BASIC ALGORITHMS MIDTERM

No books or notes. Calculators OK.
Total points: 115. You must leave out 15 (or more) problem parts and specify these parts.

I was thrown out of college for cheating on the metaphysics exam.
I looked into the soul of the boy next to me.
– Woody Allen

1. (30) Consider the Gale-Shapley algorithm with \(n\) men and \(n\) women.
   (a) (15) Describe the algorithm (you do not need to show correctness.) Be sure to include in your description a description of what data structures you used.
   (b) (15) Give a detailed analysis for how much time the algorithm takes. Consider only upper bounds and write your answer as \(O(f(n))\) for some standard function \(f(n)\). (Very little credit for only the answer, the analysis is key here.)

2. (25) A computer does a billion operations a second. An algorithm GOODSORT sorts an array of size \(n\) with \(n \log n\) operations. An algorithm BADSORT sorts an array of size \(n\) with \(n^2\) operations. An algorithm AWFULSORT sorts an array of size \(n\) with \(2^n\) operations.
   (a) (5) What is, roughly, the largest \(n\) for which GOODSORT will sort an array of size \(n\) in a day.
   (b) (5) What is, roughly, the largest \(n\) for which BADSORT will sort an array of size \(n\) in a day.
   (c) (5) What is, roughly, the largest \(n\) for which AWFULSORT will sort an array of size \(n\) in a day.
   (d) (5) How long will it take GOODSORT to sort a billion items.
   (e) (5) How long will it take BADSORT to sort a billion items.
3. (15) Let $G$ be the complete graph with vertex set $\{1, \ldots, n\}$. This means that every pair of vertices are adjacent, so that $\text{adj}[i]$ consists of all vertices $v \neq i$. Assume, for convenience, that the adjacency lists are in increasing order.

Consider the breadth first search algorithm $\text{BFS}[G, 1]$. Describe how the algorithm will work. What will be the Breadth First Tree $T$ at the conclusion of the algorithm. A nice picture would help!

4. (15) No proofs needed below

(a) (5) Describe a natural problem (best would be something described in class) that takes time $\Theta(n^2)$.

(b) (5) Describe a natural problem (best would be something described in class) that takes time $\Theta(n \lg n)$.

(c) (5) Describe a natural problem (best would be something described in class) for which there is no algorithm (currently!) known taking polynomial time

5. (30) Consider the Heap $H$ with $\text{length}(H)=10$ given by (** Mea culpa! This is not actually a heap so in class students were asked to do parts (a,b) even though the end result was garbage. Parts (c,d) were unaffected. **) 

<table>
<thead>
<tr>
<th>$i$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>$H[i]$</td>
<td>5</td>
<td>2</td>
<td>7</td>
<td>9</td>
<td>1</td>
<td>3</td>
<td>8</td>
<td>0</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

(a) (5) Draw a nice picture of the heap as a binary tree.

(b) (15) Illustrate the steps of $\text{EXTRACTMIN}[H]$, showing (as nice pictures) the intermediate steps and the final state of $H$ and with brief, but cogent explanations of each step.

(c) (5) When a heap has length $n$ how long does $\text{HEAPIFY-UP}[H, 1]$ take? (Warning, trick question!)

(d) (5) When a heap has length $n$ how long does $\text{HEAPIFY-DOWN}[H, 1]$ take? (Brief reason, please.)

‘A knot!’, said Alice, already to make herself useful, and looking anxiously about her. ‘Oh, do let me help to undo it!’
– from Alice in Wonderland, by Lewis Caroll