

Massachusetts Institute of Technology
Department of Mechanical Engineering

2.003J/1.053J Dynamics & Control I

Fall 2007

Homework 5

Issued: Oct. 12, 2007

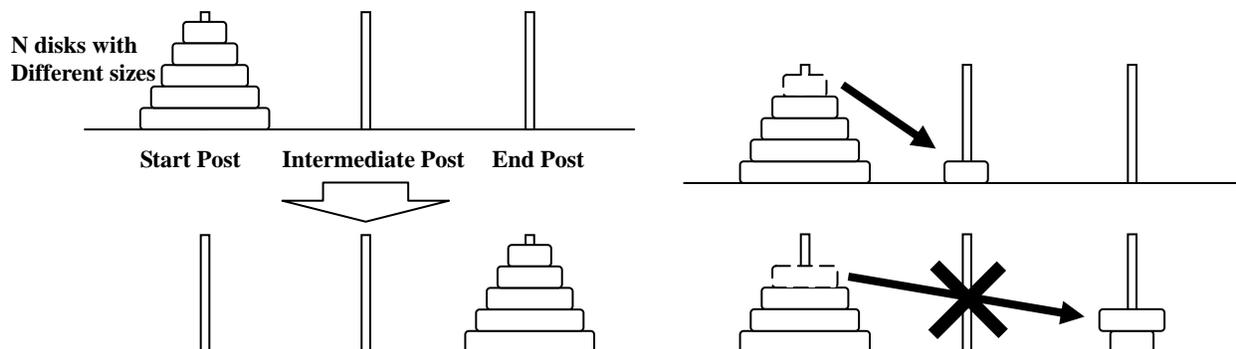
Due: Oct. 19, 2007

Problem 5.1 : Calculating the factorial of non-negative integer with recursion

Write a function to calculate the factorial of non-negative integer, $n!$ as in the last homework. However, for this time, you should use a recursive algorithm. Function name (and m-file name) should be `'fctrlrc_your_kerberos_name'` and upload it to 2.003 MIT Server site. You also submit hardcopy of your code. Calculate $125!$ with this function.

Problem 5.2 : Developing the Algorithm for the Solution of the Tower of Hanoi

The tower of Hanoi (commonly also known as the "towers of Hanoi"), is a mathematical game or puzzle invented by E. Lucas in 1883. It is also very famous example for using recursion in programming. It consists of three posts, and N different size disks. Initially, N disks are stacked on the leftmost side post (starting post or post 1) in the order of the largest one on the bottom to the smallest one on the top. Your task is to move all the disks on the rightmost side post (end post or post 3) with same order of disks. You can use the middle post (intermediate post or post 2) for temporary stacking of disks.



During the movement of disks, you must further follow these rules:

1. Only one disk is allowed to be moved at a time.

2. In each move, only the disk on the top of any post can move and be stacked on another post.
 3. The bigger disk must always be stacked below the smaller ones.
- i) Obtain the solution for the tower of Hanoi with $N=2$ different size disks. Don't try to use MATLAB at this time. Just write down the order of disk movements as below:

'Trial number' : from **'post number'** to **'post number'**
 - ii) Express your algorithm for the solution of the tower of Hanoi with N different size disks. You can assume that you have the algorithm for solving the $N-1$ disk problem. You can then develop an algorithm for the N -disk problem using the $N-1$ disk solution based on recursive.

Problem 5.3 : Solving the Tower of Hanoi with recursive algorithm

Now, generate the function or m-file for the solution of “tower of Hanoi” with 5 different size disks. Use recursive approach. Your function has input argument of 'n', number of disk on the starting post, and output argument of 'o', matrix for disk movements (each row represents each movement from the post indicated in the first column to the post shown in the second column. For example, [1 3] means movement from post 1 to post 3) Function name (and m-file name) should be 'hanoi_your_kerberos_name' and upload it to 2.003 MIT Server site. You submit hardcopy of your function code as well as the solution of the tower of Hanoi with 5 different size disks. Execution of m-file should provide the following result in case of $N=1$:

```
>> hanoi(1)
ans =
     1     3
```

Problem 5.4 : (Optional) Comparison between solving problem with and without recursion.

In problem 4.3 and problem 5.1., you are asked to make the function of calculating $n!$ with different algorithms. These functions basically work same in terms of mathematics, but they don't in terms of computer performance. Describe both advantage and disadvantage of each approach briefly.