

Announcements

Thursday Extras: CS Commons on Thursdays @ 4:00 pm

Monday Extra, Oct. 8, 4:15 pm, Science 2022

Careers in CS and Informal Talk by Grinnell CS Alumni

Thursday: CS Student Research Group

External Reviewers on campus: Thursday/Friday

CSC 207.02 Schedule

Candidates visiting in next 2 weeks

In-class conversations

questions

clicker questions

discussion

Classes, Abstract Classes, Interfaces

1. Suppose a segment of a Java program includes a private field definition, such as `private String str;`
2. Suppose a segment of a Java program includes a static field definition, such as `static String str;`
3. Suppose a segment of a Java program includes a method prototype—a method signature, but no implementation details.
4. Suppose a segment of a Java program includes a full implementation of a Java method.

The code segment could be:

- A. an Interface only
- B. an abstract class only
- C. a class only
- D. an Interface or abstract class
- E. an abstract class or a class

Algorithmic Analysis

Consider the distributed class ArrayAdd.

In reading add1, add2, and add3, analyze the amount of time required for a procedure to run for n additions to the array. Then consider the average time for each addition (i.e., total time divided by n)

Vote 1: consider the time for any integer n

Vote 2: to simplify computations, consider the time when n is a multiple of 10 (or when $n \% 10 = 1$)

Vote 3: to simplify computations, consider the time when n is a power of 2 (e.g., $n = 2^k$ for some integer k)

What is the order, Big-O for the average time of the method?

- A. $O(1)$
- B. $O(n)$, but not $O(1)$
- C. $O(n^2)$, but not $O(n)$
- D. Something else

The binary search algorithm for finding elements in an array has been touched upon from time to time in CS courses prior to CSC 207, but it may or may not have been discussed carefully.

To what extent do you understand how the binary search algorithm works (e.g., could describe it to someone else)?

- A. I'm clueless
- B. I have a vague idea of binary search, but not enough to describe
- C. I have a solid idea of the approach behind the binary search, but I likely would have trouble coding it.
- D. I am comfortable with both the idea of the binary search and how to code it.

Look at the code for the lookup method in the SchoolDirectory class, and analyze its efficiency.

In searching an array of n elements, one might be lucky and find an element in one step. Rather than be so optimistic, consider the average time or worst case.

- A. To find an item, lookup has $O(\log n)$ in the average case
- B. lookup has $O(n)$, but not $O(\log n)$, in the average case.
- C. lookup has $O(n^2)$, but not $O(n)$, in the average case.
- D. Cannot tell, since it matters what base one uses in computing the logarithm of n .
- E. Something else