HASH TABLE

My Thao Nguyen

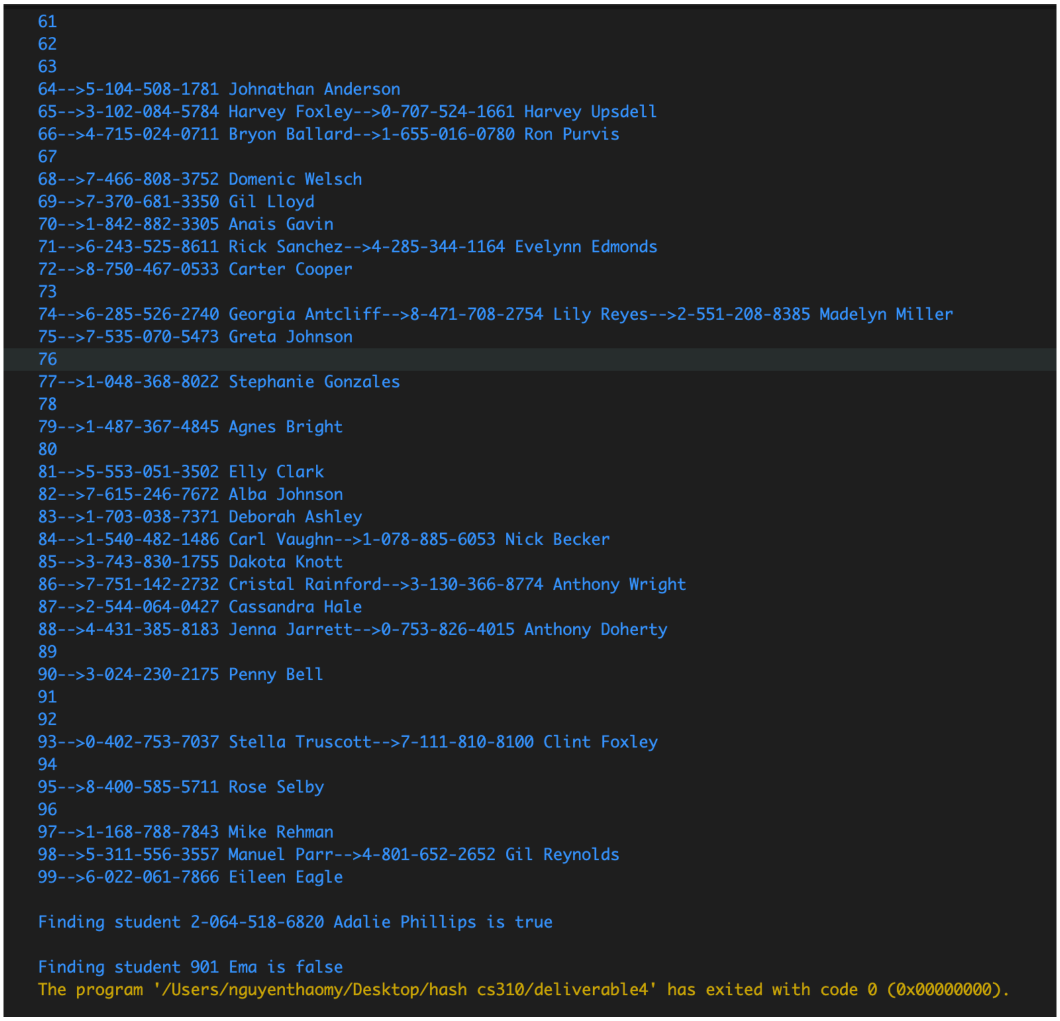
qc53865@truman.edu

Inserting and Finding Objects

Successful search: Adalie Philips – index 4 in input document

Unsuccessful search: Ema

In deliverable 4, by converting the student variable to hash key, then search at this key specifically.

