****

**COMP5511 Artificial Intelligence Concepts**

**- Assignment 1 -**

**Important Notes**

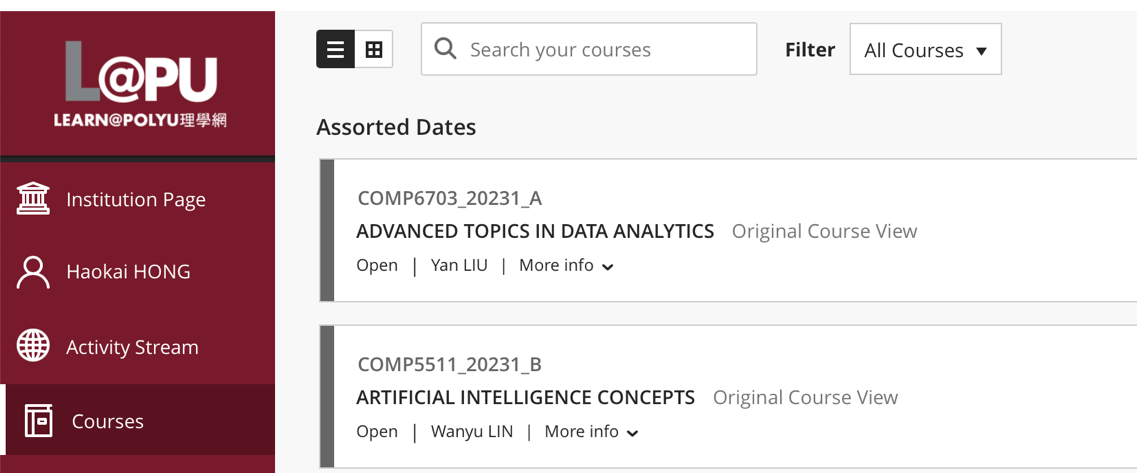
1. *Write your report using the given word template (maximum 15 pages). On top of the first page, provide your name and matriculation number.*
2. *Students can modify the Matlab file provided or geatpy (*[*https://github.com/geatpy-dev/geatpy*](https://github.com/geatpy-dev/geatpy)*) or use other programming languages to solve the problem given.*
3. *The solution and report should be the results of each individual work.*
4. *You must write the* ***readme*** *file to explain how the code works. (e.g., the code is implemented in Matlab language. The code is run on the Matlab R2020 version. The code requires the GA toolbox/package…)*
5. *The* ***report*** *together with the* ***codes*** *and* ***readme*** *should be submitted in a zip file (MatricNumber.zip) to LEARN@PolyU (*[*https://learn.polyu.edu.hk/ultra/course*](https://learn.polyu.edu.hk/ultra/course)*) under “COMP5511-> Assessments->Assignment 1” before the due date of* ***11:59 PM on 30 October 2024.*** *No late submission will be accepted.*

**Report Structure**

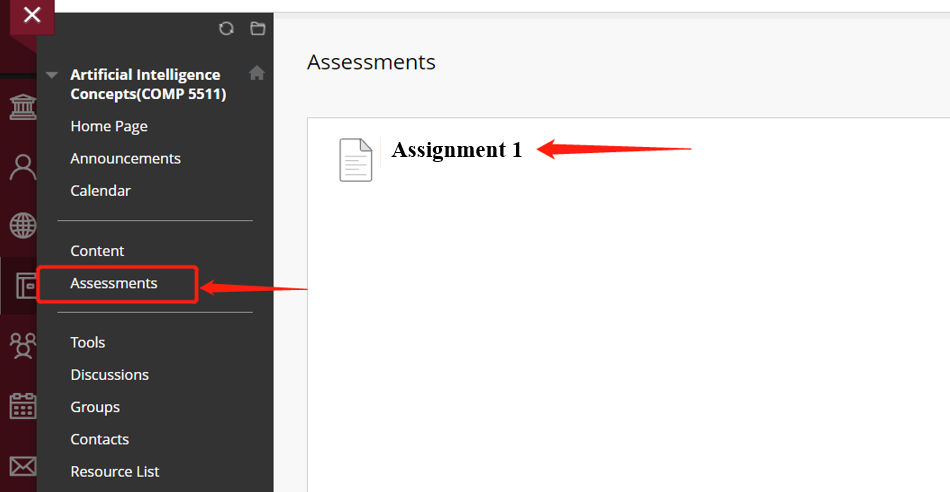
1. **Introduction**: Give a brief introduction about TSP, GA, etc.
2. **Methodology**: Include five subsections regarding the five tasks. In each subsection, you should give a detailed description of the designed algorithm, including the overall framework, crossover and mutation operators, selection operator, and other components.
3. **Experimental results**: Include five subsections regarding the five tasks. In each subsection, you should provide the experimental results and carry out ***sensitive studies*** for all parameters, including population size, mutation probability, and crossover probability, and ***discuss*** the effects of changing these parameters. You need to show the results in ***various formats***, such as tables, figures, etc. You are requested to run the algorithm ***multiple times***.
4. **Conclusion**: You should summarize what you have learned and your findings.

