

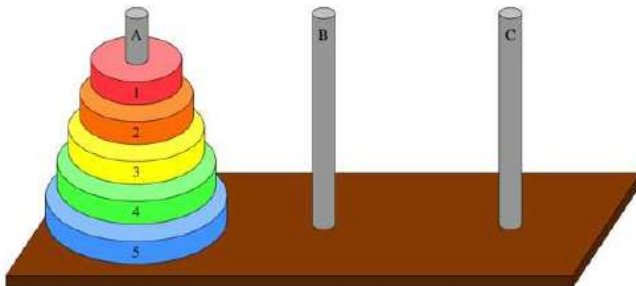
Recursion Question

One of the classic programming problems that is often solved by recursion is the towers of Hanoi problem. A good explanation and walkthrough are provided by Cormen & Balkcom (n.d.) - the link is in the reading list. (the code they used for their visual example is provided on their website as well).

- Read the explanation, study the code and then create your own version using Python (if you want to make it more interesting you can use asterisks to represent the disks). Create a version that asks for the number of disks and then executes the moves, and then finally displays the number of moves executed.
- What is the (theoretical) maximum number of disks that your program can move without generating an error?
- What limits the number of iterations? What is the implication for application and system security?

Towers of Hanoi Answer

- Given the below scenario where number of disks is $n=5$:



- The goal is to move all disks to B by:
 - 1.) Only moving one disk at a time
 - 2.) Not having a disk ever rest on another smaller in size
- The theoretical maximum number of disks is linked to the memory available in the computer running the application, as recursion causes the stack to increase in size.
- Therefore, the limit is based on the amount of memory available in the computer.
- Security-wise one could either crash a computer, or override other data in memory causing unpredictable problems.