PROPOSAL to compare the costs and time of multiple sorting algorithms

Contents

Introduction

The following is a proposal to test the speed and efficiency of multiple sorting algorithms and then determine which is most suited to find the closest desired establishment from any point on the map. We will use the most basic available technology to sort a size-varying array of distinct numbers which represent distances using multiple sorting algorithms which are Insertion Sort, Merge Sort and Quick Sort. We will compare the results and determine which sorting algorithm is most suited for this task. The algorithms will be written in C++ and the code will be compiled on the Windows platform with the aid of the g++ compiler. In addition to measuring the speed of the algorithms we will also compute the memory space required for the sorting.

Problem: What is the fastest and cheapest way to find the closest desired establishments from a specific location?

Background

We do not know the most suited algorithm to be used for finding the closest distances from a specific location to several establishments. People by nature are lazy and disinclined to any sort of compromise. That is why it is so important that we make the best program suited to find the closest restaurants or other establishments. According to Ass. Prof. Doshi, Jain and Shakwala; in the modern world, people are constantly on the move with their electronic devices that allow them to have access to nearly all knowledge known to man. By being aware of the user`s geographic location it is possible for computer programs to provide a variety of impressive services (2014). The selling factor of the program will be the sorting algorithm that is used to give customers results in the shortest amount of time as possible. Akl said that performance of algoritm is based on two steps' time: computing and memory using. Different element size can change algorithms' calculation time and this affects the performance of program (1985). The best sorting algorithms we could come up with are Insertion Sort, Merge Sort and Quick Sort.

Woznlak, Marszalek Gabriel and Nowicki stated that sorting algorithms that are used to sort large amounts of data, sometimes it can be difficult to find the right sorting especially for special entries. The most effective solutions for these problems are changing the input type or algorithm (2013). These three sorting algorithm will be tested to organize large amount of data. Our test results will give the most suitable sorting algorithm.

Proposal Summary

On Windows 10 we will use the software known as dev-C++ to compile a program. The program will read the latitude and longitude of 10, 100 or 1000 buildings and then find the closest ones from a three separately given location using the aforementioned algorithms. The software will be made by our highly esteemed team and beta tested by a group of random people (I think one is an ex-convict). There will be nine trials for each algorithm, making a total of 27 runs.

Outcome

The results of the trials will be processed and compared by our prestigious minds and published along with our professional opinions on which algorithm is suited for this particular task. The results of the experiments will be published on 23/11/2100 (not a typo, we checked with every science magazine and this was the soonest available date) in '*The Mars News'*.

Value

Finding the correct sorting algorithm for the task of finding the closest eating establishment form a user`s location is an utmost priority. Since this is a new field of study that hasn't been fully explored yet, the success of this experiment will beget civil happiness and stimulate the country`s economy, particularly the food industry.

Methods

With only %19 of the budget we believe that we can present satisfactory results with 50 days. There are three members in our group and three sorting algorithms to be tested, each of our group members will experiment on one of the algorithms. Insertion Sort, Merge Sort and Quicksort will be tested by Ibrahim, Baran and Kurshat respectively.

Schedule

	Ibrahim Turkmen	Baran Kaya	Kurshat Yasar
8/03/2017	Insertion Sort experiment 1	Merge Sort experiment 1	Quick Sort experiment 1
9/03/2017	-	-	Quick Sort experiment 2
10/03/2017	Insertion Sort experiment 2	-	Qucik Sort experiment 3
12/03/2017	Insertion Sort experiment 3	Merge Sort experiment 2 & 3	-

Conclusion

The launching of this project will undoubtedly bring good to the future and to our inflated egos. The financial gain is also not to be overlooked. With this discovery we will rise to the top of the food chain by gaining power over local establishments. Any questions, concerns, complaints or love-confessions should be directed to our project manager, Baran Kaya.

References

Pankti Doshi, P. J. (2014). Location Based Services and Integration of Google Maps in Android. *International Journal Of Engineering And Computer Science*, 5072.

Selim G. Akl, (1985). *Parallel Sorting Algorithms*. Orlando, Florida: Academic Press Inc. *https://books.google.com.tr/books?hl=tr&lr=&id=jhHjBQAAQBAJ*

Skulimowski, A. M. J. (2013). Knowledge, information and creativity support systems: recent trends, advances and solutions. In J. Kacprzyk (Ed.), *Preprocessing large data sets by the use of quick sort algorithm* (pp. 111-121). Warsaw: Polish Academy of Sciences.