

```

1  // Lab 7: BinaryVsLinear
2  // I liked this one. Graphs are fun.
3  // Andrea Smith
4  // CSCI 1913
5
6  class BinaryVsLinear
7  {
8
9      private static int linearSearch(int key, int[] keys)
10     {
11         for (int i=0; i < keys.length; i += 1)
12         {
13
14             if (keys[i] == key)
15             {
16                 return i + 1; // i is one less than the number of
17                 • comparisons, so i is the counter variable
18             }
19         }
20         return -1; // failure case if key is not in array keys
21     }
22
23     private static int binarySearch(int key, int[] keys)
24     {
25         int count = 0;
26         int left = 0;
27         int right = keys.length - 1;
28         while (left <= right)
29         {
30             int mid = left + (right - left) / 2;
31             count = count + 1;
32             // If the key is the middle value, you found it already, do
33             • nothing.
34             if (keys[mid] == key)
35             {
36                 return count;
37             }
38
39             // If key is greater, ignore left half and only search right
40             • side.
41             else if (keys[mid] < key)
42             {
43                 // count = count + 1;
44                 left = mid + 1;
45                 // return count;
46             }
47         }
48     }
49 }

```

```

45
46     // If key is smaller, ignore right half and only search left
    • side.
47     else {
48         right = mid - 1;
49         // return count;
50     }
51 }
52
53     return -1; // Failure case-- if program reaches here, keys was
    • not in the array.
54 }
55
56 public static void main(String[] args)
57 {
58     for (int length = 1; length <= 30; length += 1)
59     {
60         int[] array = new int[length];
61         for (int index = 0; index < length; index += 1)
62         {
63             array[index] = index;
64         }
65
66         double linearTotal = 0.0;
67         double binaryTotal = 0.0;
68         for (int element = 0; element < length; element += 1)
69         {
70             linearTotal += linearSearch(element, array);
71             binaryTotal += binarySearch(element, array);
72         }
73
74         double linearAverage = linearTotal / length;
75         double binaryAverage = binaryTotal / length;
76         System.out.println(length + " " + linearAverage + " " +
    • binaryAverage);
77
78     }
79 }
80 }
81

```