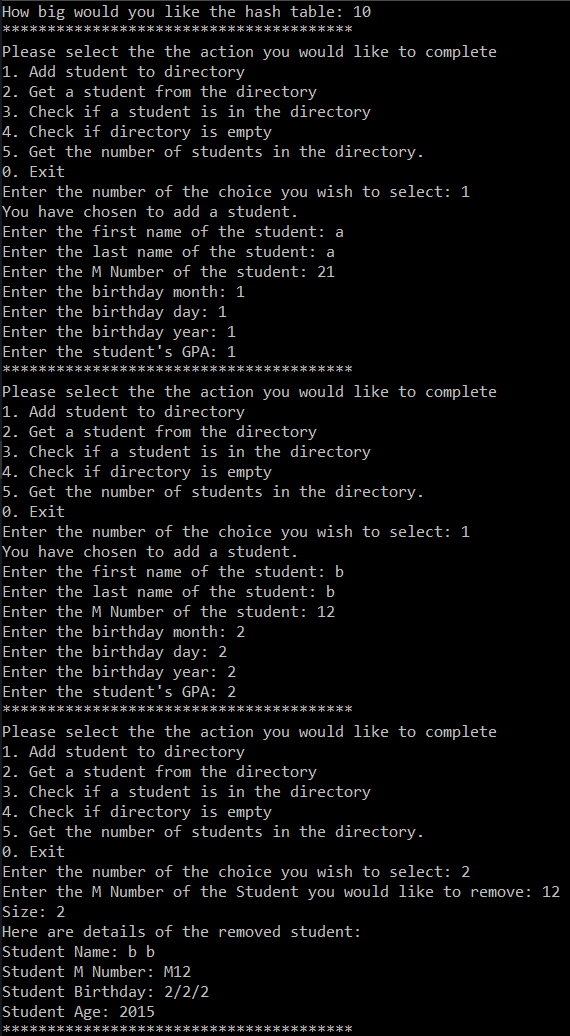
Hash tables are an important data structure used to organize a lot of data into one container. Using a key to map the different sets of data, a hash table can organize data based on hashed key given by the user. Another data structure used in this lab was the linked list data structure. A data structure that includes a piece of data and then a pointer to the next node in the list. Both of these data structures were combined in one lab to show the differences in performance of just linear probing through a hash table and using a linked list for chained linking. This shows how the use of these data structures can be used for the same thing, storing data, in different ways.

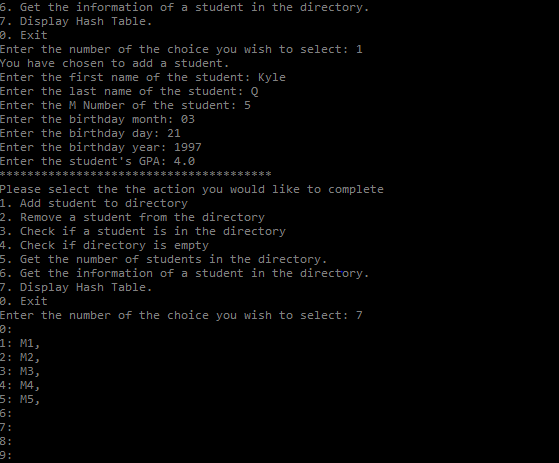
|  |  |  |  |
| --- | --- | --- | --- |
| 100 | 36 | 100 | 4 checks |
| 150 | 36 | 150 | 4 checks |
| 200 | 36 | 200 | 4 checks |
| 250 | 36 | 250 | 4 checks |
| Array Size ( Probing) | Performance | Array Size( Chained) | Performance |

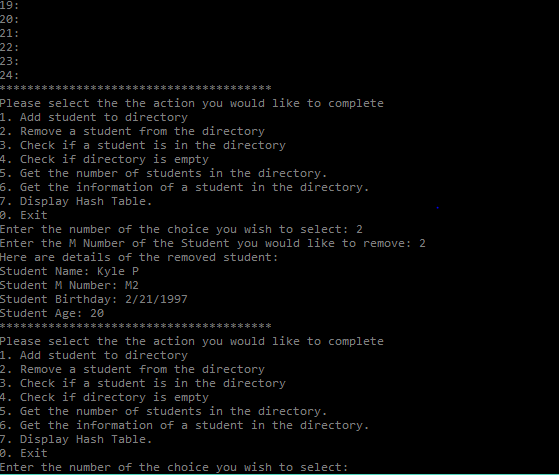
\*This is searching for mNumber 10000040 for 50 random student

Task 3 Screenshot



Task 4 Screenshots





With task 2, the only thing that needed to be changed was the overload of the toString() function. However, implementation of this class would be a little bit different. For task 3 information from each student that is made would be added as a pointer to the hash table and then the values can be looked for by traversing the table and looking at each index. In task 4 the implementation of the chained linking was used instead of just the hash table to look at and comparing each value compared to the value that is being looked for.

In task 5 for linear probing and searching for mNumber 10000040, the number of times the index is searched through and that number doesn’t change at all as the array size increased. Now searching for different mNumbers will change the number of indexes the linear probing has to go through. On the other hand, chained linking looks at the values and compares. This time there are only 4 checks when looking for a certain mNumber. This value also doesn’t change when increasing the size of the array to say 150 or 250. Something for further investigation would be to see if smaller amount of student and a larger array size, or visa-versa would greatly affect the performance of each test.

Compilation Instructions

This has been tested by creating a new project within Visual Studios with the following options:

Win32 Console Application

Create directory for solution OFF

Empty project ON

Precompiled header OFF

SDL OFF

Then:

Add the following files to the projext:

1. HashTable.h

For linear probing:

1. Lab09\_main.cpp

For chained linking:

1. Lab09Task4\_main.cpp

\*Testing (task 5) is added to both the cpp’s

Build and run

Contribution of Team Members

Evan Akers is responsible for task 1

Saylee Dharne is responsible for task 2 & 4

Kyle O’Connor is responsible for task 3 & lab report

Smit Patel is responsible for task 5