GA Algorithm (Knapsack Problem)

Problem statement:

What we decide to solve to the typical knapsack problem. For example, there are many bags and each of them has different value and weight. Given a limited max weight, we put these bags into our knapsack. So what would be the biggest total value?
Considering this is an optimization problem, we can definitely use GA algorithm to solve it although the result might not be the best solution.

Implement details

To solve this problem, we assume each solution is a separate individual carry a set of genes.
Genotype: the number of digits of genes is equal to the total amount of bags, while each digit can only be value “0” or “1” representing not in knapsack and in knapsack respectively. In this way, we can know which bags are inside the knapsack. For example, the genotype is “0110”, which represents there are 2 bags in the knapsack— the second and third one.
Expression: if the value of a digit is “0”, there will not be any change to the total value or weight of the knapsack. When the value of a digit is “1”, the total value and weight of the knapsack would be increased at the value and weight of this responsive bag.
Phenotype: the total value and weight of bags in the knapsack
Fitness function: there will be divided into two different circumstance. On the one hand, if the total weight of bags is smaller than the allowed maximum weight. The fitness is simply the value of the sum of all the values of bags inside the knapsack. On the other hand, if the total weight of bags is bigger than the allowed maximum weight, the fitness would be 0 as a result because this situation is not allowed.

GA Algorithm strategy

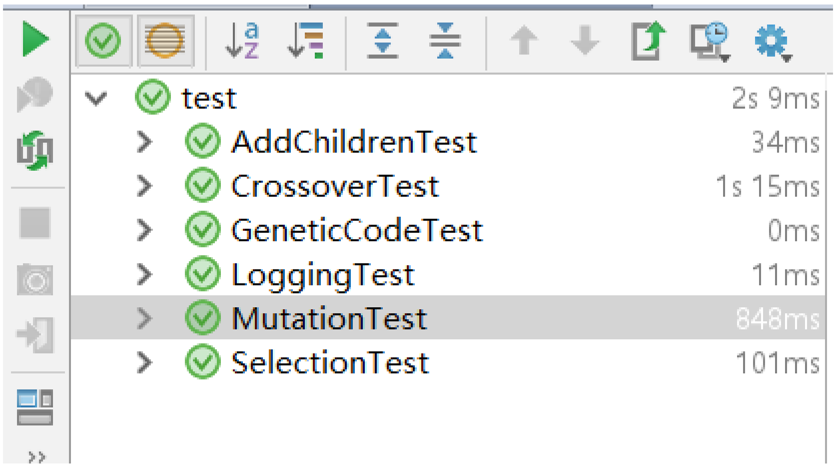
Evolution: every generation, keep the top 50% fittest individuals and let them to have off springs according to their fitness. Each pair of parents will have two children, then their children and themselves is composed of the next generation.
Population: a set of individuals. To ensure the diversity, I randomly initialize the population as 10000.
Individual: each individual holds a set of genes, which is generate randomly between “0” and “1”. And I have a method to calculate the fitness of this individual.
Selection: once I finish calculating the fitness of individuals among the whole population. I add each of them to a Priority Queue and sort them according to their fitness. Finally, only keep the top 50% fittest individuals.
Crossover: when the parents are mating, randomly choose a position in the genotype

to crossover. After this process, the genotype of their off springs would be the result of exchanging parents' genotypes.

Mutation: with a very small rate, there will be mutation, just randomly choose the digit of genes, and change it.

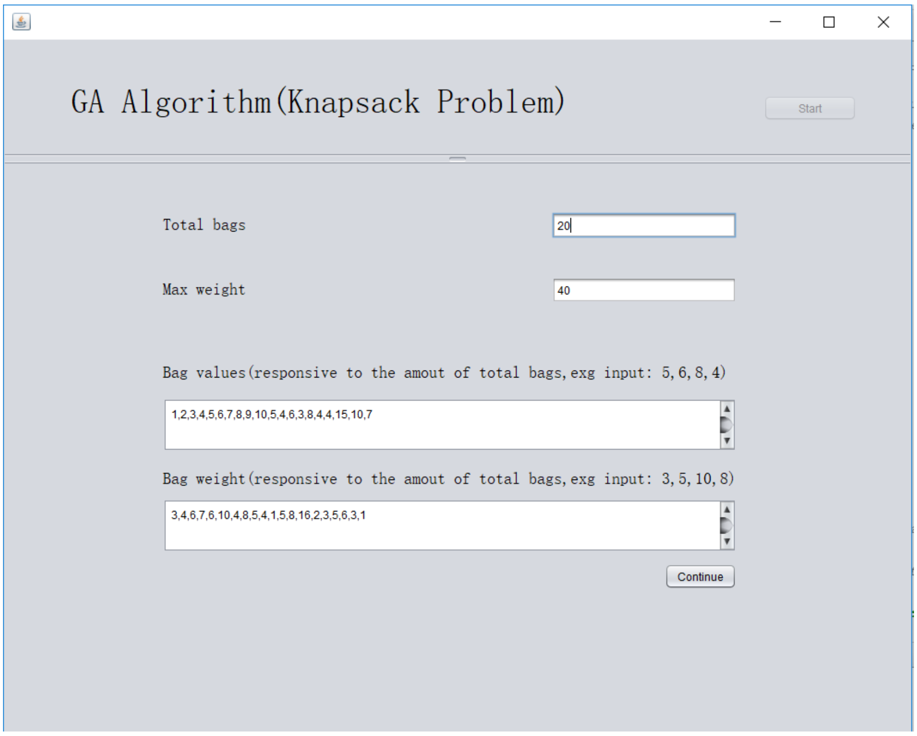
Add Children: Simply add the off springs to the remained individuals to be the next generation.

