Big-O analysis

We used map, vector and so on throughout our program. Vectors were used for storing the the items to be purchased in each stadium. The vector tracked the fan’s cart contents and was able to add items which the fans so much pleased. We use vector to store each prices of souvenirs, such as Baseball cap. The vector requires a maximum O-notation of O(n). When the fans purchase the souvenir, for example two Baseball caps, the system will push the price of Baseball cap twice. Similarly, if fans want to buy other souvenirs, their prices will be stored in the vector. The Big-O notation for purchasing souvenirs is O(n).

Our program needs to read data from a file. The read function should read all the information of MLB one by one. The read function's big-o notation is O(n). We read the team name and its corresponding stadium name. At the same time, we also need to read the information, including Seating capacity, Location, playing surface, league, date opened, distance to center field, Ballpark typology and Roof Type. We just read by order and the big-o notation is O(n).

After open the file and read the information, we store all of them in the map. The big-o notation is the same with the read function because the function also need to store the information one by one. Once the admin changes the information, the modified data will be saved in the file for permanent use. The big-o notation for open the file is O(n).

Our program should also meet the requirement to sorted by team name, stadium name, park typology and seating capacity. Each sort function using the for loop to search the information first. The big-o notation for sort function is O(n2).

For the graph part, we use the function dijkstraAlgorithm(), which is the method to perform the said algorithm from a desired starting index and output the shortest distance from the starting city that baseball fans want to visit. If the vertices reached is the total size of the graph, remove all the discovery edges. The big-o notation for dijkstraAlgorithm() function is O(n2).

We also create the function of MinimumSpanningTree() to show the MST which will be used to determine the minimum spanning tree given a starting city. It will go through the entire graph and determine the minimum distance that one could travel to reach all possible locations on the entire graph. The function will check the each city as a vertices and once the city visited, the vector will track the discovery edge and the total travel distance. The big-o notation for MinimumSpanningTree() function is O(n2).