UMBC

AN HONORS UNIVERSITY IN MARYLAND

IS 709/809: Computational Methods for IS Research

Math Review: Algorithm Analysis

Nirmalya Roy

Department of Information Systems

University of Maryland Baltimore County

Topics

- Proof techniques
  - Proof by induction
  - Proof by counterexample
  - Proof by contradiction
- Recursion
- Summary

Proof Techniques

- What do we want to prove?
  - Properties of a data structure always hold for all operations
  - Algorithm running time/memory usage will never exceed some limit
  - Algorithm will always be correct
  - Algorithm will always terminate

Proof by Induction

- Goal: Prove some hypothesis is true
- Three-step process
  - Prove the Base case:
    - Show hypothesis is true for some initial conditions
    - This step is almost always trivial
  - **Inductive hypothesis:** Assume hypothesis is true for all values $\leq k$
  - Using the inductive hypothesis, show that the theorem is true for the next value, typically $k + 1$

Induction Example

- Prove arithmetic series $$\sum_{i=1}^{N} i = \frac{N(N+1)}{2}$$
- (Step 1) Base case: Show true for N=1

$$\sum_{i=1}^{1} i = 1 = \frac{1(1+1)}{2}$$

Induction Example (cont'd)

- (Step 2) Assume true for N=k
- (Step 3) Show true for N=k+1

$$
\begin{aligned}
\sum_{i=1}^{k+1} i &= (k+1) + \sum_{i=1}^{k} i \\
&= (k+1) + \frac{k(k+1)}{2} \\
&= \frac{2(k+1) + k(k+1)}{2} \\
&= \frac{(k+1)(k+2)}{2}
\end{aligned}
$$

More Induction Examples

- Prove the geometric series $\sum_{i=0}^{N} A^i = \dfrac{A^{N+1}-1}{A-1}$

- Prove that the number of nodes N in a complete binary tree of depth D is $2^{D+1}-1$

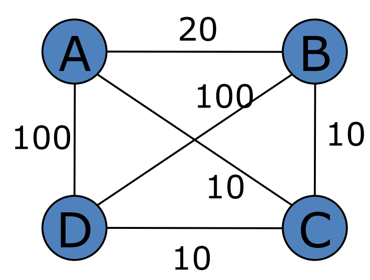
- Prove that $\sum_{i=1}^{N} i^2 = \dfrac{N(N+1)(2N+1)}{6}$

Proof by Counterexample

- Prove hypothesis is not true by giving an example that doesn't work
  - Example: $2^N > N^2$?
  - Example: Prove or disprove "all prime numbers are odd numbers"
- Proof by example?
- Proof by lots of examples?
- Proof by all possible examples?
  - Empirical proof
  - Hard when input size and contents can vary arbitrarily

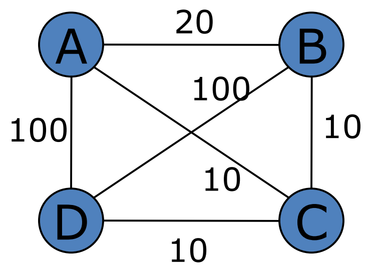
Another Example

- Traveling salesman problem
  - Given N cities and costs for traveling between each pair of cities, find the least-cost tour to visit *each* city *exactly once*
- Hypothesis
  - Given a least-cost tour for N cities, the same tour will be least-cost for (N-1) cities
  - e.g., if A→B→C→D is the least-cost tour for cities {A,B,C,D}, then A→B→C will be the least-cost tour for cities {A,B,C}



Another Example (cont'd)

- Counterexample
  - Cost (A→B→C→D) = 40 (optimal)
  - Cost (A→B→C) = 30
  - Cost (A→C→B) = 20

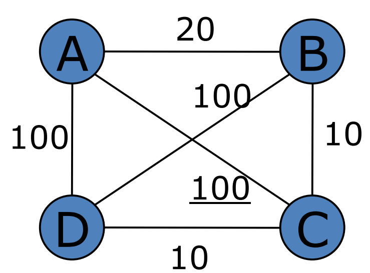


Proof by Contradiction

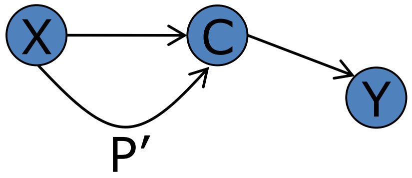
- Assume hypothesis is false
- Show this assumption leads to a contradiction (i.e., some known property is violated)
  - Can’t use special cases or specific examples
- Therefore, hypothesis must be true

