1. Review Assignment 3.

2. Construct a treap with alphabetic key and numeric min-heap order. You are to insert the items one at a time and show the treap after each rotation. Insert letters in this order: A, B, C, D, E, F. The numeric heap keys (the random numbers) are given in the following table.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>23</td>
</tr>
<tr>
<td>B</td>
<td>12</td>
</tr>
<tr>
<td>C</td>
<td>11</td>
</tr>
<tr>
<td>D</td>
<td>7</td>
</tr>
<tr>
<td>E</td>
<td>4</td>
</tr>
<tr>
<td>F</td>
<td>1</td>
</tr>
</tbody>
</table>

3. Find the asymptotic complexity, in terms of $n$, for each of these fragments, expressing the answers using $O$, $\Theta$, or $\Omega$, whichever is most appropriate.

(a) for($i = 0; i < n; i = i+1$);
    cout << "Hi!" << endl;
(b) for($i = 1; i < n; i = 2*i$);
    cout << "Hi!" << endl;
(c) for($i = 2; i < n; i = i*i$);
    cout << "Hi!" << endl;
(d) The following code models the first phase of heapsort.
    for(int $i = n; i > 0; i--$)
    for(int $j = i; 2*j <= n; j = 2*j$)
        cout << "swap" << endl;
(e) The following code models the second phase of heapsort.
    for(int $i = n; i > 0; i--$)
    {
        cout << "swap" << endl;
        for(int $j = 1; 2*j <= i; j = 2*j$)
            cout << "swap" << endl;
    }
(f) The following code models insertion of $n$ items into an AVL tree.
    for(int $i = 1; i < n; i++$)
    for(int $j = n; j > 0; j = j/2$)
        cout << "check AVL property and possibly rotate" << endl;
4. Solve each of the following recurrences, expressing the answers using $O$, $\Theta$, or $\Omega$, whichever is most appropriate.

(a) $F(n) = F(n/2) + 1$
(b) $F(n) = F(n - 1) + O(\log n)$
(c) $F(n) = F\left(\frac{n}{2}\right) + 2F\left(\frac{n}{4}\right) + n$

For this problem, as well as the next one, see https://en.wikipedia.org/wiki/Akra%E2%80%93Bazzi_method#Example

That article is difficult to understand, so I will give a simplified version in class on Monday.

(d) $F(n) = F\left(\frac{3n}{5}\right) + F\left(\frac{4n}{5}\right) + n^2$

Use the same method you used for the previous problem. Hint: $3^2 + 4^2 = 5^2$.

(e) $F(n) = F(n - 2) + n$

5. Use Huffman’s algorithm to construct an optimal prefix code for the alphabet \{A, B, C, D, E, F\} where the frequencies of the symbols are given by the following table.

<table>
<thead>
<tr>
<th></th>
<th>(A)</th>
<th>(B)</th>
<th>(C)</th>
<th>(D)</th>
<th>(E)</th>
<th>(F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\text{Freq.})</td>
<td>7</td>
<td>3</td>
<td>4</td>
<td>8</td>
<td>12</td>
<td>3</td>
</tr>
</tbody>
</table>

6. $G$ is an acyclic directed graph with vertices \{s, a, b, c, d\}. $G$ has exactly five topological orderings, namely

- $s, a, b, c, d$
- $s, a, b, d, c$
- $s, a, d, b, c$
- $s, b, a, c, d$
- $s, b, a, d, c$

Draw $G$.

7. Find a minimum spanning tree of the weighted graph shown below.
Use union/find, with path compression.

8. Insert the letters $A, B, C, D, E, F$ into an AVL tree in that order. Show the rotations (if any) after each insertion.


10. Walk through heapsort for the following array: $A\ Q\ R\ B\ X\ S\ M\ L\ N\ T$