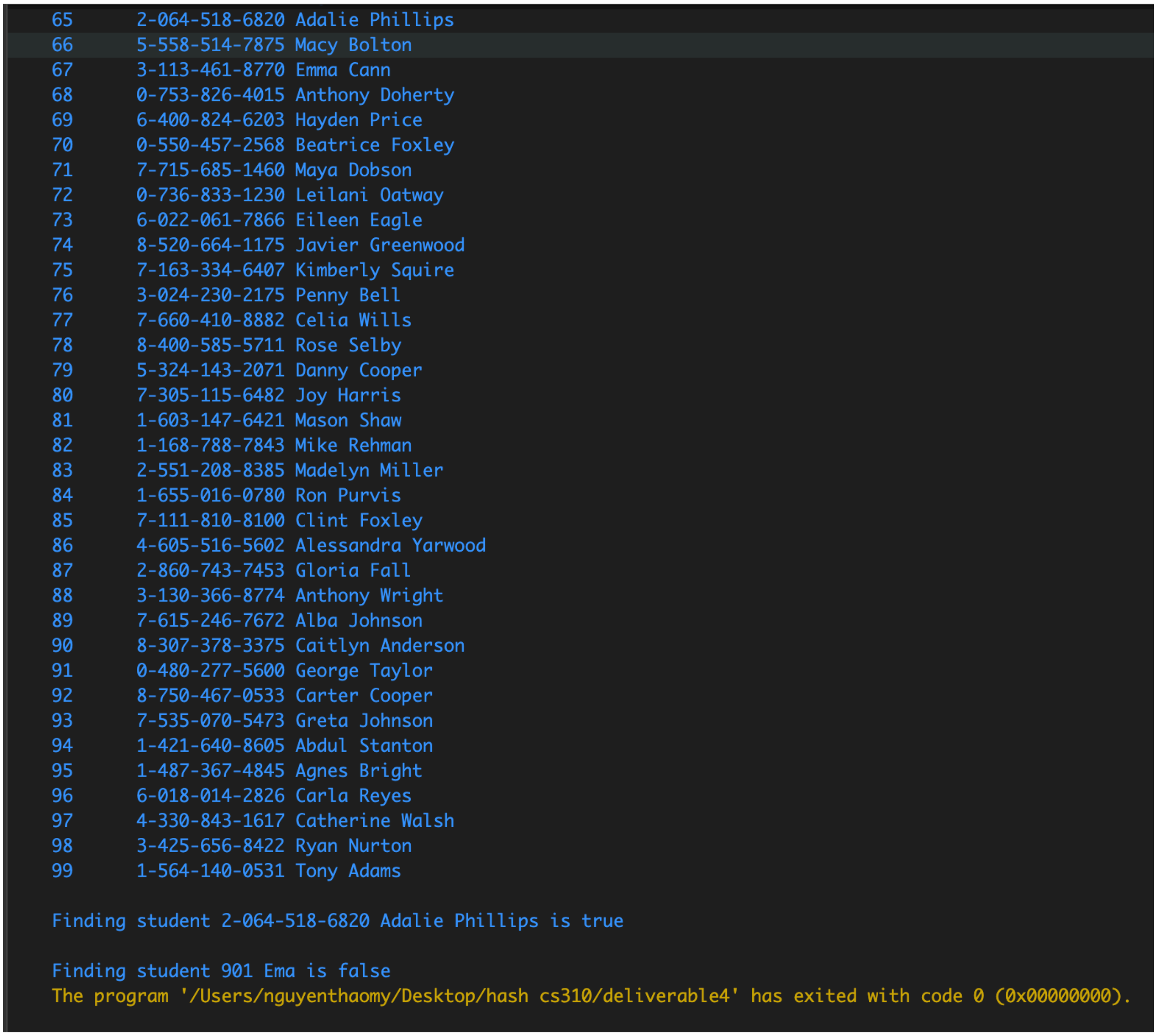


```
61
62
63
64-->5-104-508-1781 Johnathan Anderson
65-->3-102-084-5784 Harvey Foxley-->0-707-524-1661 Harvey Upsdell
66-->4-715-024-0711 Bryon Ballard-->1-655-016-0780 Ron Purvis
67
68-->7-466-808-3752 Domenic Welsch
69-->7-370-681-3350 Gil Lloyd
70-->1-842-882-3305 Anais Gavin
71-->6-243-525-8611 Rick Sanchez-->4-285-344-1164 Evelynn Edmonds
72-->8-750-467-0533 Carter Cooper
73
74-->6-285-526-2740 Georgia Antcliff-->8-471-708-2754 Lily Reyes-->2-551-208-8385 Madelyn Miller
75-->7-535-070-5473 Greta Johnson
76
77-->1-048-368-8022 Stephanie Gonzales
78
79-->1-487-367-4845 Agnes Bright
80
81-->5-553-051-3502 Elly Clark
82-->7-615-246-7672 Alba Johnson
83-->1-703-038-7371 Deborah Ashley
84-->1-540-482-1486 Carl Vaughn-->1-078-885-6053 Nick Becker
85-->3-743-830-1755 Dakota Knott
86-->7-751-142-2732 Cristal Rainford-->3-130-366-8774 Anthony Wright
87-->2-544-064-0427 Cassandra Hale
88-->4-431-385-8183 Jenna Jarrett-->0-753-826-4015 Anthony Doherty
89
90-->3-024-230-2175 Penny Bell
91
92
93-->0-402-753-7037 Stella Truscott-->7-111-810-8100 Clint Foxley
94
95-->8-400-585-5711 Rose Selby
96
97-->1-168-788-7843 Mike Rehman
98-->5-311-556-3557 Manuel Parr-->4-801-652-2652 Gil Reynolds
99-->6-022-061-7866 Eileen Eagle

Finding student 2-064-518-6820 Adalie Phillips is true

Finding student 901 Ema is false
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable4' has exited with code 0 (0x00000000).
```

This screenshot shows the chaining approach and demonstrate successful and unsuccessful searches. Chaining can be seen on multiple indexes – 74, 75, 84, 86,…

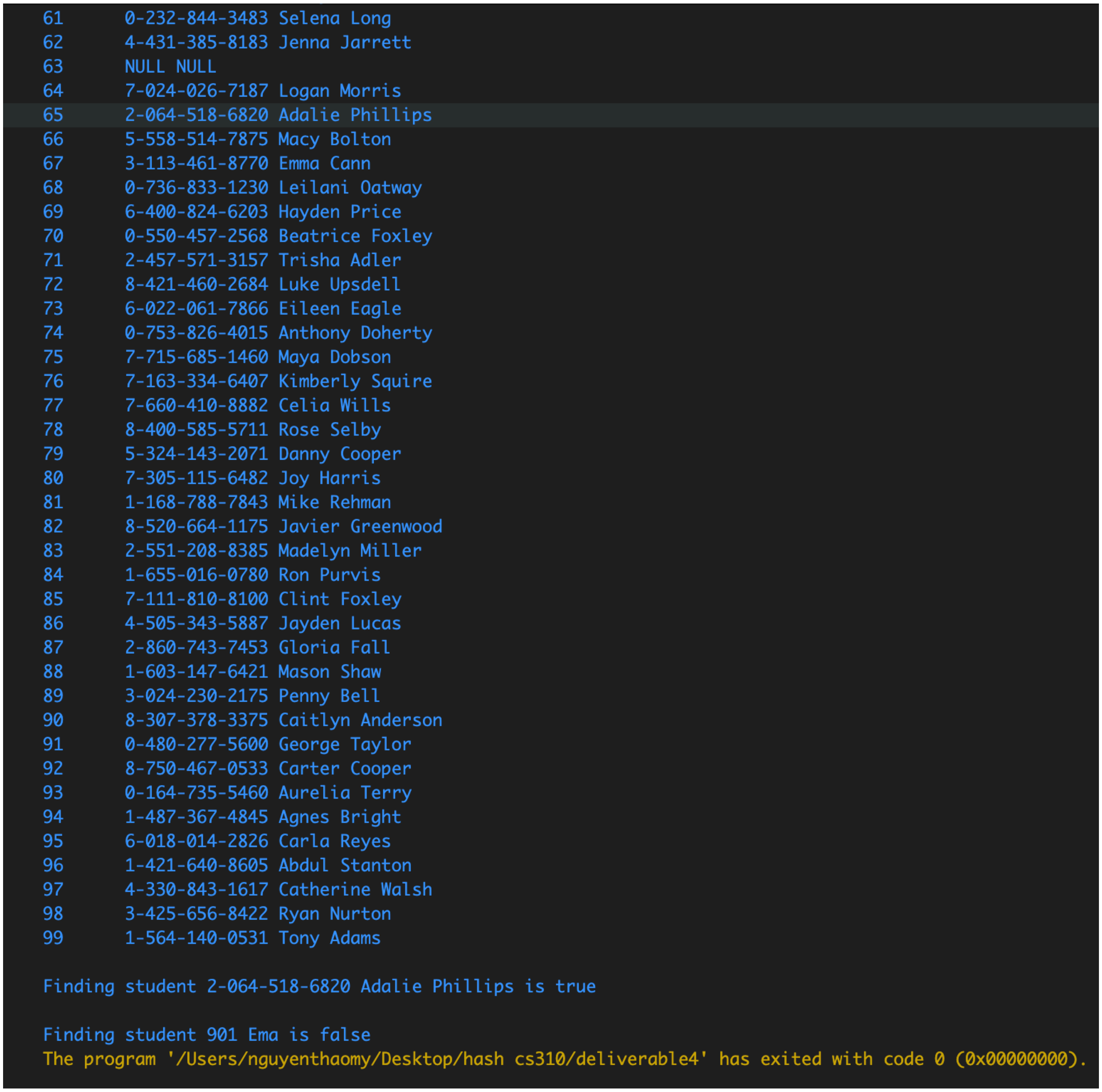


```
65      2-064-518-6820 Adalie Phillips
66      5-558-514-7875 Macy Bolton
67      3-113-461-8770 Emma Cann
68      0-753-826-4015 Anthony Doherty
69      6-400-824-6203 Hayden Price
70      0-550-457-2568 Beatrice Foxley
71      7-715-685-1460 Maya Dobson
72      0-736-833-1230 Leilani Oatway
73      6-022-061-7866 Eileen Eagle
74      8-520-664-1175 Javier Greenwood
75      7-163-334-6407 Kimberly Squire
76      3-024-230-2175 Penny Bell
77      7-660-410-8882 Celia Wills
78      8-400-585-5711 Rose Selby
79      5-324-143-2071 Danny Cooper
80      7-305-115-6482 Joy Harris
81      1-603-147-6421 Mason Shaw
82      1-168-788-7843 Mike Rehman
83      2-551-208-8385 Madelyn Miller
84      1-655-016-0780 Ron Purvis
85      7-111-810-8100 Clint Foxley
86      4-605-516-5602 Alessandra Yarwood
87      2-860-743-7453 Gloria Fall
88      3-130-366-8774 Anthony Wright
89      7-615-246-7672 Alba Johnson
90      8-307-378-3375 Caitlyn Anderson
91      0-480-277-5600 George Taylor
92      8-750-467-0533 Carter Cooper
93      7-535-070-5473 Greta Johnson
94      1-421-640-8605 Abdul Stanton
95      1-487-367-4845 Agnes Bright
96      6-018-014-2826 Carla Reyes
97      4-330-843-1617 Catherine Walsh
98      3-425-656-8422 Ryan Nurton
99      1-564-140-0531 Tony Adams

Finding student 2-064-518-6820 Adalie Phillips is true

Finding student 901 Ema is false
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable4' has exited with code 0 (0x00000000).
```

