

PA 5 due tomorrow
PR2 Late / Resubmit due tomorrow

Hash Function (same as previous)

```
int getIndex(String k) {
    return k.length();
}
```

of buckets = 4
(i.e. the size of the array)

expandCapacity() called in set()

LoadFactor = 0.75

```
set("Smith", 1);
set("Johnson", 2);
set("Williams", 3);
set("Brown", 4);
set("Jones", 5);
set("Garcia", 6);
set("Miller", 7);
set("Davis", 8);
set("Rodriguez", 9);
set("Martinez", 10);
```

Draw the picture of the HashTable using Separate Chaining (using expandCapacity)

Does the run-time change with expandCapacity?

What is the run-time for this HashTable (do picture first):

set():
Worst Case: $\Theta(N^2)$
Best Case: $\Theta(1)$

What conditions make up the best case for set()? *no expand capacity, extra or many extra buckets*

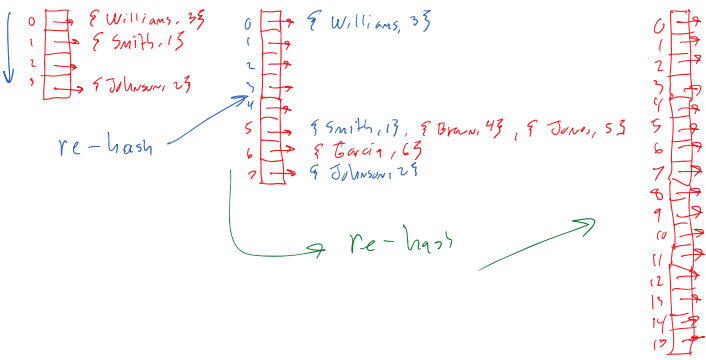
get():
Worst Case: $\Theta(N)$
Best Case: $\Theta(1)$

What conditions make up the best case for get()? *even distribution, 0, 1, or 2 elements per bucket*

Why is the hash function important?
*even distribution
less collisions*

LoadFactor = 0.75

	hash	index	LP
set("Smith", 1);	5	1	5/8 → 0
set("Johnson", 2);	7	3	7/8 → 0.5
set("Williams", 3);	8	0	8/8 → 0.5
set("Brown", 4);	5	1	5/8 → 0.5
set("Jones", 5);	6	2	6/8 → 0.5
set("Garcia", 6);	6	2	6/8 → 0.5
set("Miller", 7);	7	3	7/8 → 0.5
set("Davis", 8);	8	0	8/8 → 0.5
set("Rodriguez", 9);	9	1	9/8 → 1.125, 6/4
set("Martinez", 10);	10	2	10/8 → 1.25



clustering
↓
bad hash function