Name, SID, Date ..... Hints .....

In Class Assignment 2: Binary Search

Benjamin Sanders, MS November 25, 2020

1 Introduction

You will need to work individually to complete this assignment. Write your name at the top of all pages for this assignment. Turn in all work to Blackboard on or before the deadline to receive credit.

You may use additional libraries and online resources, if you get them approved in writing, over email, from the instructor first. If you have received approval from the instructor, write the approved libraries and any references in the space below.

.....
.....
.....
.....

2 Assignment Description

2.1 Big Picture

Binary search finds a value in a sorted array more quickly than considering every value in order.

2.2 Algorithm Implementation

Implement the following algorithm in Java, using the Vector data structure for any 1-D array, 2-D array, or linear algebra purposes.

```
BINARY-SEARCH(A, n, T)
1  L = 0
2  R = n − 1
3  while L ≤ R
4       m = ⌊(L + R)/2⌋
5       if A[m] < T
6            L = m + 1
7       elseif A[m] > T
8            R = m − 1
9       else
10           return m
11 return −1              // invalid, not found
```

Note that A is an array of values, of length n. Note that T is the value to search for.

2.3 Time Complexity Analysis

Where n is the number of data points in A, analyze the time complexity of the given algorithm with respect to n. Write the result of your analysis in big-O notation, i.e. $O(n^2)$ in the space below.

.....

2.4 Space Complexity Analysis

Where n is the number of data points in A, analyze the space complexity of the given algorithm with respect to n. Write the result of your analysis in big-O notation, i.e. $O(n \cdot log(n))$ in the space below.

.....

Name, SID, Date .....

2.5 Optimize the Algorithm for a Purpose

Choose an optimization, either in time or in space, for the given algorithm. Write your intended big-O notation, i.e. $O(1)$, $O(n)$, etc., in the space below, and write N/A in the other space.

- Time Complexity: .....
- Space Complexity: .....

What application would benefit from the purpose of the above optimization? Why? Write two sentences to answer these questions in the space below.

.....
.....
.....
.....

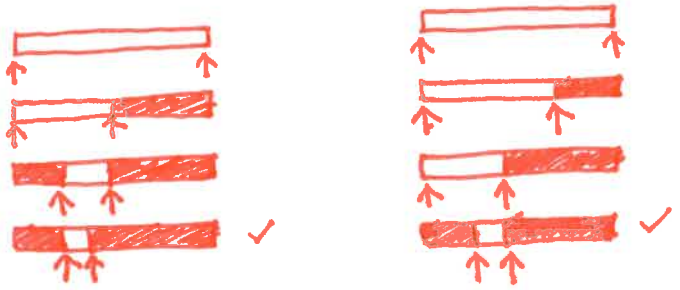
2.6 New Algorithm Design and Implementation

In the space below, design an algorithm that achieves the same purpose of the given algorithm, but includes the optimization you have specified above. Use pseudocode written in a style similar to the given algorithm, and implement it in Java. You may use as many additional pages as necessary for this purpose.

Time:

Original: Cut in ½.

Proposed: Cut in ⅓.



- What application would benefit from an algorithm that has a chance to be faster, but also a chance to be slower?
- Why stop with considering cutting in ⅓? Why not consider cutting in ¼, or ⅕, etc? Consider the cases: ½, ⅓, ¼, ... 1/n, where n is the # of data points. Where is the best choice?

Space:

Can I "compress" my data to ~~be~~ take up less than linear space?

- Consider: {2, 4, 6, 8, ... 2042, 2044, ...} → 2i. I can represent this in closed-form as 2i. This is a special case.
- What algorithm does "zip" use to minimize footprint but maintain data integrity? Huffman encoding.
- Alternately, I could use a lossy compression method if data integrity is not crucial. For example: JPEG encoding.
- What applications would lossless encoding (Huffman) vs lossy encoding (JPEG) be useful for?

3 What to Turn In

Turn in one PDF or Word document on Blackboard, containing the following items.

1. All pages scanned or photographed of the In Class Assignment completed document.
2. Any additional pages you used to complete the assignment.
3. All code created for the assignment, along with test cases.
4. One statement indicating which parts of your implementation(s) are working, and which parts are not.
5. Screenshots demonstrating the code working, if it is working.