(Provisional) Lecture 36: IRoot and Median Finding

10:00 AM, Nov 27, 2019

Contents

1 Notes 1

1 Notes

The first bit of class was dedicated to talking about testing the \texttt{idroot} function from the hw. One issue with writing the check-expects for this function was when there were multiple roots in the given interval. If you were trying to write general check-expects for this, you might run into an issue where the interval you are checking for is not the one returned by \texttt{idroot}. The solution to this issue that Spike discussed was just to plug the output of \texttt{idroot} and the output of \texttt{idroot} + 1 into the function and multiply those two values. If \texttt{idroot} gave the correct output, the product of those two values should be less than or equal to 0 because the signs are different. If the signs aren’t different, \texttt{idroot} is not functioning properly and the failed check-expect will reflect that.

The rest of class was dedicated to the discussion of median-finding: finding the median of a list. More specifically, how to do this in the most time-efficient way possible. Spike talked about how there was a group of 5 computer scientists in the 70s that worked on this problem and found the solution that he discussed in class. To be perfectly honest, it was a lot and I am not going to waste your time trying to incoherently explain it here. However, if you are really curious about it you can talk to Spike and/or the Internet :) Enjoy the rest of your break

Please let us know if you find any mistakes, inconsistencies, or confusing language in this or any other CS 17 document by filling out the anonymous feedback form: \url{http://cs.brown.edu/courses/csci0170/feedback}