# Hash Table

1. Map interface:

```
public interface Map<Key, Value> {
        public void set(Key k, Value v);
        public boolean contains(Key k);
        public Value get(Key k);
        public int size();
}
```

2. Example applications

3. Implementations

4. Recall bucket sort
    a. Example: Sort the following numbers: **57 25 38 15 29 42**
    Using their first digit to predict location into the following buckets:

0's 10's 20's 30's 40's 50's 60's

    b. **Underlying array of lists**
    c. **Function**
    d. Want **O( 1 )** elements per bucket and **O( n )** total buckets

5. Hash Table
    a. Example: Map from first name to last name
        i. Strings hashed by second letter, 5 buckets
        ii. Elements: Joey Kiernan, Elisa Chang, Jack Wadden, Emily Yu, Jesse Levitt
        iii. Build the data structure

**Decimal ASCII Codes**

| *Char* | *Code* | *Code % 5* | *Char* | *Code* | *Code % 5* | *Char* | *Code* | *Code % 5* |
|---|---|---|---|---|---|---|---|---|
| a | 97 | 2 | j | 106 | 1 | s | 115 | 0 |
| b | 98 | 3 | k | 107 | 2 | t | 116 | 1 |
| c | 99 | 4 | l | 108 | 3 | u | 117 | 2 |
| d | 100 | 0 | m | 109 | 4 | v | 118 | 3 |
| e | 101 | 1 | n | 110 | 0 | w | 119 | 4 |
| f | 102 | 2 | o | 111 | 1 | x | 120 | 0 |
| g | 103 | 3 | p | 112 | 2 | y | 121 | 1 |
| h | 104 | 4 | q | 113 | 3 | z | 122 | 2 |
| i | 105 | 0 | r | 114 | 4 | | | |

*Next time: choosing a good hash function, and the open addressing scheme discussed in the textbook*