IE 504- Heuristic Methods (Fall 2019)
Homework #2
Due date: December 15, Sunday @ 23:55

- Submit your assignment (the codes, input files, references, etc.) as a single .zip file through LMS.
- Use your last name(s) to name all of the files (e.g., code_gorgen_yazgan). Please do not use Turkish characters in the names and contents of the files.
- Do not submit .pdf files; submit your report in .doc format.
- Please remember that the format and style that you use to present your work always matter. Make sure that your discussions are clear.
- You can work alone or in groups of two.

Question 1. (36 points) Consider a single dimensional knapsack problem, where you are given a set of items n and their associated profits (p_i) and weights (w_i), and the knapsack's capacity is W.

The data for six instances of this problem are provided (instance1.txt, instance 2.txt, .., instance5.txt).

a. Use the enumerative algorithm (kp_enum.java on LMS) that can find the optimal solutions (optional: you can code your own enumerative algorithm) for the first three (small instances).
Present a table that reports your results.

b. Develop two local search algorithms to solve the knapsack problem with the following features:
- Create an initial solution based on a greedy procedure (i.e., first pick the item with the largest w_i/p_i *ratio)*
- Apply the following improving moves:
  i. Swap: Remove one item from the knapsack, while inserting another item
  ii. Add: Add an item which is not in the knapsack
- Local search algorithm version 1: Apply the swap and add moves by evaluating all candidate moves and following a best improvement strategy.
- Local search algorithm version 2: Apply the swap and add moves by generating random moves and following a first improvement strategy.

Present a table that reports your results. Submit your code (code for one experiment setting is sufficient).

c. Evaluate the performance of the local search algorithms on the given instances with respect to each other as well as the optimal solutions (which could be obtained only for the small instances). Recall that stochastic solution methods require multiple runs (e.g., 3 runs per instance). Discuss your findings.

Question 2. (40 points) Develop a Simulated Annealing (SA) algorithm for this knapsack problem. You must adopt one of the versions you develop in Question 1 (i.e., Local search algorithm version 1 or 2) into a SA framework (e.g., apply Metropolis criterion to accept solutions).

a. Set the algorithm parameters as follows (make the necessary changes in the code you developed in Question 1) and apply the algorithm to solve the given problem instances.
- Initial temperature=5000;
- A geometric cooling schedule with a cooling rate of 0.95;
- Epoch length: 3 iterations;
- Algorithm termination criteria: Algorithm should stop when the temperature achieves a very small value (e.g., 0.001)

Note that the SA algorithm is a stochastic algorithm; therefore, in order to evaluate its performance, you need to make multiple runs. Run the algorithm 3 times for each instance, and evaluate the performance of the algorithm (compared to the optimal solutions) based on its average, worst case and best case performance.

b. Make experiments with your SA algorithm with different parameters settings, which are provided in the table below. Solve each instance five times for each experimental setting.

Experiment parameters:

Experiment #	Initial temperature	Cooling schedule	Epoch length
1	10,000	Geometric (α=0.95)	3
2	10,000	Geometric (α=0.75)	3
3	5,000	Geometric (α=0.95)	3
4	5,000	Geometric (α=0.75)	3

In each experiment, the algorithm must stop when the temperature reaches a very small value such as 0.001.

Present a table that reports your results. Submit your code (code for one experiment setting is sufficient).

c. Based on the results presented in part b., discuss the performance of your algorithm, applied with different parameter settings, in solving both instances.

d. Discuss how the SA solutions compare to those of the local search algorithms and the optimal solutions that you reported in Question 1.

e. Consider problem instance #6 (i.e., instance6.txt), and apply SA by using the parameters in experiments #1 and #3 on this instance. Plot the current and best objective values achieved through the algorithm for both experiments. Discuss the effects of initial temperature on the results.

f. Consider problem instance #6 again (i.e., instance6.txt), and perform experiments #1 and #2 in this instance. Plot the current and best objective values achieved through the algorithm for both experiments. Discuss the effects of cooling schedule on the results.

Question 3. (24 points). Find two articles (per team member) related to the project problem (or a similar/related problem to your project problem) that you will work on this semester, which present i) a Simulated Annealing algorithm, and ii) a Tabu Search algorithm to solve the associated problems. Submit the articles, and answer the following questions:

- **a.** Explain the steps of the algorithm briefly. Specifically, describe i) how the initial solution was generated, ii) the neighborhood structure/moves, and iii) any other important strategies used (e.g., diversification, adaptive memory, etc.).
- **b.** How were the algorithm parameters set? Did authors perform any preliminary experiments to set the parameters?
- **c.** How were the instances generated for testing the proposed solution methods?
- **d.** How do the authors compare the performance of the heuristics? What are the important findings?