This screenshot shows the linear approach and demonstrate successful and unsuccessful searches. Linear approach could be seen because all lines are filled with one variable. When search for search value, the probe is taken into consideration as well so that all indexes are searched.



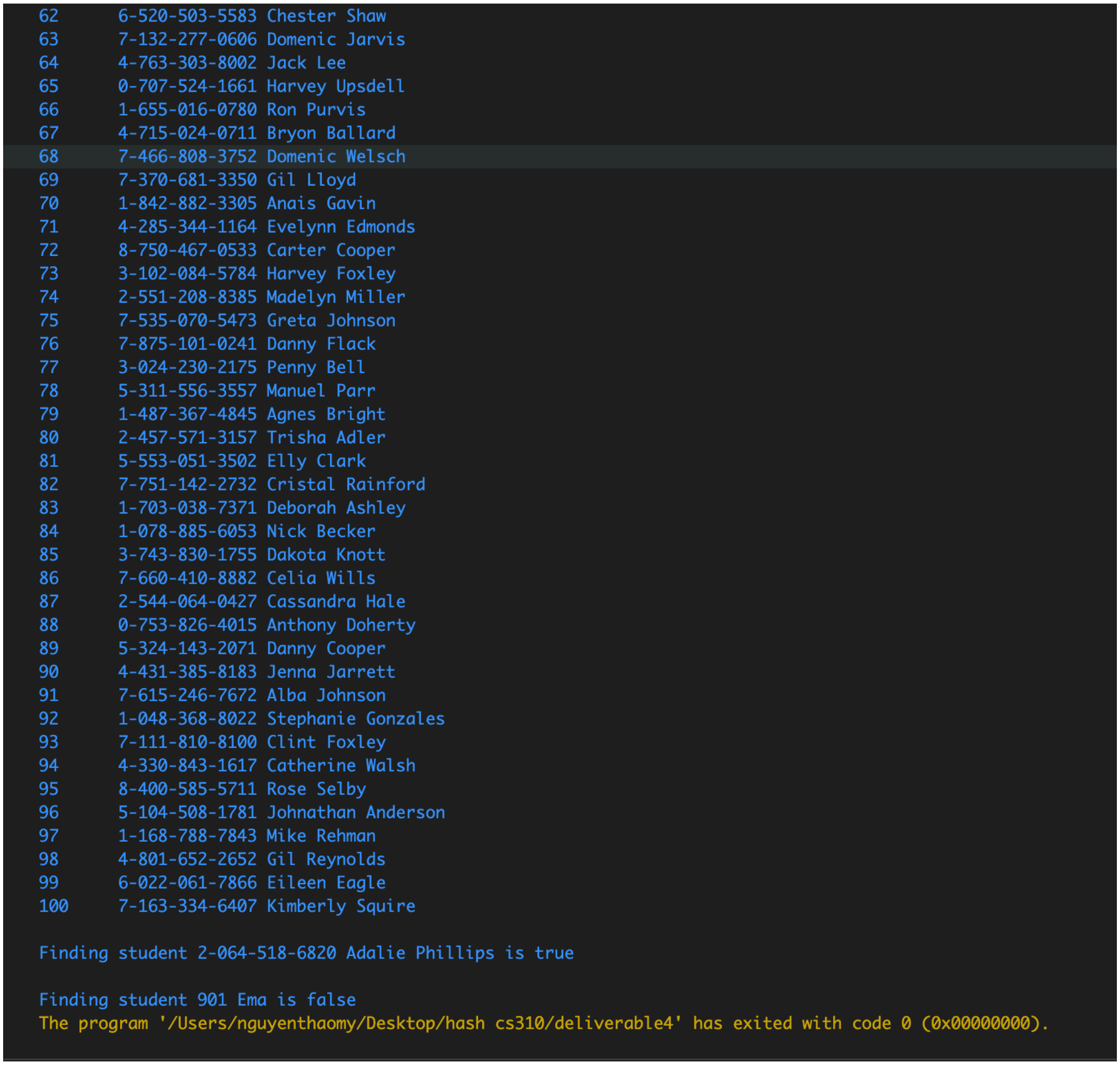
```
61      0-232-844-3483 Selena Long
62      4-431-385-8183 Jenna Jarrett
63      NULL NULL
64      7-024-026-7187 Logan Morris
65      2-064-518-6820 Adalie Phillips
66      5-558-514-7875 Macy Bolton
67      3-113-461-8770 Emma Cann
68      0-736-833-1230 Leilani Oatway
69      6-400-824-6203 Hayden Price
70      0-550-457-2568 Beatrice Foxley
71      2-457-571-3157 Trisha Adler
72      8-421-460-2684 Luke Upsdell
73      6-022-061-7866 Eileen Eagle
74      0-753-826-4015 Anthony Doherty
75      7-715-685-1460 Maya Dobson
76      7-163-334-6407 Kimberly Squire
77      7-660-410-8882 Celia Wills
78      8-400-585-5711 Rose Selby
79      5-324-143-2071 Danny Cooper
80      7-305-115-6482 Joy Harris
81      1-168-788-7843 Mike Rehman
82      8-520-664-1175 Javier Greenwood
83      2-551-208-8385 Madelyn Miller
84      1-655-016-0780 Ron Purvis
85      7-111-810-8100 Clint Foxley
86      4-505-343-5887 Jayden Lucas
87      2-860-743-7453 Gloria Fall
88      1-603-147-6421 Mason Shaw
89      3-024-230-2175 Penny Bell
90      8-307-378-3375 Caitlyn Anderson
91      0-480-277-5600 George Taylor
92      8-750-467-0533 Carter Cooper
93      0-164-735-5460 Aurelia Terry
94      1-487-367-4845 Agnes Bright
95      6-018-014-2826 Carla Reyes
96      1-421-640-8605 Abdul Stanton
97      4-330-843-1617 Catherine Walsh
98      3-425-656-8422 Ryan Nurton
99      1-564-140-0531 Tony Adams

Finding student 2-064-518-6820 Adalie Phillips is true

Finding student 901 Ema is false
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable4' has exited with code 0 (0x00000000).
```

