CS5800: Algorithms — Virgil Pavlu

Homework 8

Name: Anthony Wu
Collaborators:

Instructions:

- Make sure to put your name on the first page. If you are using the LaTeX template we provided, then you can make sure it appears by filling in the `yourname` command.
- Please review the grading policy outlined in the course information page.
- You must also write down with whom you worked on the assignment. If this changes from problem to problem, then you should write down this information separately with each problem.
- Problem numbers (like Exercise 3.1-1) are corresponding to CLRS 3^{rd} edition. While the 2^{nd} edition has similar problems with similar numbers, the actual exercises and their solutions are different, so make sure you are using the 3^{rd} edition.

1. 16.3-3: Consider an ordinary binary min-heap data structure supporting the instructions INSERT and EXTRACT-MIN that, when there are n items in the heap, implements each operation in O(lg n) worst-case time. Give a potential Φ function such that the amortized cost of INSERT is O(lg n) and the amortized cost of EXTRACT-MIN is O(1), and show that your potential function yields these amortized time bounds. Note that in the analysis, n is the number of items currently in the heap, and you do not know a bound on the maximum number of items that can ever be stored in the heap.

Solution:

- Let T(t_i) = number of nodes at time t_i, L(t_i) = number of levels in heap at time t_i
- We are given $c_i = \log n$ for inserts and $c_i = \log n$ for extract min
- We can design our potential function $\Phi = T(t_i) * L(t_i)$ to get an amortized cost of $\hat{c}_i$ = O(lg n) for INSERT and $\hat{c}_i$ = O(1) for EXTRACT-MIN. Note that Φ satisfies the property of $\Phi(t_0) = 0$ and $\Phi \geq 0$ when starting at an empty heap.
- If we perform an initial n inserts, we want $\sum_{i=1}^{n} \hat{c}_i \geq \sum_{i=1}^{n} c_i + \Phi(t_i) - \Phi(t_{i-1}) = \Phi(t_n) - \Phi(t_0) + \sum_{i=1}^{n} c_i = n \log n - 0 + n \log n = 2n \log n$. With this setup, we can set $\hat{c}_i = 2 \log n = O(\log n)$ to satisfy the inequality.
- With n elements in the heap, we can perform n extract-mins. We want $\sum_{i=n}^{2n} \hat{c}_i \geq \sum_{i=n}^{2n} c_i + \Phi(t_i) - \Phi(t_{i-1}) = \Phi(t_{2n}) - \Phi(t_n) + \sum_{i=1}^{n} c_i = 0 - n \log n + n \log n = 0$. With this setup, we can set $\hat{c}_i = 1 = O(1)$ to satisfy the inequality.
- Hence, we have shown the potential function $\Phi = T(t_i) * L(t_i)$ leads to an amortized cost of $\hat{c}_i$ = O(lg n) for INSERT and $\hat{c}_i$ = O(1) for EXTRACT-MIN