**MAKERERE****UNIVERSITY**

**COLLEGE OF COMPUTING AND INFORMATION SCIENCE**

**SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY**

**BACHELOR OF SOFTWARE ENGINEERING**

**DATA STRUCTURES AND ALGORITHMS**

|  |  |
| --- | --- |
| **Name** | **Registration number** |
| Nakaali Phionah | 15/U/9286/PS |
| Mwesigye Robert | 15/U/771 |
| Nabengo Mariam | 15/U/8919/EVE |
| Nansubuga Joyce Euzebia | 15/U/10807/EVE |
| Nabikolo Shiba | 15/U/8925/PS |
| Odoch Robert | 14/U/1033 |
| Muyambi Julius | 15/U/8610/PS |
| Bwayo Jude | 15/U/4718/EVE |

**Introduction**

**T**he Fraud detection system is a web based application that serves the purpose of checking if all the students have paid tuition before they are being allowed to sit their final exams.

Each student is expected to at least have made 10 installments meaning that each student must have 10 unique receipts, and there are 40,000 private students at Makerere University.

A student is required to come with three receipts at the entrance of the examination room and then the registrar randomly selects a receipt among those receipts and picks it number and enters it into the Fraud detection system which should return the student number associated to that particular receipt number if it corresponds with that on the receipts provided by the student then the student is allowed into the examination room.

The nature of the receipt number used is a 5-length string which can constitute of both letters and numbers for example **42Uxg, Fr8A4, 4LGJJ** and so on,

Solutions to the Questions

1. We randomly generated the receipt numbers for all the student and stored them in a file which we read from whenever we need the receipts numbers and store them into an array of 40,000 by 10.
2. Sorting.

Under sorting we used these sorting algorithms below

1. Bubble sort
2. Selection sort
3. Insertion sort
4. Merge sort
5. Quick sort

When each of the above algorithms were used to sort the 40,000 student receipts,

Insertion sort was the fastest, followed by Quick sort, then Selection sort then Merge sort, and Bubble sort used the highest time, their running time was captured and is summarized in the table below;

|  |  |
| --- | --- |
| **Sorting Algorithm** | **Time to sort 40,000 student’s receipts (seconds)** |
| Insertion sort | 6. 475796101093292 |
| Quick sort | 7. 63071203218152 |
| Selection sort | 8. 7641389369965 |
| Merge sort | 24. 078860998154 |
| Bubble sort | 35. 928941011429 |

1. Sorting the 10 by 10 array of receipts gives Insertion sort the first position as the fastest followed by Quick sort, then Selection sort, followed by Merge sort and Bubble sort in the last position, their running time in seconds is summarized as the table below shows.

|  |  |
| --- | --- |
| **Sorting Algorithm** | **Time to sort 10 students’ receipts (seconds)** |
| Insertion sort | 1. 0014021396636963 |
| Quick sort | 1. 00160479545593262 |
| Selection sort | 1. 0018889904022217 |
| Merge sort | 1. 005033016204834 |
| Bubble sort | 1. 0089399814605713 |

1. If we pick a certain receipt number randomly from the records and search for it using the two search algorithms Linear search and binary search, the times taken for the algorithms to return the results is as follows;

|  |  |
| --- | --- |
| **Search** **Algorithm** | **Time taken (seconds) for 40,000 students** |
| Linear search | 0.0005340576171875 |
| Binary search | 0.00036907196044922 |

Therefore, the Registrar should use Binary Search as it returns the results in less time compared Linear Search.

1. The Binary search algorithm takes 6.4852330684662 seconds to handle 500 students.