This screenshot shows the quadratic probing approach and demonstrate successful and unsuccessful searches. There are null values because of the secondary clustering.

Secondary clustering is when the probed key value already existed. However, keep on probing does not derive a new value. Hence, the inserted value takes the place of existing value, while many empty space indexes.



```
62      6-520-503-5583 Chester Shaw
63      7-132-277-0606 Domenic Jarvis
64      4-763-303-8002 Jack Lee
65      0-707-524-1661 Harvey Upsdell
66      1-655-016-0780 Ron Purvis
67      4-715-024-0711 Bryon Ballard
68      7-466-808-3752 Domenic Welsch
69      7-370-681-3350 Gil Lloyd
70      1-842-882-3305 Anais Gavin
71      4-285-344-1164 Evelynn Edmonds
72      8-750-467-0533 Carter Cooper
73      3-102-084-5784 Harvey Foxley
74      2-551-208-8385 Madelyn Miller
75      7-535-070-5473 Greta Johnson
76      7-875-101-0241 Danny Flack
77      3-024-230-2175 Penny Bell
78      5-311-556-3557 Manuel Parr
79      1-487-367-4845 Agnes Bright
80      2-457-571-3157 Trisha Adler
81      5-553-051-3502 Elly Clark
82      7-751-142-2732 Cristal Rainford
83      1-703-038-7371 Deborah Ashley
84      1-078-885-6053 Nick Becker
85      3-743-830-1755 Dakota Knott
86      7-660-410-8882 Celia Wills
87      2-544-064-0427 Cassandra Hale
88      0-753-826-4015 Anthony Doherty
89      5-324-143-2071 Danny Cooper
90      4-431-385-8183 Jenna Jarrett
91      7-615-246-7672 Alba Johnson
92      1-048-368-8022 Stephanie Gonzales
93      7-111-810-8100 Clint Foxley
94      4-330-843-1617 Catherine Walsh
95      8-400-585-5711 Rose Selby
96      5-104-508-1781 Johnathan Anderson
97      1-168-788-7843 Mike Rehman
98      4-801-652-2652 Gil Reynolds
99      6-022-061-7866 Eileen Eagle
100     7-163-334-6407 Kimberly Squire

Finding student 2-064-518-6820 Adalie Phillips is true

Finding student 901 Ema is false
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable4' has exited with code 0 (0x00000000).
```

This screenshot shows the double hashing approach and demonstrate successful and unsuccessful searches. To avoid secondary clustering, prime number table size and introduce additional hashing such that

H(key) + probe * g(key) % tablesize

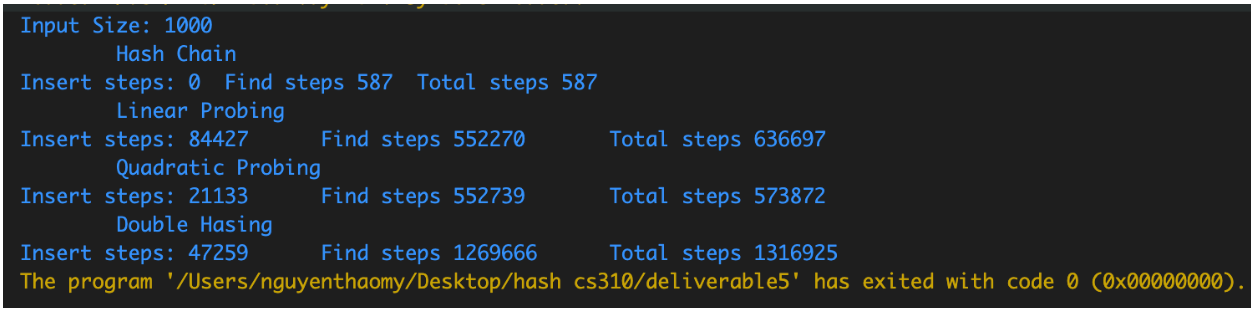
In the program, the function hashFunction_sub is to calculate the g(key) = q – key%q

Q is a prime number from 2 < q < key

Due to prime table size, we have new size of 101. the indexes range 0- 100. Additional hashing avoids hash key to repeat itself.

