Randomized Algorithms (INFO-F413)
Assignment 1: Selection

Jean Cardinal

October 2024

This homework is about randomized selection algorithms. In class, we considered the *QuickSelect* algorithm, which simply consists of picking a random pivot element, partitioning the remaining elements, and recursing at most once. We have proved that the expected number of comparisons performed by QuickSelect for selecting the kth largest element among n (assuming $k \leq n/2$) is $2n + 2k \ln\left(\frac{n-k}{k}\right) + 2n \ln\left(\frac{n}{n-k}\right)$. This expression is asymptotically equal to $2n\,(1 + h(\alpha))$, where $\alpha \sim k/n$ and $h(\alpha)$ is the entropy function in nats (using natural logarithms). This is at most

$$2(1 + \ln 2)n \simeq 3.386n.$$

In Chapter 3 of the textbook, Motwani and Raghavan describe an improved randomized algorithm for selection, called *LazySelect* that runs in time

$$2n + o(n)$$

with high probability.

Your Work

1. Read the description of the LazySelect algorithm in Section 3.3 of the Motwani-Raghavan textbook.
2. Implement both the QuickSelect and the LazySelect algorithms in your favorite programming language.
3. Devise an experimental plan for:
   (a) Verifying that the (expected) numbers of comparisons performed by the two algorithms are indeed bounded by the functions given above,

(b) comparing emprically the actual running time of the two algorithms.

4. Can you draw any conclusion as to which algorithm to use in practice?

Make sure to run your program sufficiently many times to get a valid estimation of the expected number of comparisons, and use a sufficiently large value of n for this to make sense (say, at least in the 10^4).

Requirements

1. A technical report, typeset in LaTeX, highlighting the main points of your implementations, your experimental plan, and your conclusions regarding the outcome of the experiments.
2. The source code of the programs, in an appendix.

Make sure to explicitly mention any code reuse, and give pointers to the sources. Refrain from using generative AI, see the "règlement général des études", article 40. See also this page on plagiarism.

Deadline

Thursday October 24, 2024, before midnight.