Activity 2 – Evolutionary Algorithms (10%)
Activity is due on Friday, April 8, 2022, 11:59 pm.

Consider the same problem as in Activity 1: Given a set of n packages with profit p_j and weight w_j , and a set of m containers with weight capacity c_i, select m disjoint subsets of packages so that the total profit of the selected packages is maximum, while ensuring the containers' capacity is never exceeded.

Consider the same instances as in Activity 1:

Instance 1	**Instance 2**

random.seed(1)
n = 100 #number of packages
m= 5 #number of containers
c=50
 	
#Generate random locations
pj = random.choices(range(10, 100), k=n)
wj = random.choices(range(5, 20), k=n)
	random.seed(1)
	n = 10000 #number of packages
	m= 200 #number of containers
	c=50
	
	#Generate random locations
	pj = random.choices(range(10, 100), k=n)
	wj = random.choices(range(5, 20), k=n)

Exercise 1: Propose and apply an Evolutionary Programming Algorithm (i.e. evolutionary algorithm with mutation operator only) for the problem.

- **Initial Population:** uniformly randomly generated
- **Population Size:** 10 to 50 solutions
- **Selection Strategy:** roulette wheel selection
- **Reproduction Strategy:** only mutation
- **Replacement strategy:** generational replacement with elitism

Exercise 2: Propose and apply a Genetic Algorithm (i.e. evolutionary algorithm with crossover operator and mutation operator) for the problem.

- **Initial Population:** uniformly randomly generated
- **Population Size:** 10 to 50 solutions
- **Selection Strategy:** roulette wheel selection
- **Reproduction Strategy:** crossover + mutation
- **Replacement strategy:** generational replacement with elitism

Deliverables: (1) textfile (or screenshot) of the code used to run each exercise**;** (2) pdf report including: (2.1.) solution encoding used ; (2.2) explanation of the design inputs considered (see slide 35 – lecture 10); (2.3) explanation of the mutation and crossover operators used ; (2.4) results obtained for each exercise.

The results should include: (2.3.1) a table with information of the objects included in each container, the total weight in each container, the profit in each container (only for instance 1); (2.6.2) a plot with the cpu performance (for instance 1 and 2). See example below:

Table 1 - Table of results

Container	Packages	Profit	Weight
1	3,5,7	54	50
2	10,24,1	64	49
3	89,21	48	49
4	55,63,29	61	50
5	8,14,13,47	75	48
Total		**302**	

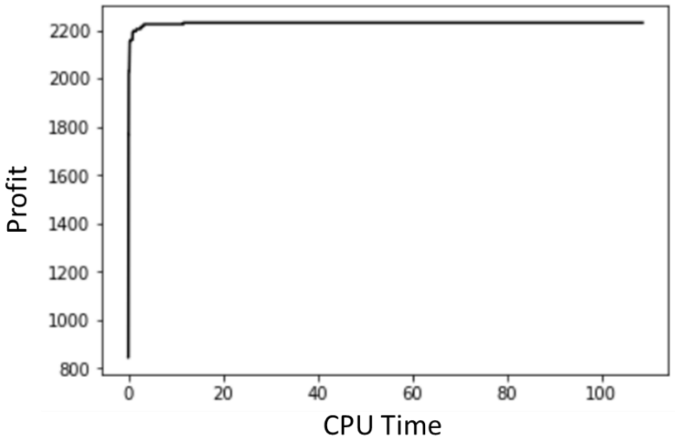

Fig 1 – CPU performance