**CENG 482 Evolutionary Computation Course Project**

**Objective**

This course project is designed to give you **hands-on experience** in designing, implementing, and experimenting with **evolutionary algorithms (EAs).** You are supposed to **select topics aligned with your interests**, enabling you to explore real-world applications or theoretical problems.

**Project Scope**

- **Topic Selection**: You are required to **propose** your project topic, ensuring it is **relevant to the field of evolutionary computation**. The topic must be discussed with and approved by the lecturer before proceeding.
- **Group Composition**
    - Projects can be undertaken **individually** or as a **group of 2–3 members**.
    - **Collaboration** should be well-organized, and each member's contributions must be clearly **specified in the final report** for group projects.

**3. Project Requirements**

- **Algorithm Design**
    - Every project should include **the design of an evolutionary algorithm**.
    - In the report (in IEEE conference paper format), the algorithm design section should explain all parts of the EA, such as **representation, initialization, selection, crossover, mutation, and termination criteria**, in detail.
- **Programming Language**
    - You are **free to choose any programming language** that you are comfortable with.

**4. Experimental Work**

In the experimental work section, start by completing the hyperparameter optimization, followed by the algorithm evaluation for a detailed analysis.

- **Algorithm Evaluation**:
    - The designed algorithm must **be tested through experimental work**.
    - The algorithm should be run **10 times for each case**, and the following statistics must be reported for the **final hyper-optimized version**:
        - Mean
        - Standard deviation
        - Best
        - Worst

- You should present your results using a **table** containing all final raw stats. In addition, you should present your results using **plots** (box-plot stats, line plot for convergence process, ...).

- **Hyperparameter Optimization**:
  - The project must include hyperparameter optimization as an experimental study.
  - At least **three parameters** should be optimized (e.g. crossover operator, mutation operator, population size, ...), with **three values per parameter**.
  - This results in **27 combinations** (3x3x3), and each combination should be evaluated using the **10-run approach**. (resulting in a total of 27 x 10 = 270 runs)
  - For hyperparameter optimization, detailed statistics **are not required**; only the **mean performance** of each combination should be reported for comparison for simplicity and ease of comparison.
  - Discuss the results and select best configuration.
  - Although not mandatory, you may also consider extending the hyperparameter optimization process to include more than three parameters, if necessary.

**5. Deliverables**

- **Project Report (paper)**
  - **Abstract**: Summarize the objective, approach, and key findings of the project in a concise manner.
  - **Introduction**: Provide background information, motivation, and an overview of the problem tackled in the project.
  - **Method (Algorithm Design)**: Describe the designed evolutionary algorithm in detail, including its components and implementation specifics.
  - **Experimental Work**: Present the experimental setup, results, and statistical analysis of the algorithm's performance, including hyperparameter optimization.
  - **Conclusion and Discussion**: Summarize the key findings, discuss their implications, and suggest potential future work or improvements.
- **Source Code**:
  - The implemented code must be submitted along with the project report.
  - The code should be well-documented and organized for readability.

**Timeline**

- **Topic Proposal and Approval Deadline**: [5 DEC 2024]
- **Demo of the Project + Report (paper format)**: [2 JAN 2025]

**Assessment Criteria**

The evaluation will be based on:

- Completeness and clarity of the algorithm design

- Rigor and depth of the experimental work
- Quality of the report
- Code functionality and organization
- Presentation of the project and demo

**Additional Notes**

- You are encouraged to consult with the lecturer regularly for guidance.
- Plagiarism will result in a penalty mark of zero and may also entail disciplinary penalties.