Basic Algorithms, Assignment 4, Solutions

Due, Thursday, Oct 4

MIDTERM October 9 In Class!
Wednesday, October 3: NO OFFICE HOURS
SPECIAL PREMIDTERM OFFICE HOURS Monday, October 9, 7-8:30p.m.

1. When H is a heap with length fifty million and `HEAPIFY-UP(A,300)` is called what is the maximum number of exchanges that can take place. What is the minimiml number of exchanges that can take place.
   Solution. Minimal is zero, when $A(300)$ is already bigger than its parent. Then maximum is if you exchange up to the root with $150, 75, 37, 18, 9, 4, 2, 1$ for 8 exchanges.

2. When H is a heap with length fifty million and `HEAPIFY-DOWN(A,300)` is called what is the maximum number of exchanges that can take place. What is the minimiml number of exchanges that can take place.
   Solution. Minimal is zero, when $A(300)$ is already bigger than its children. Then maximum is if you exchange down to the leaf with 600, 1200, 2400, ... You would do this $\lfloor \lg(50000000/300) \rfloor = 17$ times.

3. Consider a heap H with length 1023. [1] Assume the elements of the array are distinct. Let x be the third largest element in the array. What are the possible positions for x.
   Solution: It can be anywhere on the second or third levels but nowhere else so it can be in positions $2, 3, 4, 5, 6, 7$.
   Let $y = H[700]$. Can y be the largest element in the array?
   Solution: Yes, the largest element can be anywhere in the last row.
   Can y be the smallest element in the array?
   Solution: No, thats always at the top in position one.
   What is the smallest i so that it is possible that y is the i-th smallest element of the array.
   Its ancestors must be smaller than it but thats it. Node x has $\lfloor \lg x \rfloor$ ancestors (or you can just enumerate them for 700: $350, 175, \ldots$ so that $i = 9$.

[1] Did you recognize 1023 as a special number? Its one less than $1024 = 2^{10}$. The binary tree with that many nodes just fills out a row!

4. Page 69, Problem 8. (Do part (b) only for $k = 3$.)
Solution: (With two jars) Drop the first jar from $x, 2x, 3x, \ldots$ until in crashes. When it crashes between ix and $(i+1)x$ do the second jar one by one between them. So the first jar is used at most N/x times and the second one at most x times for a total of $\frac{N}{x} + x$ times. Some calculus gives that it is best to take $x = \sqrt{N}$ and then the total drops is $2\sqrt{N}$ which is certainly $o(N)$.
With three jars again drop the first jar from $x, 2x, 3x, \ldots$ until in crashes. This takes $\frac{N}{x}$ drops. Now we have x possibilities and we are in the two jar case so we can do the rest with $2\sqrt{x}$ drops. The total is $\frac{N}{x} = 2\sqrt{x}$. Some calculus gives that it is best to take $x = N^{1/3}$ and then the total drops is $3N^{1/3}$ which is $o(\sqrt{N})$ so the third jar helped. This pattern continues.

A person who never made a mistake never tried anything new.
– Albert Einstein