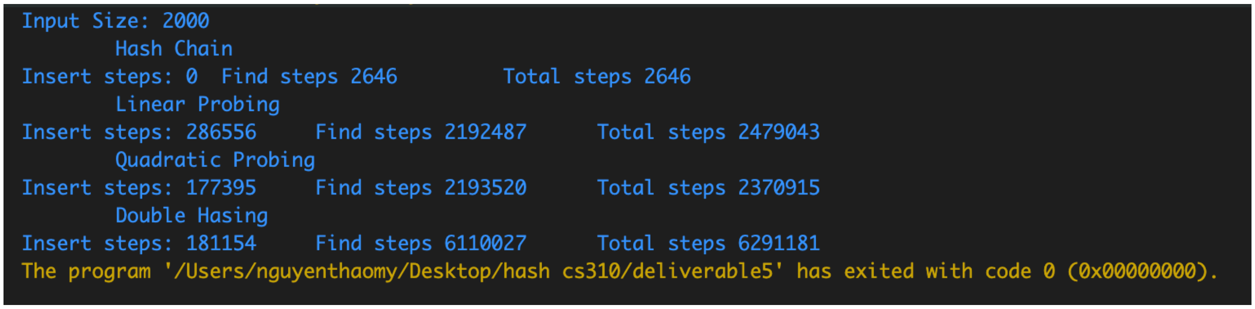
Input Sizes and Steps count

Input Size 1000



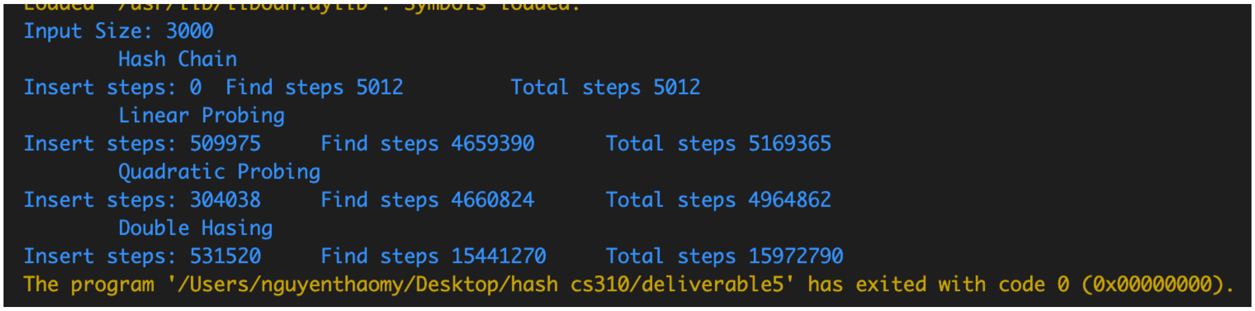
```
Input Size: 1000
        Hash Chain
Insert steps: 0  Find steps 587  Total steps 587
        Linear Probing
Insert steps: 84427      Find steps 552270       Total steps 636697
        Quadratic Probing
Insert steps: 21133      Find steps 552739       Total steps 573872
        Double Hasing
Insert steps: 47259      Find steps 1269666      Total steps 1316925
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable5' has exited with code 0 (0x00000000).
```

Input Size 2000



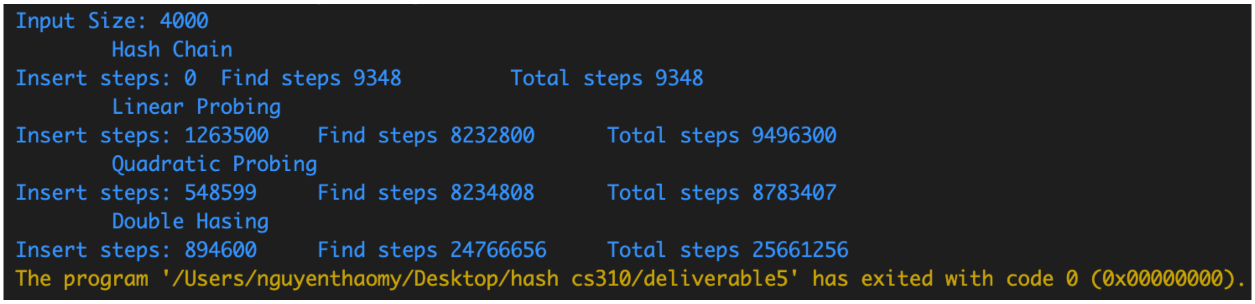
```
Input Size: 2000
        Hash Chain
Insert steps: 0  Find steps 2646         Total steps 2646
        Linear Probing
Insert steps: 286556     Find steps 2192487      Total steps 2479043
        Quadratic Probing
Insert steps: 177395     Find steps 2193520      Total steps 2370915
        Double Hasing
Insert steps: 181154     Find steps 6110027      Total steps 6291181
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable5' has exited with code 0 (0x00000000).
```

Input Size 3000



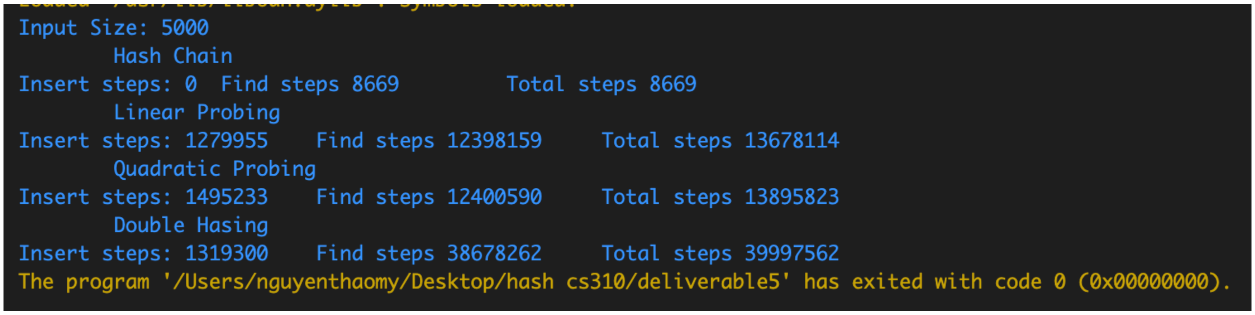
```
Input Size: 3000
        Hash Chain
Insert steps: 0  Find steps 5012         Total steps 5012
        Linear Probing
Insert steps: 509975     Find steps 4659390      Total steps 5169365
        Quadratic Probing
Insert steps: 304038     Find steps 4660824      Total steps 4964862
        Double Hasing
Insert steps: 531520     Find steps 15441270     Total steps 15972790
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable5' has exited with code 0 (0x00000000).
```

