

CSE 20—Discrete Math

Winter, 2006

February 23, Day 14
Summations (portions from *Concrete Mathematics* by Graham, Knuth, Patashnik)
Induction

Instructor: Neil Rhodes

Perturbation method

Given:

$$S_n = \sum_{0 \leq k \leq n} a_k.$$

Rewrite S_{n+1} by splitting off first and last term:

$$\begin{aligned} S_n + a_{n+1} &= a_0 + \sum_{1 \leq k \leq n+1} a_k \\ &= a_0 + \sum_{1 \leq k+1 \leq n+1} a_{k+1} \\ &= a_0 + \sum_{0 \leq k \leq n} a_{k+1} \end{aligned}$$

Then, work on last sum and express in terms of S_n . Finally, solve for S_n .

2

Perturbation method example

$$S_n = \sum_{0 \leq k \leq n} x^k$$

3

Standard closed forms

Arithmetic series

$$\sum_{k=0}^n k = \frac{1}{2}n(n+1)$$

Sums of squares and cubes

$$\begin{aligned} \sum_{k=0}^n k^2 &= \frac{n(n+1)(2n+1)}{6} \\ \sum_{k=0}^n k^3 &= \frac{n^2(n+1)^2}{4} \end{aligned}$$

4

Standard closed forms

Geometric series ($x \neq 1$)

$$\sum_{k=0}^n x^k = \frac{x^{n+1} - 1}{x - 1}$$

Infinite Geometric series ($|x| < 1$)

$$\sum_{k=0}^{\infty} x^k = \frac{1}{1 - x}$$

5

Standard closed form

Harmonic series

$$H_n = \sum_{k=0}^n \frac{1}{k} \approx \ln n$$

n	0	1	2	3	4	5	6
H_n	0	1	$\frac{3}{2}$	$\frac{11}{6}$	$\frac{25}{12}$	$\frac{137}{60}$	$\frac{49}{20}$

6

Application of harmonic series

Given a stack of 52 playing cards (of length 2 units). Can you stack them overlapping so the top card completely overhangs the table?

7

Induction

Method of proving a statement $P(n)$ is true for $n \in a, a+1, \dots \infty$

- Base case (show $P(a)$)
 - Prove true for $n = a$ (often, $a = 1$ or 0)
- Inductive step
 - Show $(n > a) \wedge P(n-1) \rightarrow P(n)$
 - Assume true for $n-1$
 - Show true for n
 - Or, show $(n > a) \wedge P(0) \wedge P(1) \wedge \dots \wedge P(n-1) \rightarrow P(n)$
 - Assume true for $0 \dots n-1$
 - Show true for n

How do we know that $P(b)$ ($b \geq a$) is true?

- $P(a)$
- $P(a) \rightarrow P(a+1)$
- $\therefore P(a+1)$
- $P(a+1) \rightarrow P(a+2)$
- $\therefore P(a+2)$
- ...
- $P(b-1) \rightarrow P(b)$
- $\therefore P(b)$

8

Example

Prove that $1+2+\dots+n = n(n+1)/2$

Base case: $n = 0$

- $0 = 0(0+1)/2 = 0$

Inductive step:

- Suppose that $n > 0$ and $1 + 2 + \dots + n-1 = (n-1)(n)/2$ (inductive hypothesis)
- Show that $1 + 2 + \dots + n = n(n+1)/2$

$$\begin{aligned}
 1 + 2 + \dots + n - 1 + n &= (1 + 2 + \dots + n - 1) + n \\
 &= \frac{(n-1)n}{2} + n \\
 &= \frac{(n-1)n + 2n}{2} \\
 &= \frac{(n-1+2)n}{2} \\
 &= \frac{(n+1)n}{2}
 \end{aligned}$$

9

Example (Strong Induction)

Any integer > 1 is divisible by some prime

- Base case
- Inductive step
 - Inductive hypothesis:
 - To show:

10

Example

Sometimes, you need to prove more than one base case

Define sequence $a_1, a_2, a_3, \dots$:

- $a_1 = 2$
- $a_2 = 6$
- for $k \geq 3$, $a_k = a_{k-1} + a_{k-2} + 8$

Prove that a_n is even for $n \geq 1$

- Base case:
- Inductive step
 - Inductive hypothesis
 - To show:

11

Sometimes Easier to Prove Stronger Result

In trying to devise a proof by mathematical induction, you may fail for two opposite reasons.

- You may fail because you try to prove too much: Your $P(n)$ is too heavy a burden.
- Yet, you may also fail because you try to prove too little: Your $P(n)$ is too weak a support.

In general, you have to balance the statement of your theorem so that the support is just enough for the burden.

—Polya

12

Example of Strengthening the Statement

Given:

- $K_0 = 1$
 $K_{n+1} = 1 + \min(2K_{\lfloor \frac{n}{2} \rfloor}, 3K_{\lfloor \frac{n}{3} \rfloor})$

Show $K_n \geq n$

- Base case:
- Inductive step:
 - Inductive hypothesis
 - To show:

13

Example

Prove that $1 + 3 + 5 + \dots + (2n-1) = n^2$

14

Example

As each of a group of business people arrives at a meeting, each shakes hands with all the other people present. Use induction to show that if n people come to the meeting, then $n(n-1)/2$ handshakes occur.

15

Example

In a round-robin tournament, show that the n teams can be labeled $T_1, T_2, \dots, T_n$ such that T_i beats T_{i+1} for all $1 \leq i \leq n-1$.

16

Loop Invariant

Format:

- //Loop precondition:
 Loop
 // Loop invariant:
 exit when ExitCondition
 Loop Contents
 End
 // Loop postcondition:

To Prove:

- loop precondition $\rightarrow$ Loop invariant
- Loop Invariant (at one iteration) $\wedge$ Loop contents $\wedge$!ExitCondition $\rightarrow$ Loop Invariant (at next iteration)
- ExitCondition is eventually true
- Loop invariant $\wedge$ ExitCondition $\rightarrow$ Loop postcondition

17

Loop Invariant

```
double total = 0;
int i = 0;
// loop precondition: i = 0, m: non-negative integer: r is real, total=0
loop
  // loop invariant: total = i * r
  exit when i = m
  total = total + r;
  i = i + 1;
end
// loop postcondition: total = m * r
```

18

Example

Prove that $1 + 3 + 5 + \dots + (2n-1) = n^2$

19

Induction and Algorithms

Algorithm correctness

- Precondition: predicates describing the initial state (involving the input variables)
- Postcondition: predicates describing the final state (involving the input and output variables)

Example:

- Product(int m, double r) // precondition: m a non-negative integer, r: a real
 // postcondition: returns m * r
- Sort(int[] y) // precondition: y is an array of integers
 // postcondition: y is a permutation of the original array such that
 $y[i] \leq y[i+1]$ for $0 \leq i < y.length$

20

Loop Invariant

Format:

- //Loop precondition:
 Loop
 // Loop invariant:
 exit when ExitCondition
 Loop Contents
 End
 // Loop postcondition:

To Prove:

- loop precondition $\rightarrow$ Loop invariant
- Loop Invariant (at one iteration) $\wedge$ Loop contents $\wedge$!ExitCondition $\rightarrow$ Loop Invariant (at next iteration)
- ExitCondition is eventually true
- Loop invariant $\wedge$ ExitCondition $\rightarrow$ Loop postcondition

Loop Invariant

```
double total = 0;
int i = 0;
// loop precondition: i = 0, m: non-negative integer: r is real, total=0
loop
  // loop invariant: total = i * r
  exit when i = m
  total = total + r;
  i = i + 1;
end
// loop postcondition: total = m * r
```