

3.1.6

Number of `put()` calls : $P(w, D) = W$

The method `put()` is called once for every word in `StdIn`, regardless of whether or not the symbol table already contains the word.

Number of `get()` calls: $G(w, D) = (W - D) + 2D + 1$

$$G(w, D) = W + D + 1$$

For the first part in the while loop, the method `get()` is called for every non-distinct word, so that would be represented as $(W - D)$.

For the second part (the for loop), the method is called twice in every loop for the boolean comparison in the if statement. Since the loop repeats itself D times, the `get()` method is called $2D$ times.

And the `StdOut` calls the method once, so $+1$.

A symbol-table client

```
public class FrequencyCounter
{
    public static void main(String[] args)
    {
        int minlen = Integer.parseInt(args[0]); // key-length cutoff
        ST<String, Integer> st = new ST<String, Integer>();
        while (!StdIn.isEmpty())
        { // Build symbol table and count frequencies.
            String word = StdIn.readString();
            if (word.length() < minlen) continue; // Ignore short keys.
            if (!st.contains(word)) st.put(word, 1);
            else
                st.put(word, st.get(word) + 1);
        }
        // Find a key with the highest frequency count.
        String max = "";
        st.put(max, 0);
        for (String word : st.keys())
            if (st.get(word) > st.get(max))
                max = word;
        StdOut.println(max + " " + st.get(max));
    }
}
```