**Instruction: How to submit it online on LEARN@PolyU**

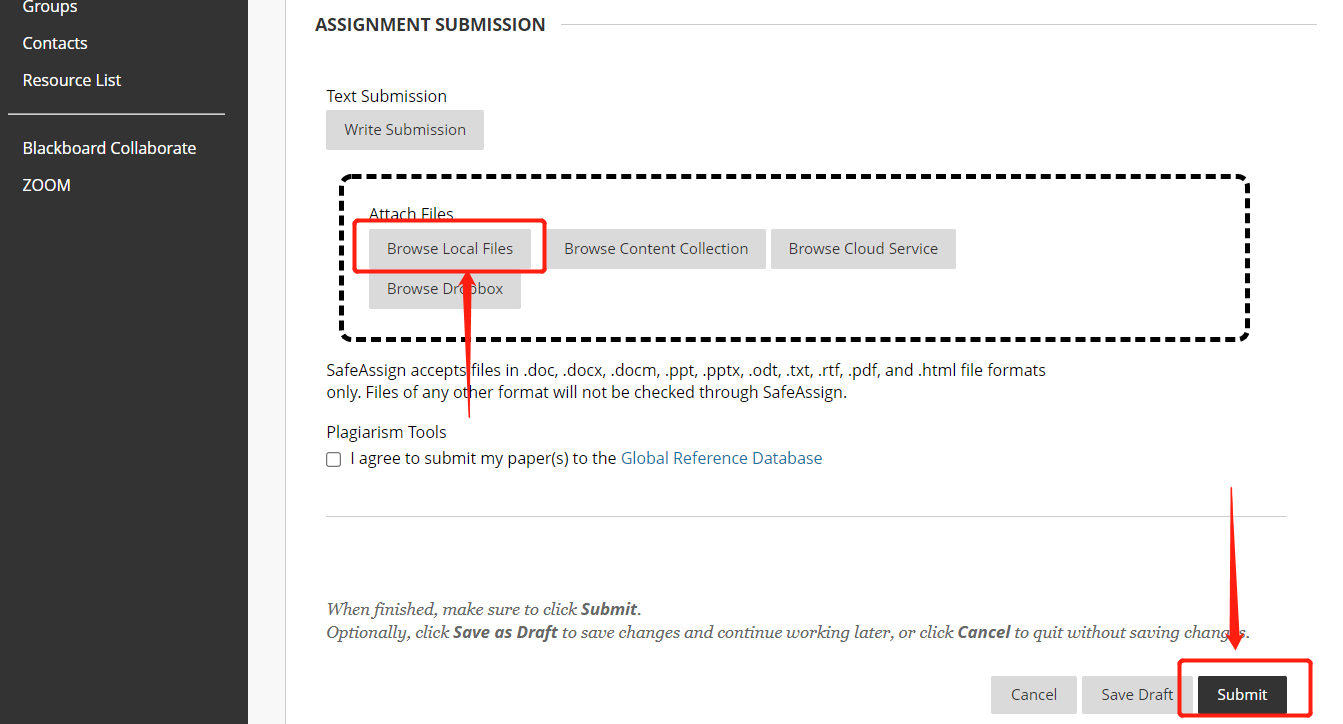
1. *Find and click COMP5511 on your course menu (*[*https://learn.polyu.edu.hk/ultra/course*](https://learn.polyu.edu.hk/ultra/course)*).*

**

1. *Find and click Assessments and Assignment 1*

**

1. *Attach your zip file and submit it*

**

# Assignment Description

***Abstract***: Traveling Salesman Problem (TSP) is a classical combinatorial problem that is deceptively simple. This problem is about a salesman who wants to visit n customers cyclically. In one tour, the salesman must visit each customer just once and should finish up where he started. Since each customer is situated in a different locations, the distance between every customer will be different. The objective is to find the shortest round-trip route that visits each customer once and then returns to the starting customer. The dataset with 100 customers is provided in the zip file (‘TSP.csv’).

