**Genetic Algorithms for TSP and VRP**

**Genetic Algorithms for solving the travelling salesman problem and the vehicle routing problem (TSP, VRP)**

This practical assignment requires to develop, using Python, an implementation of genetic algorithms for solving the [**Travelling Salesman Problem**](https://en.wikipedia.org/wiki/Travelling_salesman_problem)**-- TSP** and the [**Vehicle Routing Problem**](https://en.wikipedia.org/wiki/Vehicle_routing_problem)**-- VRP** (at least should include TSP)  
  
**Travelling Salesman Problem.** *Find the optimum itinerary for a salesman that needs to visit a set of cities, visiting each city exactly once, except the city where the trip started, that must be the last city to visit.*  
**Vehicle Routing Problem.** *Find routes for shipping supplies to a set of customers having different demands. The routes should be adjusted to the available fleet of trucks in order to get minimum costs*.

<https://www.cs.us.es/cursos/ia1-2011/trabajos/propuesta-gen-tsp.html>

**First part: genetic operators**

* A full standard genetic algorithm should be implemented in Python, including several (**at least one**) permutation-specific operators. For example:
  + *Partially Mapped Crossover (PMX)* ([slides 41 and 42](http://www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt)).
  + *Edge Crossover* ([slide](http://www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt)s 45, 46 and 47).
  + *Order Crossover* ([slide](http://www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt)s 39 and 40). ✔
  + *Insert mutation* ([slide](http://www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt) 34). ✔
  + *Swap mutation* ([slide](http://www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt) 35). ✔
  + *Inverse mutation* ([slide](http://www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt) 36). ✔
  + any other options found in alternative bibliography

<https://www.researchgate.net/figure/226665831_fig2_Figure-3-Cycle-crossover-CX>

**Second part: Variants over the standard GA**

* Modify the standard version of genetic algorithms developed in the previous step, by **choosing only one of the following:**
  1. *Genetic Algorithm with Varying Population Size*
  2. *Cellular Genetic Algorithm*

***Genetic Algorithm with Varying Population Size***The idea is to introduce the concept of "ageing" into the population of chromosomes. Each individual will get a "life-expectancy" value, which directly depends on the fitness.Parents are selected randomly, without paying attention to their fitness, but at each step all chromosomes gain +1 to their age, and those reaching their life-expectancy are removed from the population. It is very important to design a good function calculating life-expectancy, so that better individuals survive during more generations, and therefore get more chances to be selected for crossover.

***Cellular Genetic Algorithm***The idea is to introduce the concept of "neighbourhood" into the population of chromosomes (for instance, placing them into a grid-like arrangement), in such a way that each individual can only perform crossover with its direct neighbours.

**Third part: Experimentation**

* Run over the same instances both the standard GA (from first part) as well as the modified version (from second part). Compare the quality of their results and their performance. Due to the inherent randomness of GA, the experiments performed over each instance should be run several times.

**Final part: Documentation**

* Prepare a pdf report explaining the details of the implementations developed:
  + representation for genes and individuals, crossover and mutation operations, etc.
  + modifications performed over the standard algorithm,
  + instances considered,
  + number of executions for each instance, showing average statistics and best result found,
  + bibliography.

**Evaluation Criteria**

The following aspects will be taken into account for the mark of the assignment:

* General style and clarity of the code
* Efficiency of the implementation, and quality of the results obtained
* General impression during the interview
* General style and clarity of the report
* Any original additional contribution, going beyond the requirements displayed here
* **Important:** Using any non-original material without clearly indicating its source will result in a mark of zero in the course for all students involved, and may even yield other disciplinary measures.

**Bibliography**

* Slides from [unit 4](https://www.cs.us.es/docencia/aulavirtual/pluginfile.php/5152/mod_page/content/4/unit-04-2014-15.pdf).
* Chapters 2 and 3 of the book [*Introduction to Evolutionary Computing*](http://www.cs.vu.nl/~gusz/ecbook/ecbook.html) by A.E. Eiben and J.E. Smith. [Chapter 2](http://www.cs.vu.nl/~gusz/ecbook/Eiben-Smith-Intro2EC-Ch2.pdf) is available at the book's web page. Although full version of Chapter 3 might be useful for the assignment, it may be sufficient to check the [slides](http://www.cs.vu.nl/~gusz/ecbook/slides/Genetic_Algorithms.ppt) corresponding to this chapter that can be found at an online [course](http://www.cs.vu.nl/~gusz/ecbook/ecbook-course.html) using this textbook.
* It is also allowed to search the web for further references and/or related material. It is mandatory to include in the bibliography all references used.
* *Artificial Intelligence: A Modern Approach*. S. Russell and P. Norvig.
* [*GAVaPS - a Genetic Algorithm with Varying Population Size*](http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=350039&tag=1). J. Arabas, Z. Michalewicz, and J. Mulawla. Proc. 1st IEEE Conf. on Evolutionary Computation, pp. 73 - 78.  
  The direct link might require to have an IP within the campus to grant the download, but you can access from home via the library catalog ([Fama](http://fama.us.es/record=b2056567~S4*spi)).
* [Solving the Vehicle Routing Problem by Using Cellular Genetic Algorithms](http://www.springerlink.com/content/ynnj6qttmpxcn71h/?p=d48bae2115374966917456b46bbc0884&pi=1). E. Alba and B. Dorronsoro. LNCS 3004, pp. 11-20.
* Other books available at the University library. For instance:
  + [*Algoritmos Evolutivos: Un enfoque práctico*](http://fama.us.es/record=b2044773~S5*spi). L. Araujo, C. Cervigón.
  + [*Genetic algorithms and genetic programming : modern concepts and practical applications*](http://fama.us.es/record=b2058197~S5*spi). M. Affenzeller...[et al.]
  + [*Genetic algorithms + data structures = Evolution programs*](http://fama.us.es/record=b1283036~S5*spi). Z. Michalewicz.