ECE 130 - INDUCTION AND RECURSION - INVESTIGATION 10
RECURSIVELY DEFINED SEQUENCES

WINTER 2004 **A.P. FELZER**

To do "well" on this investigation you must not only get the right answers but must also do neat, complete and concise writeups that make obvious what each problem is, how you're solving the problem and what your answer is. You also need to include appropriate graphs and tables.

In the last two Investigations we showed how to write and solve recursive equations. The main objective of this Investigation is to show how Languages made from strings of letters and symbols can be defined recursively.

1. We begin with some review problems. Write pseudocode for determining how long it takes a Fibonacci family of rabbits to reach 1000

2. Go backwards to solve the following recursive equation for S(4)

$$S(n) = S(n-1) + 3n \qquad S(0) = 0$$

3. The objective of this problem is to introduce *Gambler's Ruin*. In this scenario

   (1) We bet 1$ at a time by flipping a coin. We win $1 if the coin comes up heads and lose our $1 if the coin comes up tails

   (2) p is the probability the coin comes up heads (the fraction of time it comes up heads) and q = 1–p is the probability it comes up tails (the fraction of time it comes up tails)

   (3) The betting stops if we lose all our money or we reach some predetermined goal of N dollars

   (4) P(n) is the probability we eventually lose all our money if we presently have n dollars

   a. Explain in words why $P(n+1) = pP(n+2) + qP(n)$

   b. When $p \neq q$ it can be shown that the solution to the recursive equation in part (a) is given by

   $$P(k) = \frac{r^k - r^N}{1 - r^N}$$

   with $r = q/p$. Make use of this result to find P(20) if we start out with n = $20, p = 0.48, q = 0.52 and N = $40

   c. Make use of your result in part (b) to decide if the gambler is better off betting $1 at a time or betting all $20 at once

4. The objective of this and the next problem is to show how recursion can be used to define "Languages" consisting of strings of letters and symbols like the following

$$x = abbaba$$

   For this problem the Language *S* has *words x* equal to strings of *a's* and *b's* as follows

   (1) S1: The empty string ε is in Language S

   (2) S2: If x is in S then so is xa

   (3) S3: If x is in S then so is xab

a. Determine if x = abaaab is in the Language S. Hint - start from the right hand side. Be sure to justify each step with an S1, S2 or S3
b. Determine if x = aababa is in Language S
c. Determine if x = abbaa is in Language S

5. This objective of this problem is to make use of recursion to define the Language P of parenthesis - the language for specifying valid strings of parenthesis like (())()

   (1) P1: The empty string ε is in Language P

   (2) P2: If x is in P then so is (x)

   (3) P3: If x and y are in P then so is xy

   a. Is it possible to determine if a given string of parenthesis is valid by simply determining if there are an equal number of left and right parenthesis. Either prove or give a counterexample
   b. Determine if x = (( )( )) is in P. Hint - as we go from left to right the number of left parenthesis must always be greater than or equal to the number of right parenthesis. Be sure to justify each step with a P1, P2 or P3
   c. Determine if x = (( )))(is in P

6. Make up your own Language and then find some valid and invalid strings