DYNAMIC PROGRAMMING

By Alyssa Dalton

Outline

- What is Dynamic Programming?
- Algorithm Steps
- Tower of Hanoi Example

What is Dynamic Programming?

- Similar to divide-and-conquer algorithm
- Differs in the fact that sub problems are not independent
- Solves each subproblem and saves answer to avoid repetition
- Typically applied to optimization problems

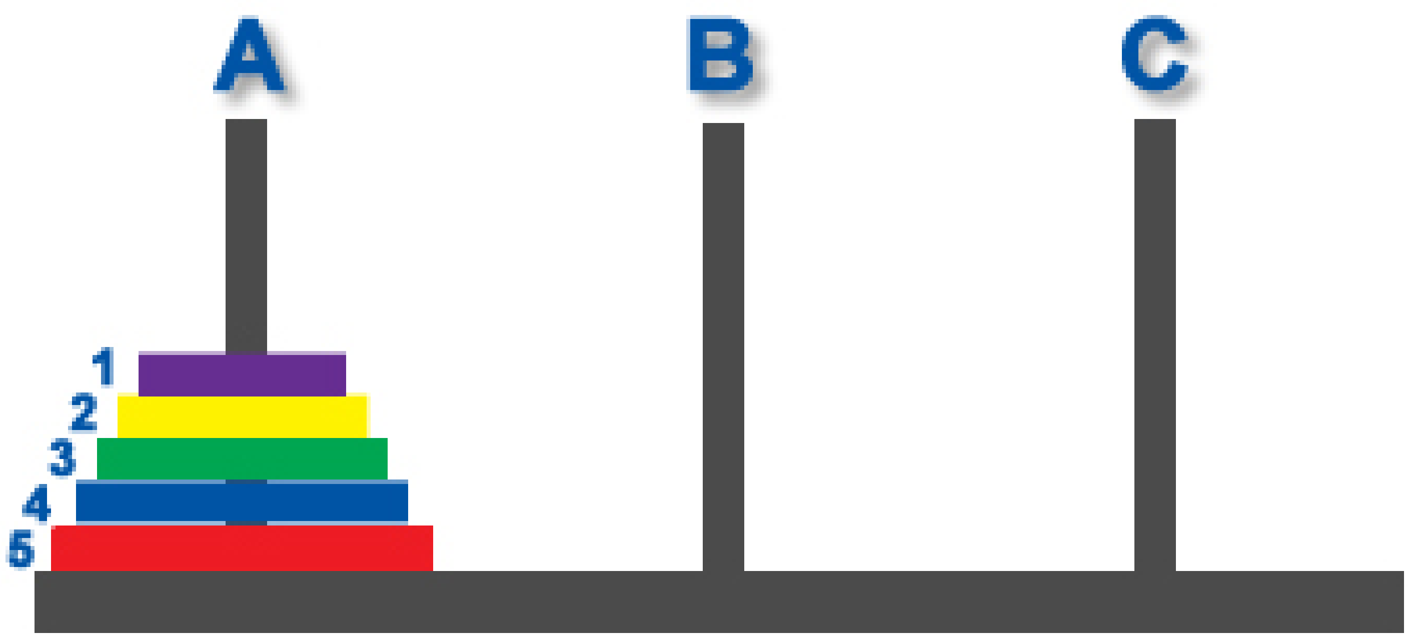
(Introduction to Algorithms)

Algorithm Steps

1. Characterize the structure of an optimal solution.
2. Recursively define the value of an optimal solution.
3. Compute the value of an optimal solution in a bottom-up fashion.
4. Construct an optimal solution from computed information.

(Introduction to Algorithms)

Tower of Hanoi



http://haubergs.com/hanoi

Tower of Hanoi

- The objective of the puzzle is to move the entire stack to another rod, obeying the following rules:

1. Only one disk may be moved at a time.
2. Each move consists of taking the upper disk from one of the rods and sliding it onto another rod, on top of the other disks that may already be present on that rod.
3. No disk may be placed on top of a smaller disk.

(Wikipedia)

Tower of Hanoi – Step 1

- Characterize the structure of an optimal solution:
  - Move all disks (3) from one peg to another, minimizing the number of moves.

Tower of Hanoi – Step 2

- Recursively define the value of an optimal solution:
  - move $n-1$ discs from A to B. This leaves disc n alone on peg A
  - move disc n from A to C
  - move $n-1$ discs from B to C so they sit on disc n
- Each step is considered a subproblem

Tower of Hanoi – Step 3

- Compute the value of an optimal solution in a bottom-up fashion:
  - Compute for n = 1, 2, etc.
  - Save steps in table.

Number of Discs	Disc moved	Moves (subproblem)	Moves (problem)
1	1	1 to E	n to C
2	1	1 to I	n-1 to B
	2	2 to E	n to C
	1	1 to E	n-1 to C

S = start
I = intermediate
E = end

Tower of Hanoi – Step 4

- Construct an optimal solution from computed information:
  - move $n-1$ discs from A to B. This leaves disc n alone on peg A

Number of Discs	Disc moved	Moves (subproblem)	Moves (problem)
2	1	1 to I	
	2	2 to E	
	1	1 to E	
3	1	1 to I	1 to C
	2	2 to E	2 to B
	1	1 to E	1 to B

Tower of Hanoi – Step 4 cont.

- move disc *n* from A to C

Number of Discs	Disc moved	Moves (subproblem)	Moves (problem)
1	1	1 to E	
3	1	1 to I	n-2 to C
	2	2 to E	n-1 to B
	1	1 to E	n-2 to B
	3	3 to E	n to C

Tower of Hanoi – Step 4 cont.

- move *n*−1 discs from B to C so they sit on disc *n*

Number of Discs	Disc moved	Moves (subproblem)	Moves (problem)
2	1	1 to I	
	2	2 to E	
	1	1 to E	

3	1	1 to I	n-2 to C
	2	2 to E	n-1 to B
	1	1 to E	n-2 to B
	3	3 to E	n to C
	1	1 to I	n-2 to A
	2	2 to E	n-1 to C
	1	1 to E	n-2 to C

Outline

- What is Dynamic Programming?
- Algorithm Steps
- Tower of Hanoi Example

References

- http://polaris.umuc.edu/~flazarus/8j_def/8J_cont15b.html
- http://mathworld.wolfram.com/TowerofHanoi.html
- http://haubergs.com/hanoi
- http://en.wikipedia.org/wiki/Dynamic_programming
- Introduction to Algorithms, Second Edition