Assignment. Evolutionary Computation

Release: **4th September 2024**
Supervision: **19th September 2024**
Due: **3rd October 2024**

Dr. Alexandros Tzanetos

1 Introduction

The assignment work is carried out in groups of 3–5 students. The assignment comprises **2.5 credits**. The work includes coding in a programming language and writing a report. The assignment ends with a short viva (oral examination).

The assignment falls within the area of evolutionary optimization. The students can use any imperative or object-oriented programming language for the assignment work. Prolog or functional programming languages (e.g., LISP or Haskell) may not be used for these tasks.

2 Work Process

The assignment work is carried out in groups of 3–5 students, and you perform group division yourselves in Canvas. The assignment comprises **2.5 credits** and has a grading scale of 5, 4, 3, or Fail.

There are three examination occasions: the first is in October 2024, the second is in December 2024, and the last one is in August 2025. Students who have not obtained a grade for the assignment on any of these occasions will have to complete the assignment work the next time the course is given.

The assignment work includes:

- A hackathon session on Thursday, 19th September 2024, 10:15–12:00.
- A tutoring session given group-wise on Thursday, 19th September 2024, between 13:15–17:00.

- A short viva (oral examination) on Thursday, 3rd October 2024, 10:15–12:00 and 13:15–15:00.

All groups should attend the Hackathon session, which will take place in room B4066.

The tutoring is scheduled for room E2405. The viva sessions are scheduled in multiple rooms. You will find your room in the timeslot that I will book for you when the viva session is approaching. Each group may book **one slot per occasion**, and booking takes place via the Canvas calendar.

3 Tasks

This assignment is intentionally very **open-ended**, and the marking will depend on the quality of the submitted solution and report.

Optimization Problem. You shall implement and evaluate an evolutionary solution for an optimization problem of your own choice. Two possible candidate problems are n-queens and TSP (Traveling Salesperson Problem). If you want to work on another problem, you have to discuss that with Alexandros, by Monday, 19th September 2024.

Problem Instances. An important part of the task is to decide on the exact experimentation, but one thing that should be varied is *the size of the problem*: for n-queens, the number of queens and the corresponding size of the board; for TSP the number of cities. You must choose a problem where this is possible.

Parameter Tuning. In addition to the problem size, you may try different parameter values or vary some other aspects of your system.

Evaluation. For the actual evaluation, you should look at the quality of the solution, the running time, and maybe some other metrics. Here, you should test your system "to its limits", i.e., not just try a bunch of small problems.

Programming Language. The choice of programming language is free, except for Prolog or functional languages like Haskell, LISP, etc. - such programming languages may not be used.

There are, of course, many existing solutions to similar tasks online. While you may get some inspiration from those, you must write your own code from scratch, i.e., you may **not** copy-paste any code or exchange code between groups.

Note: The assignment aims to familiarize students with Evolutionary Computation. Therefore, other algorithms, such as Backtracking, are **not allowed**.

4 Report with Results Summary

You need to write a report where you describe:

- the problem,
- the system design (i.e., representation, fitness function, operations, parameter values, etc.),
- an experimental setup,
- results from the experimentation and
- some analysis with conclusions.

The report should be well-organized into different sections describing the items listed above. Use appropriate graphs to illustrate your results. Think carefully about the best design of the graphs.

A LaTeX template will be provided. It is advisable to follow the template's structure and familiarize yourself with LaTeX since you will find it useful for future courses.

5 Deliverables

The deliverables for this assignment are:

- A **pdf** document with your written report where you describe the problem, the system design (i.e., representation, fitness function, operations, parameter values, etc.), an experimental setup, results from the experimentation, and some analysis with conclusions.
- The numerical results from the experimentation saved in a text file.
- A **zip** file containing the source code for implementing a solution of an optimization problem and the executable files.

The source code must be thoroughly tested, and the executable should be able to run. Include any information about the setup needed to run your implementation. Moreover, the source code has to include a reasonable number of comments, i.e., the code is supposed to be well-readable.

Note: Do not forget to add **names of all the group members in a comment** at the beginning of the file with the main program.

6 Grading Criteria

The maximum score for the assignment is 30 points, and the grading is:

3	12–17.5	points
4	18–23.5	points
5	24–30	points

The points division is as follows:

Implementation	8	points
Experimentation	7	points
Report	10	points
Viva	5	points

Note: The basic condition to get points for the viva is to attend it. Any group member who does not show up in the viva they do not get points for that.