Recursion and the Determinant of a Matrix

Tommy Luckner
Department of Mathematics

Overview

Our goal this week is to explore some examples of recursive function definitions and to write a recursive function to compute the determinant of a square matrix.

Activities

The process of solving a problem by reducing it to smaller instances of itself is called *recursion.*

- Compute $1 + 2 + 3 + \cdots + n$:

```
function s = mysum(n)

if n == 1
s = 1;
    return;
else
s = n+ mysum(n-1);
end
end
```

- Compute the nth Fibonacci number: $F_{n+2} = F_{n+1} + F_n$.

```
function F = fibonacci(n)

if n==0  || n==1
F=1;
return;
else
F=fibonacci(n-1)+fibonacci(n-2) ;
end
end
```

- Tower of Hanoi:
  You are given three columns, the first of which contains n rings, stacked with decreasing size. The object of this game is to move all the rings from the first column to the third column. The rings should be moved one by one, and a ring cannot be placed on top of a smaller ring. Play it online:

http://www.novelgames.com/en/tower/

Write a function whose input is the number of rings, the starting rod, the auxiliary rod, and the ending rod. Its output is the printed directions that will tell you the minimal number of steps to win the game. (An optimal strategy exists!) Test your code by implementing the outputted steps for some n in the game. It will tell you if you used the minimal number of steps.

Game strategy: The trick to move a pile of n pieces is to move the first $n-1$ pieces from the starting rod to the auxiliary rod, then move the last piece to the ending rod, and then move the first $n-1$ pieces to the ending rod.

Algorithm:

1. Print the step for if there is 1 ring.
2. Move all the rings, except the last one, from column 1 to column 2.
3. Move the last ring from column 1 to column 3.
4. Move all the rings, except the last one, from column 2 to column 3.