

**COMP 587**  
**Spring 2020**

**Property-Based Testing**

1.) For this problem, you're interested in applying property-based testing to the `HashMap` class in Java. Relevant code is shown below:

```
public class HashMap<K, V> {  
    // Creates an empty map  
    public HashMap() { ... }  
  
    // Inserts a key/value pair in the map. Overwrites the  
    // existing key/value pair, if there is one.  
    public void put(K key, V value) { ... }  
  
    // Gets the value associated with the key, or null if the  
    // given key is not in the map.  
    public V get(K key) { ... }  
  
    // Gets the number of key/value pairs in the map.  
    public int size() { ... }  
}
```

1.a.) Using any preferred manner (English, psuedocode, math), write out properties which should hold of `put`, `get`, and `size`. It's ok (and expected) that not all properties will involve all methods. You should define at least 4 properties in all; it's ok if you think there is redundancy.

1.b.) Using `junit-quickcheck`, implement one of the properties you defined. As a hint, for the keys and values in the map, values of any type should work, and `junit-quickcheck` already has generators of `String` and `int`.

1.c.) Using `junit-quickcheck`, implement one other property you defined.