Input Size 4000



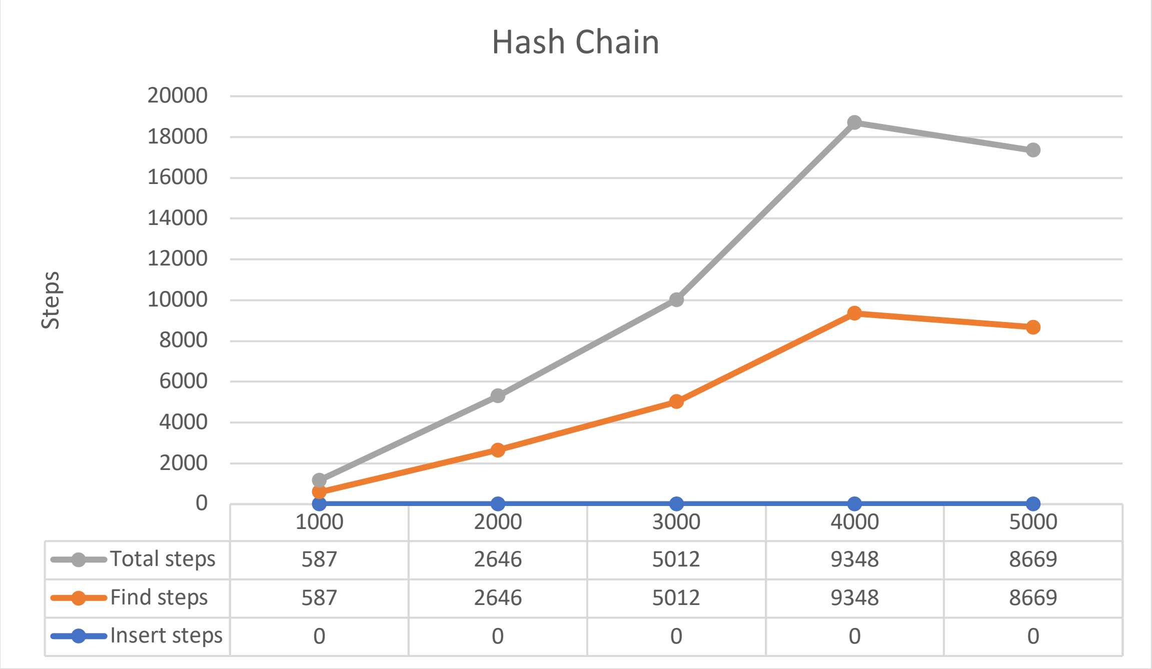
```
Input Size: 4000
        Hash Chain
Insert steps: 0  Find steps 9348         Total steps 9348
        Linear Probing
Insert steps: 1263500    Find steps 8232800      Total steps 9496300
        Quadratic Probing
Insert steps: 548599     Find steps 8234808      Total steps 8783407
        Double Hasing
Insert steps: 894600     Find steps 24766656     Total steps 25661256
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable5' has exited with code 0 (0x00000000).
```

Input Size 5000



```
Input Size: 5000
        Hash Chain
Insert steps: 0  Find steps 8669         Total steps 8669
        Linear Probing
Insert steps: 1279955    Find steps 12398159     Total steps 13678114
        Quadratic Probing
Insert steps: 1495233    Find steps 12400590     Total steps 13895823
        Double Hasing
Insert steps: 1319300    Find steps 38678262     Total steps 39997562
The program '/Users/nguyenthaomy/Desktop/hash cs310/deliverable5' has exited with code 0 (0x00000000).
```

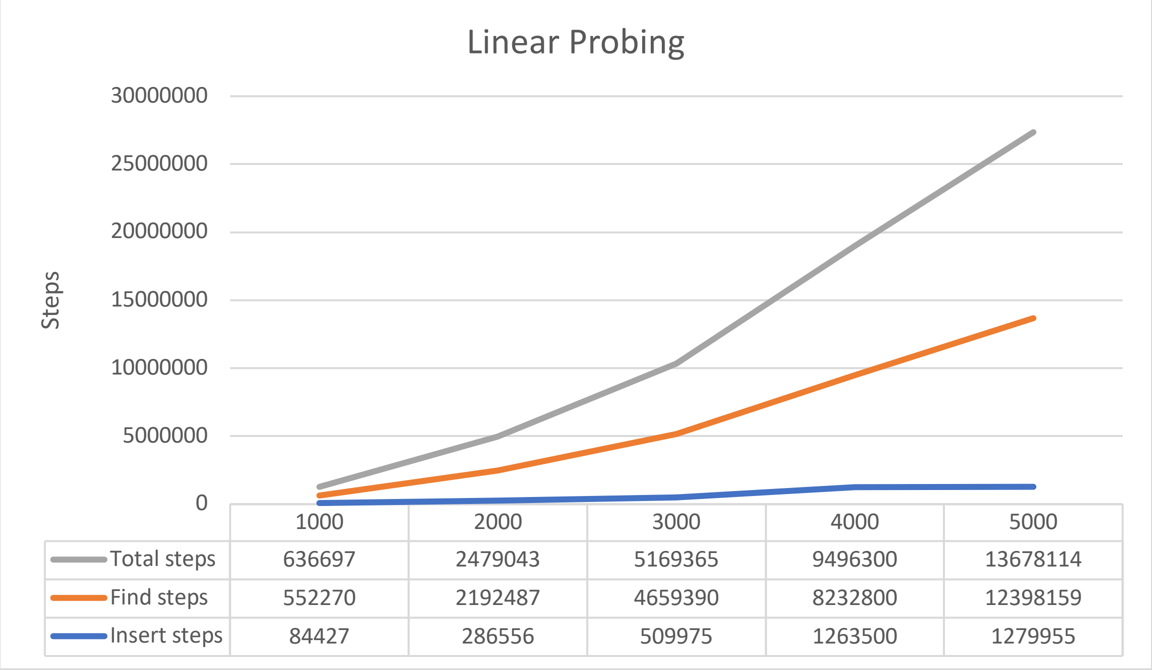
Hash Chain				
Hashtable Size	Input Size	Insert steps	Find steps	Total steps
1100	1000	0 / Constant	587	587
2100	2000	0 / Constant	2646	2646
3100	3000	0 / Constant	5012	5012
4100	4000	0 / Constant	9348	9348
5100	5000	0 / Constant	8669	8669



