Hanoi Algorithm Design, Complexity, & Implementation Analysis
by Carlos Flores

Implement an algorithm and program it to solve a Tower of Hanoi puzzle for the following graph $\mathcal{G}$ =($\mathcal{V}$,$\mathcal{E}$) with the Vertices, $\mathcal{V}$ = {Start, Aux 1, Aux 2, Aux 3, Aux 4, and Dest}, and Directed Edges $\mathcal{E}$ = {(Start, Aux 1), (Aux 1, Aux 2), (Aux 2, Aux 3), (Aux 3, Aux 4), (Aux 4, Aux 1), and (Aux 1, Dest)}.

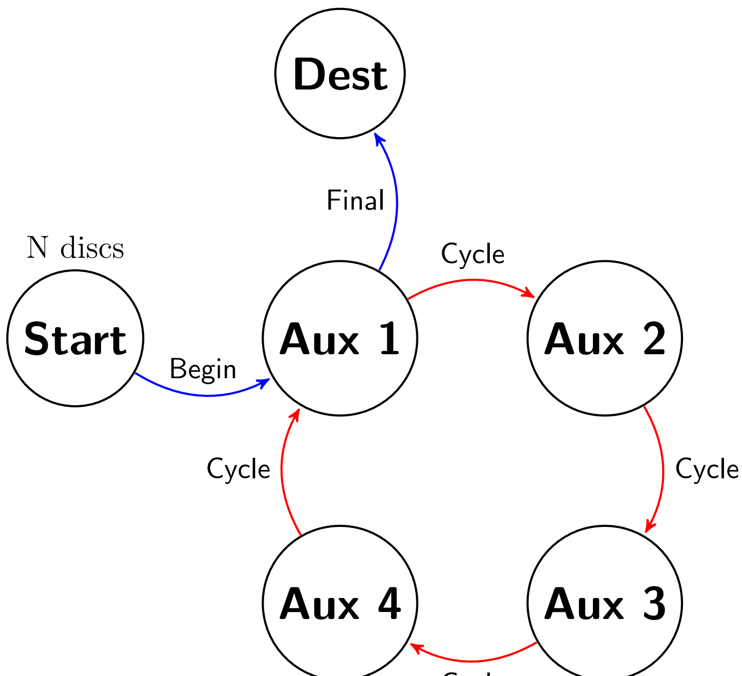

Figure 1.a Tower of Hanoi puzzle for $\mathcal{G}$ =($\mathcal{V}$,$\mathcal{E}$).

Design Rationale

Our goal is to move a stack of N number of discs from Start to Dest. In order to move N number of discs from Start to Dest we must move (N-1) discs to access the last disc. This can be achieved by the following recursive pattern:

Move N-1 discs from Start to Aux 3 (this frees up the largest disc on Start). Move the largest disc from Start to Dest Move N-1 discs from Aux 3 to Dest (this remakes the original pile on Dest).