Contradiction Example

- Variant of traveling salesman problem
  - Given N cities and costs for traveling between each pair of cities, find the least-cost path to go from city X to city Y
- Hypothesis
  - A least-cost path from X to Y contains least-cost paths from X to every city on the path
  - E.g., if X→C1→C2→C3→Y is the least-cost path from X to Y, then
    - X→C1→C2→C3 is the least-cost path from X to C3
    - X→C1→C2 is the least-cost path from X to C2
    - X→C1 is the least-cost path from X to C1



Contradiction Example (cont’d)



- Assume hypothesis is false
  - i.e., Given a least-cost path P from X to Y that go is a better path P’ from X to C than the one in P
- Show a contradiction
  - But we could replace the subpath from X to C in P with this lesser-cost path P’
  - The path cost from C to Y is the same
  - Thus we now have a better path from X to Y
  - But this violates the assumption that P is the least-cost path from X to Y
- Therefore, the original hypothesis must be true

More Contradiction Example

- Example: Prove that the square root of 2 is irrational (a number that cannot be expressed as a fraction a/b, where a and b are integers, $b \neq 0$)

- Proof: will be derived in the class
  - assume root of 2 is a rational number
  - assume a/b is simplified to the lowest terms
    - can be done with any fraction
    - in order for a/b to be in its simplest terms, both a and b must not be even. One or both must be odd. Otherwise, you could simplify

Recursion

- A recursive function is defined in terms of itself
- Examples of recursive functions
  - Factorial
  - Fibonacci

$$n! = \begin{cases} 1 \text{ if } \ n = 0 \\ n * (n-1)! \text{ if } \ n > 0 \end{cases}$$

```
Factorial (n)
    if n = 0
    then return 1
    else return (n * Factorial (n-1))
```

Example

- Fibonacci numbers
  - F(0) = 0
  - F(1) = 1
  - F(2) = 1
  - F(3) = 2
  - F(4) = 3
  - F(5) = 5
  - \- - - - - - - - - - -
  - F(n) = F(n-1) + F(n-2)

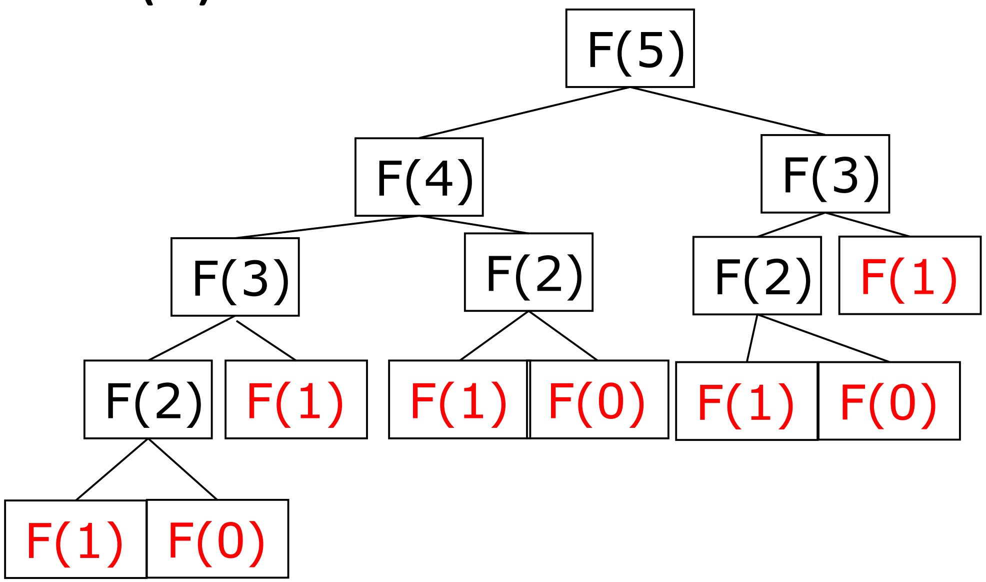
```
Fibonacci (n)
    if (n ≤ 1)
    then return 1
    else return (Fibonacci (n-1) + Fibonacci (n-2))
```

Basic Rules of Recursion

- Base cases
  - Must always have some base cases, which can be solved without recursion
- Making progress
  - Recursive calls must always make progress toward a base case
- Design rule
  - Assume that all the recursive calls work
- Compound interest rule
  - Never duplicate work by solving the same instance of a problem in separate recursive calls

Example (cont'd)

- Fibonacci (5)



- The Fibonacci numbers are the numbers in the following integer sequence:
  - 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144…………

Summary

- Proofs by mathematical induction, counterexample and contradiction
- Recursion
- Tools to help us analyze the performance of our data structures and algorithms

- Next:
  - Floors, ceilings, exponents, logarithms, series, and modular arithmetic

Questions

?