Section 5: Hashing & Sorting

0. Hash... Browns?

For the following scenarios, insert the following elements in this order: 7, 9, 48, 8, 37, 57. For each table, TableSize = 10, and you should use the primary hash function $h(k) = k$. If an item cannot be inserted into the table, please indicate this and continue inserting the remaining values.

a) Linear Probing - Insertion

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

Linear Probing - Delete 37, 7, 57

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

b) Quadratic Probing

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

c) Separate chaining hash table - Use a linked list for each bucket. Order elements within buckets in any way you wish.

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

1. Double Double Toil and Trouble

a) Describe double hashing.

b) List 2 cons of quadratic probing and describe how one of those is fixed by using double hashing.

c) Compare open addressing and separate chaining.

2. Sorting Hat

Suppose we sort an array of numbers, but it turns out every element of the array is the same, e.g.,{17, 17, 17, ..., 17}. (So, in hindsight, the sorting is useless.)

a) What is the asymptotic running time of **insertion** sort in this case?

b) What is the asymptotic running time of **selection** sort in this case?

c) What is the asymptotic running time of **merge** sort in this case?

d) What is the asymptotic running time of **quick** sort in this case?