BBM203 PROGRAMMING LAB. - REPORT GENETİC ALGORITHM

Yusuf Emre Genç

21803663

1 Problem Definition

In this assignment, I am expected to implement a genetic algorithm by using linked list implementation. This report focuses on how I implemented the linked list operations and how constructed the structure of solution.

2 Methods and Solution

I defined only one struct for linked list implementation. Actually I needed 2 struct for linked list of chromosomes and genes separately, but thanks to utilization of void pointer of C programming language, I could implement a generic structure for any type of linked list. This made my life easier and my program more modular. Also I stored a rear pointer for the sake of time efficiency, by this way I avoid traversing to end of list to add new element.

3 Functions

3.1 readPopulation

This function initialize the population by reading data from population file. It uses a function named addRear that adds new element directly accessing to end of list efficiently. addRear function exploits from the generic property of the linked list so that there is only one function to add for any type of linked lists.

3.2 mergeSortPopulation

I used merge sort algorithm to sort chromosomes. This algorithm makes my program more time efficient than some other algorithms. It is a recursive algorithm working the idea of divide-conquer.

3.3 iterateGene

This is the main loop of the program. It supports the main idea of the program (Selection, Cross Over, Mutate). It selects the chromosomes and so that mixes the order of chromosomes.

3.4 xOver

This function swaps the parts of two chromosomes. It uses utilizations of linked lists when swapping parts of chromosomes. It basically updates the next data of the nodes instead of swapping individually all gene nodes of chromosomes. It provides so much efficiency in large problems.

3.5 mutateChromosome

This function accesses to appropriate gene of the chromosome and complements the value.

3.6 calculateFitnesses

This function takes the chromosomes and calculates the fitness by traversing through it.