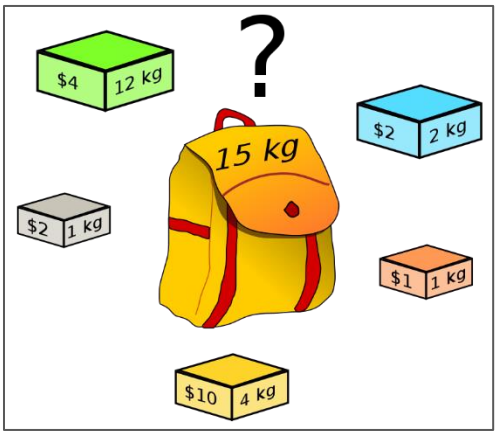
Cairo University
Faculty of Computers & Artificial Intelligence
Department of Computer Science
Soft Computing Course - 2022

Assignment 1 – Knapsack Problem

About the problem:

The knapsack problem is a very well-known optimization problem. Given a knapsack that can carry weights up to a certain amount, and a number of items; each item has a weight and a value, we want to select the items to carry in the knapsack in order to maximize the total value.



What you are required to do:

Write a genetic algorithm to solve the knapsack problem. (A solved example is provided in lab 1)

What the input looks like:

You'll be given an input file with the following format:

- *First line:* Number of test cases (must be at least 1)
  For each test case:
- Size of the knapsack
- Number of items
  For each item:
- Weight and value separated by space

Example:

```
2

10
3
4 4
6 6
5 3

15
5
12 4
1 2
4 10
1 1
2 2
```

Important remarks to help you solve the problem:

1. Use a **binary, one-dimensional** chromosome.

2. The population size and initialization method you use are up to you. You can actually try different population sizes to see how this will affect your results. The maximum number of generations is also up to you.

3. Think about how you will handle **infeasible solutions**. Infeasible solutions are solutions that violate the constraints of the problem; therefore, they are not allowed.

4. Use **rank selection and one-point crossover.** Choose the methods of mutation and replacement that you find appropriate.

5. **The output** should consist of the test case index, the number of selected items, the total value, and the weight and value of each selected item.

Assignment submission notes:

- The **maximum** number of students in a team is **3** and the **minimum** is **2**.
- The **deadline will be announced, and no late submission** is allowed.
- Please submit **one compressed folder**. The folder name should follow this structure: ID1_ID2_ID3_MAJOR_MINOR_GROUP
- **Cheating** students will take **negative grades** and no excuses will be accepted. If you have any problems during the submission, contact your TA but don't, under any circumstances, give your code to or take the code from your friends.

Grading Criteria: (5 marks)

Representation, structure and initialization	0.5
Fitness function	0.5
Handling infeasible solutions	0.5
Selection, crossover, mutation and replacement	3
Output	0.5

Good luck