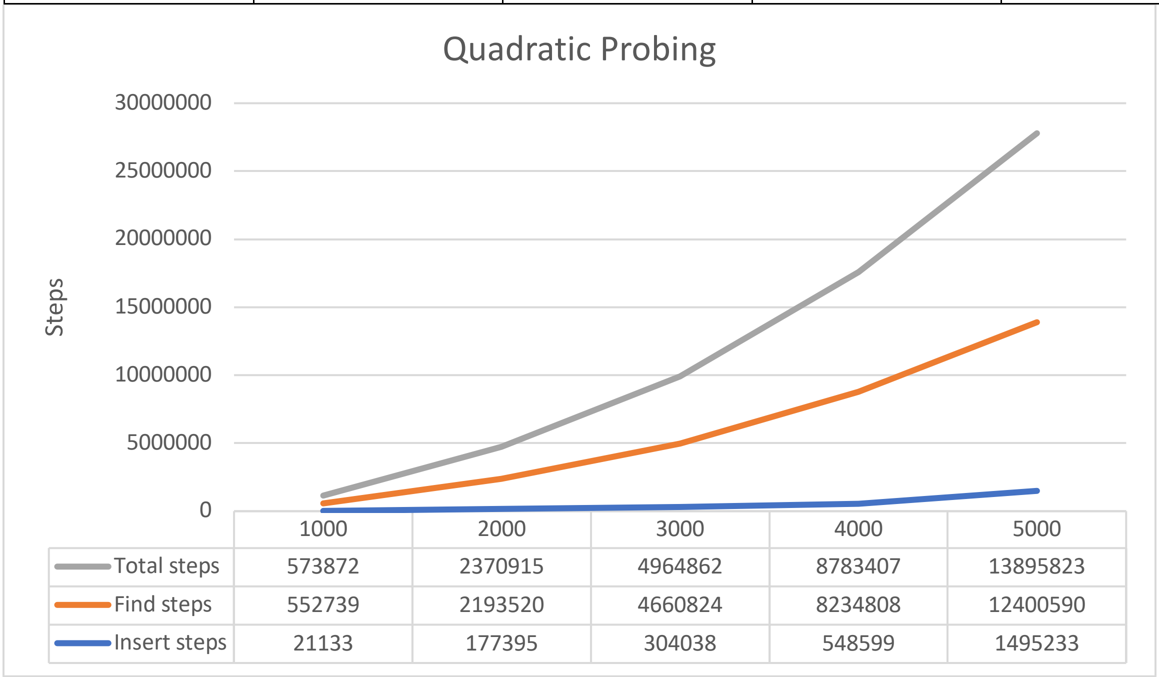
	1000	2000	3000	4000	5000
Total steps	587	2646	5012	9348	8669
Find steps	587	2646	5012	9348	8669
Insert steps	0	0	0	0	0

The changes in hash find is a result of the 5000 input size did not have many chaining / search objects were the only node in its linked list.

Linear Probing				
Hashtable Size	Input Size	Insert steps	Find steps	Total steps
1100	1000	84427	552270	636697
2100	2000	286556	2192487	2479043
3100	3000	509975	4659390	5169365
4100	4000	1263500	8232800	9496300
5100	5000	1279955	12398159	13678114

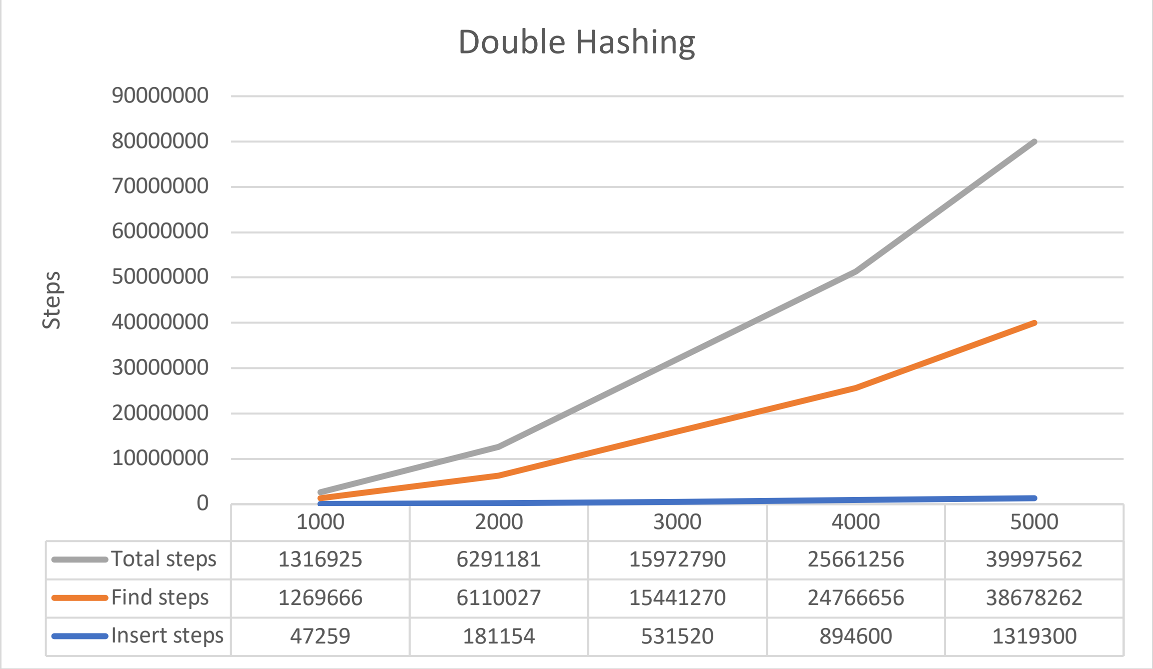


Quadratic Probing				
Hashtable Size	Input Size	Insert steps	Find steps	Total steps
1100	1000	21133	552739	573872
2100	2000	177395	2193520	2370915
3100	3000	304038	4660824	4964862
4100	4000	548599	8234808	8783407
5100	5000	1495233	12400590	13895823



