**CIS 360 Lab #1: Implementing Algorithms**

**Name \_\_\_\_Tanmay Sonawane\_\_\_\_\_\_\_\_\_ ID \_\_\_\_\_01996456\_\_\_\_\_\_\_**

Sequential Search Algorithm

**A1.1-Sequ_Search.tiff**Input:

* positive integer n
* array of keys (integers) S, indexed from 1 to n
* a key x

Output: the location of x in S (0 if x is not in S)

**A1.5.tiff**Binary Search Algorithm

Input and Output are the same as above, except the input array S must be sorted.

*Note: The ⌊ ⌋ symbols represent the floor function.*

**The following results are from one trial run:**

**Task 1.** Implement Sequential Search Algorithm, test it on 1000; 100,000 and 1,000,000 randomly generated numbers, and record the number of keys that were checked until x was found.

Count for 1,000: \_\_\_72\_\_\_\_\_\_\_\_\_\_\_\_

Count for 100,000: \_\_14153\_\_\_\_\_\_\_\_\_

Count for 1,000,000: \_\_335496\_\_\_\_\_\_

**Task 2**. Implement Binary Search Algorithm, test it on 1000; 100,000 and 1,000,000 randomly generated (**sorted**) numbers, and record the number of keys that were checked until x was found.

Count for 1,000: \_\_\_\_\_8\_\_\_\_\_\_\_\_

Count for 100,000: \_\_\_\_15\_\_\_\_\_\_\_\_\_

Count for 1,000,000: \_\_\_18\_\_\_\_\_\_\_\_\_