Unit tests to verify functions

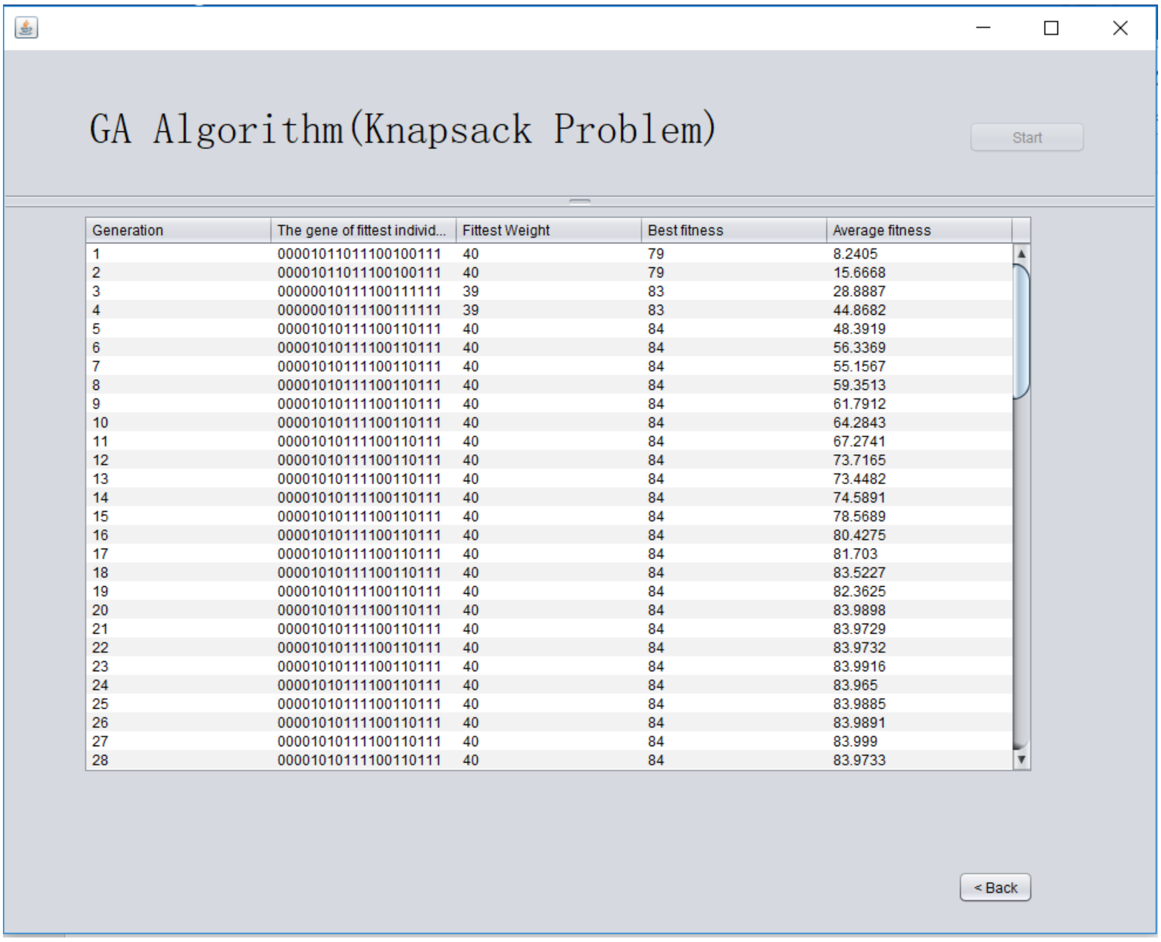


Results

Inputs:



Results



Generation	The gene of fittest individ...	Fittest Weight	Best fitness	Average fitness
1	00001011011100100111	40	79	8.2405
2	00001011011100100111	40	79	15.6668
3	00000010111100111111	39	83	28.8887
4	00000010111100111111	39	83	44.8682
5	00001010111100110111	40	84	48.3919
6	00001010111100110111	40	84	56.3369
7	00001010111100110111	40	84	55.1567
8	00001010111100110111	40	84	59.3513
9	00001010111100110111	40	84	61.7912
10	00001010111100110111	40	84	64.2843
11	00001010111100110111	40	84	67.2741
12	00001010111100110111	40	84	73.7165
13	00001010111100110111	40	84	73.4482
14	00001010111100110111	40	84	74.5891
15	00001010111100110111	40	84	78.5689
16	00001010111100110111	40	84	80.4275
17	00001010111100110111	40	84	81.703
18	00001010111100110111	40	84	83.5227
19	00001010111100110111	40	84	82.3625
20	00001010111100110111	40	84	83.9898
21	00001010111100110111	40	84	83.9729
22	00001010111100110111	40	84	83.9732
23	00001010111100110111	40	84	83.9916
24	00001010111100110111	40	84	83.965
25	00001010111100110111	40	84	83.9885
26	00001010111100110111	40	84	83.9891
27	00001010111100110111	40	84	83.999
28	00001010111100110111	40	84	83.9733

Conclusion

As can be seen from the results above, the average fitness tends to be convergence since generation 20, which means the phenotype of population is not a lot changed anymore. The population has nearly become the most appropriate group. Therefore, we can conclude the genes of the fittest individual in this group would be the solution.

However, GA algorithm can not ensure the accuracy, and the result might not be the same at each running. To get the accurate answer, we need to make more effort on this.