

University of the Witwatersrand  
School of Computer Science and Applied Mathematics  
COMS3005A:  
Advanced Analysis of Algorithms  
Lab 0 (Classical Search)

## 1 Instructions

For this lab you will generate a list of numbers and implement the linear search algorithm for the case where the key may not exist in the list.

### 1.1 Implementation

I suggest you begin by defining a static list to implement the linear search. Let's call this list *myList*. Implement the linear search function so that it takes in *myList* and the desired key in the list as arguments. Note that you do not need to assume that the key is in the list. Once you have the linear search function you should move on to generating lists of desired length where each item is unique. To generate the list of numbers you will need to write a function which:

1. accepts an argument specifying how long the list should be ( $n$ ).
2. samples an integer ( $k$ ).
3. if  $k$  is not already in the list (hint: use your linear search) then append it.
4. iterate by going back to step 2 until the desired length array is reached.

### 1.2 Experiments

Please perform the following experiments using the two functions above:

- Experiment on the best case performance of the linear search algorithm for lists of different sizes (different  $n$ ).
- Experiment on the worst case performance of the linear search algorithm for lists of different sizes (different  $n$ ).
- Experiment on the average case performance of the linear search algorithm for lists of different sizes (different  $n$ ).

Note that you are likely to have to use lists of integers which are quite big (i.e. really large  $n$ ) in order to get reasonable running times for the worst and average case performances of the algorithm.

### 1.3 Results

Plot the best, worst and average case graph to verify the growth rate is  $\mathcal{O}(1)$ ,  $\mathcal{O}(n)$  and  $\mathcal{O}(n)$  respectively. Place the  $n$  variable on the x-axis and then plot the time taken to search for the item on the y-axis. Finally, confirm that the average case growth rate indeed has a smaller constant factor than in the worst case.

### 1.4 Submission requirements

This lab is not for marks and so there is no due date. However, the code you create here will be helpful for later submissions, so I recommend that you complete this lab soon and use it to get feedback on your early results.

## **2 On the Use of ChatGPT**

These labs do not count much individually for the course, and yet they are crucial to understand the material and prepare for other assessments. Thus, I urge you to take them seriously and engage with the material. The use of generative AI is not permitted to complete the labs. The code you submit will be checked for plagiarism. Please check the course outline for a discussion on the academic integrity expectations of the course. Additionally, the Lab Exam will follow the format of these labs extremely closely and so it is to your benefit to make sure you are able to reproduce these algorithms in a closed-book settings.