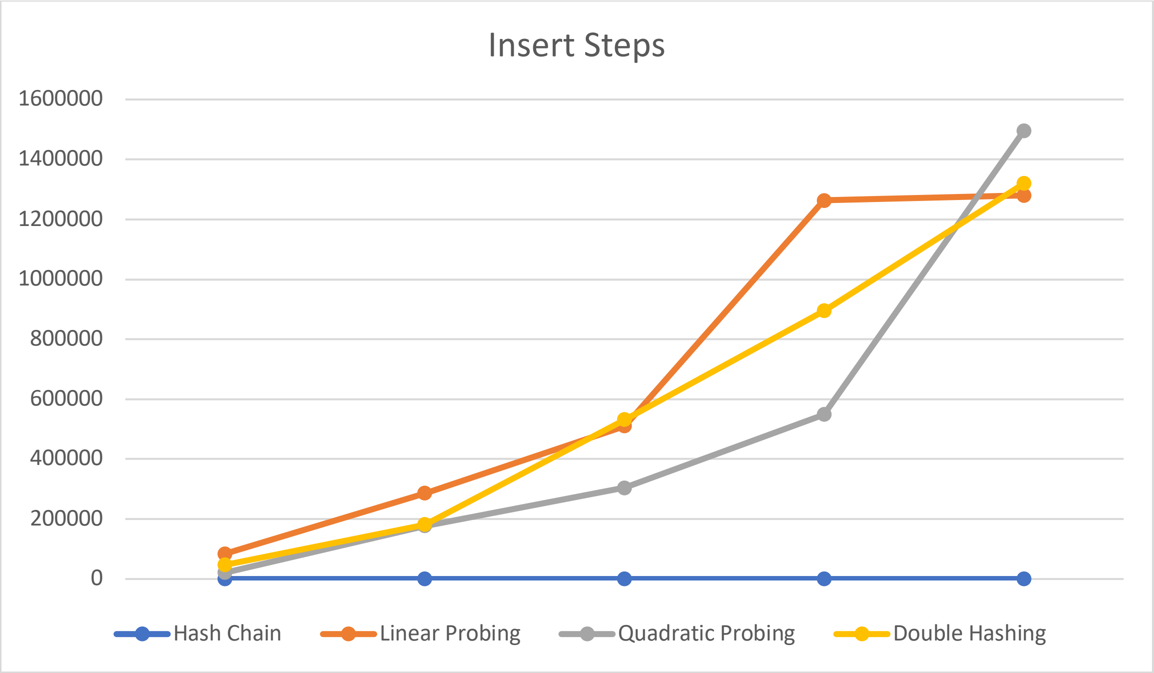
	1000	2000	3000	4000	5000
Total steps	573872	2370915	4964862	8783407	13895823
Find steps	552739	2193520	4660824	8234808	12400590
Insert steps	21133	177395	304038	548599	1495233

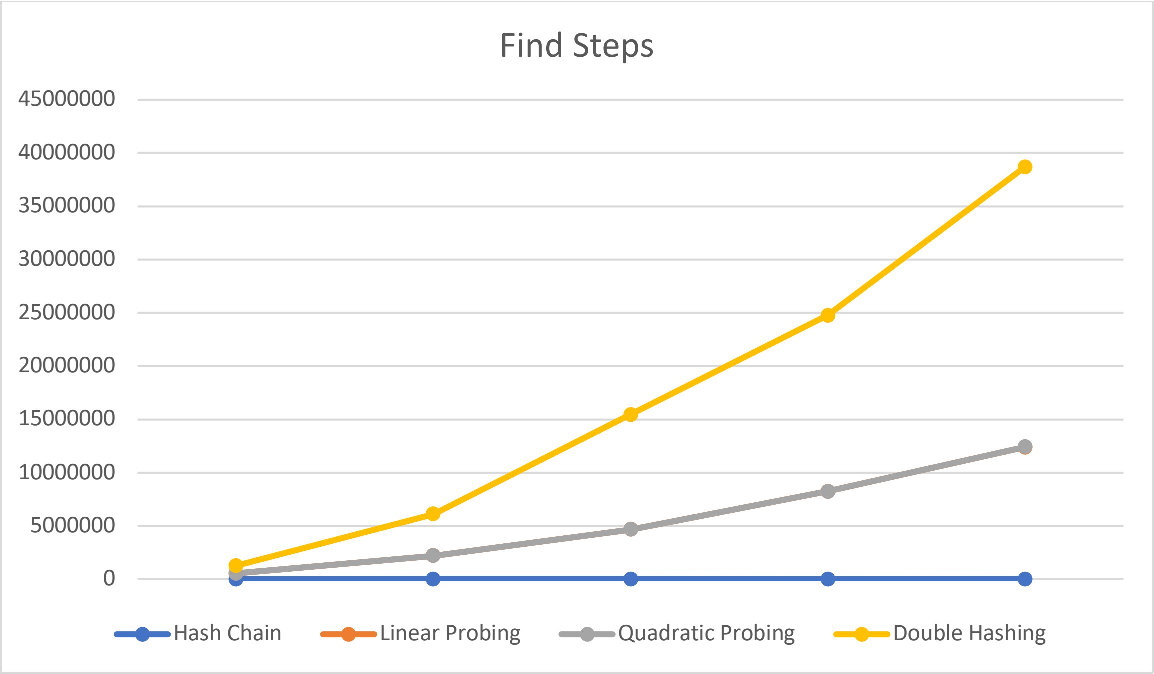
Double Hashing				
Hashtable Size	Input Size	Insert steps	Find steps	Total steps
1100	1000	47259	1269666	1316925
2100	2000	181154	6110027	6291181
3100	3000	531520	15441270	15972790
4100	4000	894600	24766656	25661256
5100	5000	1319300	38678262	39997562



	1000	2000	3000	4000	5000
Total steps	1316925	6291181	15972790	25661256	39997562
Find steps	1269666	6110027	15441270	24766656	38678262
Insert steps	47259	181154	531520	894600	1319300

Graph for steps algorithms





a) Which hash table algorithm requires the least number of steps to store the Student objects? Comment why that is the case.

Hash chain requires the least number of steps. This is because the insert function has 2 function calls (hashFunction – to calculate h(k) and insert – put the information at index = h(k)). Both functions are constant run time because there is no iteration. Hence, the insert function that composes of these two constant time functions is also a constant time function. O(1). That is, the steps are independent from input size.

b) Which hash table algorithm requires the most number of steps to store the Student objects? Comment why that is the case.

For most of the cases, Linear Probing has the highest number of steps. This is the case because to store a value with the same hash key, the function will iterate from 0 – (tableSize -1) to find suitable place for the value. However, since the function has to go through all the values until it finds the vacant index, while loop iteration increases runtime.

c) Compare the performance of the hash tables when the input size was 1000 compared to 5000.Why does the insert algorithm requires more steps to store the objects when the input size is 5000 compared to the input size 1000?
The steps are determined by how many times an iteration occurs. For instance, a simple for loop that goes from 0 – n has run n times. From the graph, linear, quadratic and double hashing all require loops for its insert algorithm. Hence, these algorithms have runtime that is not constant time. In other words, these algorithms have some dependency on input size. Increasing in input size will cause an increase in steps occurred.