CS6033 Homework Assignment 5*

Due Tuesday, Oct 15th at 5:00 p.m.
Turn in this assignment as a PDF file on NYU classes
No late assignments accepted

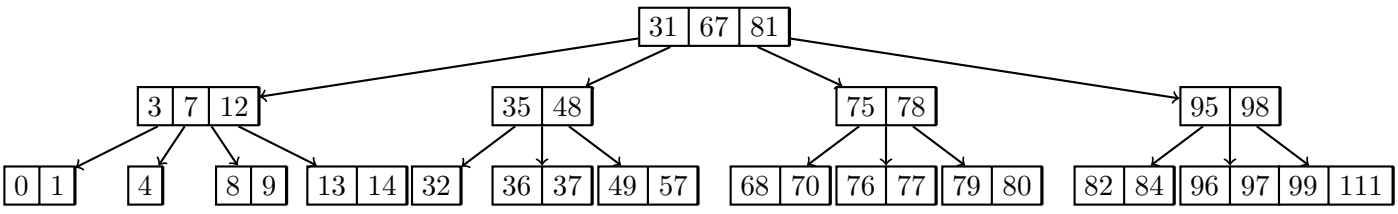
1. Prove that if you *insert* $k > 1$ nodes into a *red-black* tree you will have at least one red node. (Note: we are *only* inserting into this tree. No deletions.)

2. If you had a B-tree of degree 1000, and height 6, what is the maximum number of keys you could store in the tree? What is the minimum number of keys you could store in the tree?

3. Write the pseudo code for how to find the minimum key stored in a B-tree and how to find the predecessor of a given key stored in a B-tree. Provide the CPU running time, and the number of disk accesses using Big-Oh notation.

4. Insert the following keys: $16, 19, 25, 22, 28, 30$ in this order into the b-tree of degree 2 below using the algorithm discussed in class. Show the tree after each item is inserted.

   How many disk accesses occurred when 16 was inserted? Give an exact number (note that the root is always in main memory.)



5. (5 points) Use the substitution method to prove that $T(n) = 2T(n/2) + cn \log n$ is $O(n \log^2 n)$.

*Many of these questions came from outside sources.

6. For the following function:

   - Show the recursion tree when $x = 4^4$
   - (10 points) determine the recurrence relation for the run time using Big-Theta notation

   If dividing `X` into `Q` and `R` takes $\Theta(1)$, and `COMBINE` takes $\Theta(n \log(n))$ time.

```
SOME_FUNCTION(X, n)
     if (n = 1) return

        Divide X into Q and R
        A  = SOME_FUNCTION( Q, n/4)
        B = SOME_FUNCTION(R, n/4)
       COMBINE(A, B)
```

7. (10 points) Design an efficient algorithm to multiply a $n \times 3n$ matrix with a $3n \times n$ matrix where you use Strassen's algorithm as a subroutine. Justify your run time. No points will be given for an inefficient algorithm.

8. The blackhole `Vanishing Cortex` is experiencing record numbers of space travelers going from `NYU` to `L-Space`.[1]

   Concern is mounting that spaceships are not traveling at a safe distance from each other.

   All spaceships entering `Vanishing Cortex` must announce their arrival time.

   Design a data structure that has the following methods:

   - `near_miss()` returns the smallest time difference between two ships entering the black hole in *worst case* time $O(\log n)$ time where n is the number of ships entering the blackhole
   - `already_reserved`(t) determines if a ship is scheduled to enter the blackhole at time t in *worst case* time $O(\log n)$. t is an integer from 0 to ∞
   - `reserve`(t) makes a reservation to enter the black hole at time t in *worst case* time $O(\log n)$

[1] L-space is short for library-space.