CIS101 Handout #16
David Helmbold
November 12, 1998

# Written Assignment 4

20 pts, due at the start of lecture Thursday, November 19

1. (15 pts) Induction. First, read the induction handout posted on the class web page. Then use the following notation to prove that every binary tree $T$ has the property that $N(T) = E(T)+1$.

   Here we use $T$ for a binary tree, $N(T)$ is a function taking a binary tree and returning the number of nodes in the tree, $E(T)$ is a function taking a binary tree and returning the number of edges in the tree, and $T_R$ and $T_L$ are the subtrees of $T$.

   Recall that every (non-empty) binary tree is either:

   1. a single node (the root), or
   2. a single node (the root) connected by an edge to the root of a single (left or right) non-empty binary subtree, or
   3. a single node (the root) connected by edges to the roots of both a non-empty left subtree and a non-empty right subtree, and the two subtrees are disjoint;

   and use induction on the number of nodes.

   a. (1 pts) What is the size measure on binary trees?

   b. (2 pts) Formally define IH($k$).

   c. (2 pts) Show that IH(1) is true.

   d. (2 pts) In the inductive step you must show that IH($j$) is true for an arbitrary $j > 1$. What should you assume at the start of the inductive step?

   e. (4 pts) Complete the inductive step by examining an arbitrary $k$-node binary tree $T$ and doing a case analysis based on the definition of binary tree. Clearly indicate where you are using the inductive assumptions.

   f. (2 pts) Is it right or wrong to write $N(T+1)$? Why?

   g. (2 pts) Is it right or wrong to write IH($N(T_R)$)? Why?

2. (2 pts) Introduction: If Breadth-First search is started at any vertex $a$ in any (undirected) graph $G$ it will find a shortest path tree rooted at $a$. By changing the orders of the adjacency lists, Breadth-First search can often find a different shortest path tree from source $a$.

   Explain why BFS will never find the following shortest path tree rooted at $a$

```
      a
     / \
    /   \
   b     c
   |     |
   |     |
   |     |
   d     e
```

   when run on this graph:

```
      a
     / \
    /   \
   b     c
   |\   /|
   | \ / |
   |  X  |
   | / \ |
   |/   \|
   d     e
```

3. (3 pts) Exercise 23.3-8 on page 494. (Give an example graph and DFS where a vertex with both incoming and outgoing edges winds up as the only node in its DFS tree).

Recommended problems, not to be turned in

1. Exercise 23.1-5 on page 468 of the text (Graph squaring).

2. Exercises 23.2-2 on page 476 of the text (BFS example). Assume all adjacency lists are in alphabetical order. Show the order that the vertices are enqueued, the distances from $u$ to each vertex, and the resulting BFS tree.

3. Exercises 23.3-2 on page 484 of the text (DFS example)