Name: Jeremiah Webb

Module 7: Hash Tables - Worksheet

Problem 1: For this problem, utilize the string hash technique covered within the lecture video for the following keys. We shall use the indices computed in the next two problems.

Let m=11 and let R=13

a. hash(“Bilbo”); _0 6 10 8 6_____

b. hash(“Frodo”); __4 1 3 7 4_____

c. hash(“Sam”); __6 10 8_____

d. hash(“Gimli”); __5 5 9 5 5 _____

e. hash(“Merry”); _____0 2 8 9 7_____

f. hash(“Pippin”); ___3 1 4 10 4 8_____

Problem 2: Insert each of the items from page 1 into the hash table below using **separate chaining**. Draw each chain as a list-like data structure.

Key	hash(Key)
Bilbo	6
Frodo	4
Sam	8
Gimli	5
Merry	7
Pippin	8

0	Chain (Dynamic size)	
1		
2		
3		
4	Frodo	
5	Gimli	
6	Bilbo	
7	Merry	
8	Sam	Pippin
9		
10		

How many collisions occurred with this algorithm? i.e. how many elements were written to buckets already occupied (if any)? Sam and Pippin are in collision.

1

Problem 3: Insert each of the items from page 1 into the hash table below using **linear probing**. Draw each chain as a list-like data structure.

Key	hash(Key)
Bilbo	6
Frodo	4
Sam	8
Gimli	5
Merry	7
Pippin	8

0	Buckets (b=1)
1	
2	
3	
4	Frodo
5	Gimli
6	Bilbo
7	Merry
8	Sam
9	Pippin
10	

How many collisions occurred with this algorithm? i.e. how many elements were written to buckets already occupied (if any)? Sam and Pippin in collision, Pippin moved to index 9

1

_____

Problem 4: Update the linear probing table after table.delete("Sam"); Start with your solution from Problem 3 and then show all changes made to the table to handle the deletion and restoration of the hash table.

Starting

0	
1	
2	
3	
4	Frodo
5	Gimli
6	Bilbo
7	Merry
8	Sam
9	Pippin
10	

Deletion of Sam

0	
1	
2	
3	
4	Frodo
5	Gimli
6	Bilbo
7	Merry
8	
9	Pippin
10	

Movement of Pippin
Key to redo = Pippin

0	
1	
2	
3	
4	Frodo
5	Gimli
6	Bilbo
7	Merry
8	Pippin
9	
10	