**COMP304 Assignment 2**

**Name: Sabelo**

**Surname: Zulu**

**Student numbe : *217023502***

**Report**

a) **Chromosome in the initial population:**

A chromosome in the population of this problem consist of **genes** each gene has the following properties: Gene\_Name(a string in alphabets A-Z) Gene\_Weight( in the form of an integer) and a Gene\_Value( also in a form of an integer) and a length of a gene according to the Alphabet range (A-Z).

For example if a chromosome has 9 alphabets its length will also be 9, Alphabets A-I will be represented as [ A B C D E F G H I J ]. A-J are the names of a gene in a chromosome respectively.

In a binary representation(Binary Encoding) form we represent a randomly put a genes in an Chromosome and represent it as a 1 if it will be transported else a 0, e.g if A can be transported it’s represented as 1 in a chromosome. E.g in a chromosome [ A B C] binary representation , [ 1 0 0 ] means only gene { A } can be transported.

**Population Size:**  Initial population size is set to 10.

b) **Fitness Function :**  A chromosome is considered to be fit if and only if its total weight is less or equal to the capacity of the vehicle, and if it meets that requirement it is flagged 1 else 0. Total weight is the sum of all genes that are randomly putted in the chromosome (genes represented in binary value of 1). To be putted in a chromosome means to be in the delivery vehicle.

c) **Selection Method :**  A random selection was used, I used an array list that took every chromosome in the population, but the chromosomes that were considered fit when evaluated with the fitness function were added twice to increase the chances of being selected.

d) **Mutation operator :** For mutation I used **Bit Flip Mutation** . I created a method that takes a chromosome as a parameter then it randomly chooses an index in that chromosome then flip the gene in that index e.g in a chromosome [ 1 0 1 ] if we randomly choose index 0 that would result in a new chromosome [ 0 0 1].

**Mutation Rate :**  I used a floating number constant that I assigned to 0.1 as my mutation rate, before mutating I used a random floating number between 0 and 1 so then check if it is less or equal my mutation rate, if it is less or equal to my mutation rate I do mutation in the selected chromosome and return it, else I return the selected chromosome as it is.

e) **Crossover Operator :** I used a **One Point Crossover**, I created the method that takes two parent Chromosomes selected in the population, and then selected a random crossover point which is a random index of the chromosome length, After that I swapped the tails of each parent Chromosome to produce two new off-springs then returned them.

**Crossover Rate :** I used a floating number constant that I assigned to 0.85 as my crossover rate, before mutating I used a random floating number between 0 and 1 so then check if it is less or equal my crossover rate, if it is less I do crossover in the selected parent chromosomes and return the children, else I return the fittest selected parent chromosome as it is.

f) **Termination Criterion :**  we terminate the program when we found a chromosome that has a total weight that is less or equal to the capacity of the vehicle and a total value greater or equal to the vehicle’s quota.