Today’s announcements:

Code challenge! 11/19, 9p, in Siebel 0224.

Let $R$ be an equivalence relation on the set of students in this room, where $(s,t) \in R$ if $s$ and $t$ have the same favorite among \{AB, FN, DJ, ZH, FB\}. 

\[
\begin{array}{cccccccccc}
2 & 5 & 9 & 0 & 1 & 4 & 8 & 3 & 6 & 7 \\
4 & 8 & 5 & 6 & -1 & -1 & -1 & -1 & 4 & 5
\end{array}
\]
Smart unions:

Union by height:
- Keeps overall height of tree as small as possible.

Union by size:
- Increases distance to root for fewest nodes.

Both of these schemes for Union guarantee the height of the tree is __________.
Smart unions:

```cpp
int DS::Find(int i) {
    if (s[i] < 0) return i;
    else return Find(s[i]);
}

void DS::UnionBySize(int root1, int root2) {
    int newSize = s[root1]+s[root2];
    if (isBigger(root1,root2)) {
        s[root2] = root1;
        s[root1] = newSize;
    }
    else {
        s[root1] = root2;
        s[root2] = newSize;
    }
}
```
Path Compression:
Path Compression:

```cpp
int DS::Find(int i) {
    if (s[i] < 0) return i;
    else return Find(s[i]);
}

void DS::UnionBySize(int root1, int root2) {
    int newSize = s[root1] + s[root2];
    if (isBigger(root1, root2)) {
        s[root2] = root1;
        s[root1] = newSize;
    }
    else {
        s[root1] = root2;
        s[root2] = newSize;
    }
}
```
Analysis:

\[ \log^* n := \begin{cases} 
0 & \text{if } n \leq 1; \\
1 + \log^*(\log n) & \text{if } n > 1
\end{cases} \]

Example:

\[ 2^{65536} \]

Relevant result:

In an upTree implementation of Disjoint Sets using smart union and path compression upon find...

any sequence of \( m \) union and find operations results in worst case running time of \( O(\_\_\_\_\_\_\_) \), where \( n \) is the number of items.

http://research.cs.vt.edu/AVresearch/UF/
This graph can be used to quickly calculate whether a given number is divisible by 7.

1. Start at the circle node at the top.
2. For each digit d in the given number, follow d blue (solid) edges in succession. As you move from one digit to the next, follow 1 red (dashed) edge.
3. If you end up back at the circle node, your number is divisible by 7.

3703