Final Exam
(due by midnight on Wednesday, March 22)

This is the usual open-everything, but no outside help take-home test. Check "Class News", where I will post “frequently asked questions” about the test. Make sure that your answers are neat (preferably one problem per page) and legible – no first drafts, please. Also, please include this cover page with your submission.

Loop Invariant

1. A candidate has majority of a vote if more than half voters support her. Describe the result of the following algorithm. Prove your answer by a loop invariant argument.

procedure find(vote: array[1..n] of name);
  name cand; int index, count;
  begin count:=0;
    for index:=1 to n do
      if count=0 then begin cand:= vote[index]; count:=1 end
      else if cand=vote[index] then count:=count+1
      else count:=count-1
  end
2. A binomial tree of order $k$, $B_k$ is obtained from two copies of $B_{k-1}$, where one is made a principal subtree of the other ($B_0$ being the trivial tree of one node.) Nodes of such a tree can be represented in an array $A[1..2^k]$ so that the two definitional binomial trees of order $k - 1$ are represented in $A[1..2^{k-1}]$ and $A[2^{k-1} + 1..2^k]$ (with the root in $A[1]$).

Design and prove correct a linear-time algorithm that heapifies a binomial tree of order $k$, $B_k$ with node values stored in an array $A[1..n]$ (for $n = 2^k$), as above.

Amortized Complexity

3. Given an array of $n$ binary digits $A[0..n-1]$ representing an integer $a = \sum_{0 \leq i < n} A[i]2^i$. Analyze (by the credit invariant method) the complexity (number of bit changes) of performing $m$ Decrement operations, each changing the contents of $A$ so that $a$ is decremented (modulo $2^n$) by 1. Assume that initially, $A$ contains $\ell$ ones ($\{|i : A[i] = 1\} = \ell$). Consider (i) $m \in o(n)$; (ii) $m \in \Theta(n)$; (iii) $m \in \Omega(n)$.

Dynamic Programming

4. The following table defines a binary operation $\circ$ on the set $\Sigma = \{a, b, c\}$.

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>a</td>
<td>b</td>
<td>a</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>c</td>
<td>c</td>
</tr>
<tr>
<td>c</td>
<td>b</td>
<td>a</td>
<td>c</td>
</tr>
</tbody>
</table>

Thus, $a \circ a = a$, $a \circ b = b$, $a \circ c = a$, etc. Describe an algorithm which, upon input of a string $w = w_1w_2\cdots w_n$ of symbols of $\Sigma$, determines whether or not it is possible to parenthesize $w$ so that the product, according to the above operation, is $a$.

$\mathcal{P}$ vs. $\mathcal{NP}$

6. Prove that the relation $\propto_p$ is transitive and reflexive. Is it symmetric?

7. Assume that there is a polynomial time algorithm CLQ to solve the MaximumClique decision problem (Instance: graph $G$ and integer $K$; Question: Does $G$ have a completely connected set of $K$ vertices?).

(i) Show how to use CLQ to determine the maximum clique size of a given graph in polynomial time.

(ii) Show how to use CLQ to find a maximum clique of a given graph in polynomial time.