

<http://www.cs.cornell.edu/courses/cs1110/2018sp>

Lecture 15: Recursion

(Sections 5.8-5.10)

CS 1110

Introduction to Computing Using Python



Cornell CIS
COMPUTING AND INFORMATION SCIENCE

[E. Andersen, A. Bracy, D. Gries, L. Lee, S. Marschner, C. Van Loan, W. White]

3

Recursion

- **Recursive Function:**
A function that calls itself (directly or indirectly)

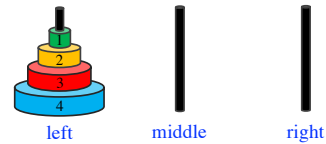
Examples

- **Blast-off**
- **Gift in gift**
- **Family Trees**
- Towers of Hanoi
- Deblanking

4

Tower of Hanoi

- Three towers: *left*, *middle*, and *right*
- n disks of unique sizes on *left*
- **Goal:** move all disks from *left* to *right*
- Cannot put a larger disk on top of a smaller disk



5

Divide and Conquer

Goal: Solve really big problem P

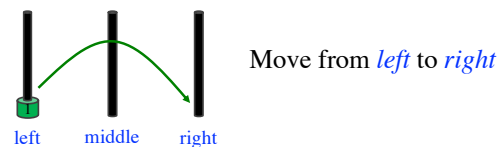
Idea: Split into smaller problems, solve, combine

3 Steps:

1. Decide what to do for simple cases
2. Decide how to break up the task
3. Decide how to combine your work

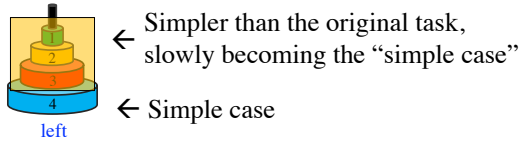
18

Decide what to do for simple cases



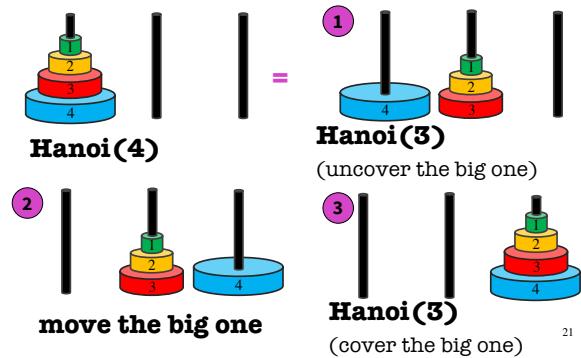
19

Decide how to break up the task



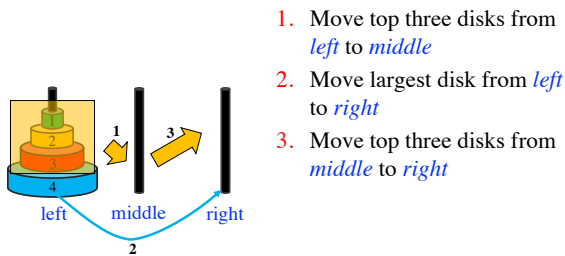
20

Decide how to combine your work



21

4 Discs: High-level Idea



1. Move top three disks from *left* to *middle*
2. Move largest disk from *left* to *right*
3. Move top three disks from *middle* to *right*

22

Recursion vs Iteration

- **Recursion** is *provably equivalent* to **iteration**
 - Iteration includes **for-loop** and **while-loop** (later)
 - Anything can do in one, can do in the other
- But some things are easier with recursion
 - And some things are easier with iteration
- Will **not** teach you when to choose recursion
- We just want you to *understand the technique*

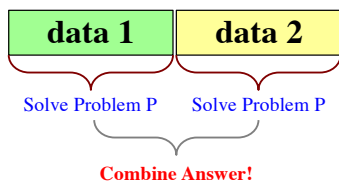
24

Divide and Conquer

Goal: Solve problem P on a piece of data



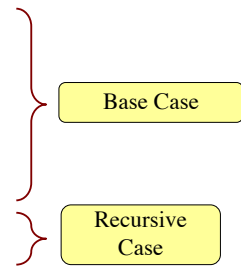
Idea: Split data into two parts and solve problem



28

Putting it All Together

```
def deblank(s):
    """Returns: s w/o blanks"""
    if s == "":
        return s
    elif len(s) == 1:
        if s[0] == ' ':
            return ""
        else:
            return s
    left = deblank(s[0])
    right = deblank(s[1:])
    return left+right
```



32