In this assignment, students are required to finish the following five tasks:

## (1) Classical TSP

A genetic algorithm is used to find the shortest round-trip route of these 100 customers. The locations of customers are given in TSP.csv. Students should visualize the round-trip route and provide the total distance.

## (2) Dynamic optimization problem

In the real-world TSP, the location and number of customers may vary with time, such a TSP problem can be considered a dynamic optimization problem. Considering such a scenario:

For each environment , the first customers are allowed to be visited. Besides, the location of a customer varies with the environment ,

where and are the new X coordinate and Y coordinate at environment , respectively. and are the X coordinate and Y coordinate provided in the TSP.csv. Assuming that the environmental variable is changed every 100 generations, students should try to design a genetic algorithm to track the shortest round-trip route for each environment by reusing the solutions from the last environment to accelerate the search in the new environment and then compare the results from the genetic algorithm without reusing the solutions from the last environment.

## (3) Large-scale optimization problem

By adding 100 to the X coordinate for each customer in the TSP.csv, additional 100 customers can be formed. Regarding the newly formed 100 customers and the original 100 customers as a whole, the new problem can be regarded as a large-scale problem. For this large-scale problem, the customers can be divided into several small-scale regions by using clustering techniques, e.g., K-means. The salesman must finish visiting all the customers within the region before visiting any other customers in other regions. In this task, students are required to combine the clustering technique with a genetic algorithm to handle the large-scale optimization problem.

## (4) Multi-objective optimization problem

The salesman may consider more than one objective. For example, the salesman not only wants to minimize the travel distance of the round-trip route but also maximize the sales profit. Assume that the sales profit of each customer can be by the ‘PROFIT’ minus traveled distance, the two objective functions, (i.e., total travel distance and total sales profit ) may be conflicting, that is, a solution cannot satisfy the maximal sales profit and minimal travel distance at the same time. The multi-objective optimization problem can be formulated as An alternative approach is to change the multi-objective optimization problem to a single-objective optimization problem by weighting the two objective functions,

where () is the weight on . Students can specify the value to get the optimal solution.

In addition to the weighting objective functions-based method, students should develop a Pareto dominance selection-based evolutionary algorithm to handle the multi-objective optimization problem and discuss the advantages and disadvantages of the weighting objective functions-based method and Pareto dominance selection-based method.

## (5) Time window constraint problem

The salesman is required to visit a certain customer within a certain time window, i.e., the salesman should visit the customer between “READY TIME” and “DUE TIME”, and the time window for each customer is given in TSP.csv. The travel time between customers is computed by the Euclidean distance between customers. Considering the time window as an additional objective, students are required to develop a Pareto dominance selection-based evolutionary algorithm to solve the problem by optimizing the following three objectives: minimize total travel distance, maximize total sales profit, and minimize the total violation value of the time window, where the total violation value of the time window is the summation of the violation value of the time window for each customer. For example, “READY TIME” and “DUE TIME” of the “CUST NO 2” are 273 and 289, respectively. If the salesman visits the “CUST NO 2” at time 272, the violation value of the time window is 273-272=1. If the salesman visits the “CUST NO 2” at time 291, the violation value of the time window is 291-289=2.

## Discussion and analysis:

Students must give the details of the designed algorithms and perform sensitive studies for the above tasks with the various parameters, for example, the crossover and mutation rates, the population size, and the number of generations, and discuss the effects of changing these parameters. Students need to show their results in various formats, such as tables, figures, etc.

**References**

[1] [*http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/*](http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/)

[2] [*http://myweb.uiowa.edu/bthoa/TSPTWBenchmarkDataSets.htm*](http://myweb.uiowa.edu/bthoa/TSPTWBenchmarkDataSets.htm)

[3] [*https://github.com/geatpy-dev/geatpy*](https://github.com/geatpy-dev/geatpy)

[4] Deb, Kalyanmoy, et al. "A fast and elitist multiobjective genetic algorithm: NSGA-II." *IEEE Transactions on Evolutionary Computation,* 2002.

[5] Yang, Jin-Qiu, et al. "Solving large-scale TSP using adaptive clustering method." *IEEE International Symposium on Computational Intelligence and Design*, 2009.