**Data Structures & Algorithms – Final Report**

**A technical report covering the applications of hash tables & graphs.**

**A black and white logo

Description automatically generated**

**By: Mark Lambert (C00192497), Darius Herdes (C00296548)**

**Course: Software Development (CW\_KCSOF\_B)**

A black and grey logo

Description automatically generated**Submission Date: 10/12/2024**

**Table of Contents Page**

[**Introduction** 3](#_Toc183508453)

[**Hash Table Application** 4](#_Toc183508454)

[**What is a Hash Table?** 4](#_Toc183508455)

[**Application** 4](#_Toc183508456)

[**Table of Data** 4](#_Toc183508457)

[**Diagram of Hash Table Produced** 4](#_Toc183508458)

[**Graph Application** 5](#_Toc183508459)

# **Introduction**

This report was commissioned by Áine Byrne as a final project for the Data Structures & Algorithms module. The purpose of the report is to display an understanding of the implementation of a hash application, a graph application including a coded protype to show said implementation. The report should also show one’s ability to perform as part of a group dynamic.

# **Hash Table Application**

## **What is a Hash Table?**

A hash table is a data structure wherein a piece of information is stored in an encrypted manner after performing what’s known as a “hash function” on some input. The hash function produces an index position for one single unit of information to be stored in the hash table, which is implemented as an array.

## **Application**

The hash table application will take in a username (max characters: 6) and create a unique index for each username. The key is calculated by accumulating the total decimal ASCII value of the characters and performing a modulo of 20. In the case of a collision, a linear probe will be considered.

For this implementation, the final hash table will be of length 20 and the hash function will be % 20.

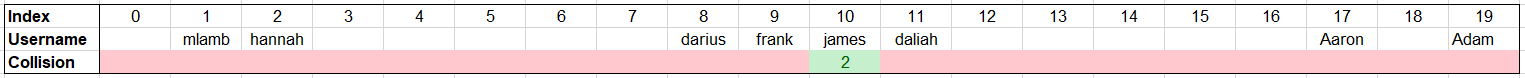
Example hash function equation

## **Table of Data**

The following table of data shows an example of 8 username inputs, some with uppercase characters.

|  |  |  |
| --- | --- | --- |
| **Username** | **ASCII** | **Index**  **(after hash function %20)** |
| mlamb | 521 | 1 |
| darius | 648 | 8 |
| AdAm | 339 | 19 |
| hannah | 662 | 2 |
| Aaron | 497 | 17 |
| frank | 529 | 9 |
| james | 528 | 8 |
| daliah | 611 | 11 |

# **Diagram of Hash Table Produced**

****Below is a diagram visualising what the hash table will look like with each username in their respective index as well as any collisions which occurred.

# **Graph Application**