Criterion 5: How well-written is the report? | | | | | | |
| 0 | 1 | 2 | - | | | |
| The submitted report is unfit to be graded. | The report is partially acceptable. | The report is complete and concise. | | | | |
| Total Marks: | | | | | | |