MATH/CSCI 2112, DISCRETE STRUCTURES I, FALL 2005

Handout 5: Problems for Induction

Problem 1. (a) Prove by induction that $n^3 \leqslant 3^n$, for all n. (b) Why do you need more than one base case for the proof in part (a)? For which n does the induction step work?

Problem 2. Consider the sequence of numbers defined by $X_0 = 0$ and $X_{n+1} = 3X_n + 1$. (a) Write down the first 6 members of this sequence. (b) Prove that $X_n \leqslant 3^n$, for all n. Hint: it might be necessary to strengthen the induction hypothesis.

Problem 3. Consider the following Java method (or C procedure):

```
int foo(int n)
{
  if (n == 0) {
    return 1;
  } else if (n == 1) {
    return 3;
  } else {
    return 3 * foo(n-2) + 2 * foo(n-1);
  }
}
```

Prove that for all natural numbers $n \geqslant 0$, $\text{foo}(n) = 3^n$. Hint: it might be useful to first translate this program into mathematical notation, where $X_n = \text{foo}(n)$.

Problem 4. Consider the sequence of numbers defined by

$$\begin{array}{lcll} X_0 & = & 0, & \\ X_n & = & X_{n-1} + 2, & \text{when } n \text{ is odd,} \\ X_n & = & 2X_{n-1}, & \text{when } n \text{ is even and } n \neq 0. \end{array}$$

(a) Write down the first 10 members of this sequence. (b) Prove that $X_n \leqslant 2^n$, for all $n \geqslant 0$.

Problem 5. Consider two sequences of numbers defined by mutual recursion:

$$\begin{array}{lcl} A_0 & = & 0, \\ B_0 & = & 1, \\ A_n & = & A_{n-1} + 2B_{n-1}, \text{ for } n \geqslant 1, \\ B_n & = & 2A_{n-1} + B_{n-1}, \text{ for } n \geqslant 1. \end{array}$$

(a) Prove that $A_n - B_n = -1$ for all even n, and $A_n - B_n = 1$ for all odd n. (Hint: Use just a single induction). (b) Prove that $A_n + B_n = 3^n$, for all n. (c) Can you find closed formulas (= non-recursive formulas) for A_n and B_n?