Divide and Conquer

example problem: sorting an array

$A = \underbrace{[A|L|G|O|R}_{L}\underbrace{|I|T|H|M|S]}_{R}$

$S_L = [\cancel{A}|\cancel{G}|\cancel{L}|\cancel{O}|\cancel{R}]$ ↑ ↑ ↑ ↑ ↑

$S_R = [\cancel{H}|\cancel{I}|\cancel{M}|S|T]$ ↑ ↑ ↑ ↑

to_return = [A G H I L M R S T]

w/ table:
- 4 parts of pf by induction
- corresponding parts of rec alg
- explanations/examples for each part

```
                  input
mergeSort(A):
if len(A) > 1:   recursive case
    S_L = mergesort(L)
    S_R = mergesort(R)
    return merge(S_L, S_R)
else:             non-recursive
    → return A    base case
```

smaller instances of input

Proofs by Induction

0) claim

① Universal Declaration

"Let X be an arbitrary..."

② Inductive hypothesis

"Assume property holds for all Y smaller than X"

Recursive Algs

arbitrary input

"Recursion fairy"

Assume alg works for all smaller instances

3) Base case

Prove property directly

4) Inductive case

Use the claim on smaller instances to show that claim holds for X

5) conclusion

solve directly

Use the result of calling alg on smaller instances to construct a fully correct output

function always returns

w/ table:

- 4 parts of pf by induction
- corresponding parts of rec alg
- explanations/examples for each part