Assignment 1: Genetic algorithms

Note: you may work in pairs if you wish. On the submission include ID number and name for both parties. Only one member need submit.

Part (a)

i) One-max problem.

Consider the simple case of evolving a string that contains all 1s in every location. Let the length of the strings be 20.
The initial population should be randomly created. Use standard mutation and one-point crossover. The fitness of a solution is the number of 1s in the string.

Plot the average fitness of the population versus the generations passed.

This exercise is to show the operation of a genetic algorithm.

ii) Evolving to a target string

Define a target string (a sequence of 1s and 0s) and adopt the same approach as before with a different a fitness function (number of matching values).
Plot the average fitness as before.

iii) Deceptive Landscape

Modify the fitness function in i) such that the fitness function is equal to the number of 1s in the string for all cases except when there are no 1s present. In this case the fitness should be 2*(length of the solutions).

The above are to show the basic operation; how the same approach can be used for different fitness functions and how GAs can struggle with deception.

Part (b) The Knapsack problem

The classical knapsack problem involves selecting a subset of items from a set such that all items fit in the knapsack and the value is maximised.

More formally, all items *i* have a weight *wi* and a value *vi*. We can represent this as a vector of weights and vector of values where location *i* in these vectors represent the weight and value for item *i.*

The task it so select a set of these items such that the sum of the weights is less than or equal to a threshold (the capacity of the knapsack) and the sum of the values is maximised.

i) Choose a representation for your candidate solutions
ii) Devise a fitness function
iii) Use simple mutation and one point crossover.

Test your GA on the following test problem:

Values: 78, 35, 89, 36, 94, 75, 74, 79, 80, 16

Weights: 18, 9, 23, 20, 59, 61, 70, 75, 76, 30

Problem 1: Knapsack size: 103

Problem 2: Knapsack size: 156

Submit: solution to both problems; submit plot of the evolution (best and average fitness versus time/generation)

Overall submission requirements.

Document containing:

- Names and IDs (if a pair submission)
- Part (a):
    - Short description of GA implemented.
    - Plots of the evolutionary progress for each of the three problems
    - Short description of results.
- Part (b):
    - Description of representation
    - Description of fitness function
    - Description of operators used
    - Plots of the evolutionary progress
    - Results for the two knapsack sizes above.
- Code: zip file or a link to github