

B3209E – Algorithm Design

Lab 01: Linear Search vs. Binary Search

8 October 2025

Objective: Analyze the computational complexity of two different algorithms (linear search and binary search) for the problem of finding whether an integer is in a sorted sequence.

Expected time: 60 minutes.

Input: Sorted array A , number x .

Task: Identify whether a number is in a sorted sequence (return 1/True) or not (return -1/False).

Output: 1/True if x is in A , -1 if x is not in A .

Tasks:

1. Download the material of Lab 01. Inspect the code for **Binary Search** algorithm and test/verify that the code is working as intended. Fix any problems, if any.
2. Automate the creation of a sorted list that will eventually represent the input of the problem.
3. Develop **Linear Search**.
4. Compute the time complexity (Big-O) of each of the algorithms.
5. Compute and plot the computational time (in milliseconds)* of each of the two algorithms varying the size of the input array size $n = \{1000, 10000, 50000, 100000, 200000\}$ **. See Table below.

*Import time and find time spent by algorithm;

**Repeat each evaluation k times and report the average.

Plot this
↓

n	Eval.1	Eval.2	Eval...	Eval...	Eval. k	Average
1000						
10000						
50000						
100000						
200000						

↑

As a function of this