Mathematical induction

Wednesday, October 18, 2023 8:19 PM

3 Steps

1. Test
2. Assume
3. Prove

Example 1

$\sum_{k=1}^{n} 2k-1 = n^2$ for $n \geq 1$ (n odd nos)

1. Test Base case

$n=1 \quad 2(1)-1 = (1)^2$

$1 = 1$ ✓

2. Assume that stmt is true

$\sum_{k=1}^{n} 2k-1 = n^2$ —— (1)

3. Prove for n+1

$\sum_{k=1}^{n+1} (2k-1) = (n+1)^2$ —— to prove

$\sum_{k=1}^{n+1} 2k-1 = 2(1)-1 + 2(2)-1 + \dots 2(n)-1 + 2(n+1)-1$

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

using assumption

$= n^2 + 2(n+1) - 1$

$= n^2 + 2n + 1$

$= (n+1)^2$ Proved

Example 2

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

1. Test

for $n=0$, $\sum_{k=0}^{0} r^0 = \frac{1-r^1}{1-r}$

$1 = 1$ ✓

2. Assume

$\sum_{k=0}^{n} r^k = \frac{1-r^{n+1}}{1-r}$ —— (1)

3. Prove for n+1

$\sum_{k=0}^{n+1} r^k = r^{n+1} + \sum_{k=0}^{n} r^k$

using (1)

$= r^{n+1} + \frac{1-r^{n+1}}{1-r}$

$= \frac{\cancel{r^{n+1}} - r^{n+2} + 1 - \cancel{r^{n+1}}}{1-r}$

$= \frac{1-r^{n+2}}{1-r}$ proved

Example 3

$n^3 - n$ is divisible by 3 {for all n}

1. Test

for $n=0$ $\quad 0^3 - 0$ is divisible ✓

$n=1 \quad 1-1$ is divisible

2. Assume

$n^3 - n$ is divisible —— (1)

3. Prove for n+1

$(n+1)^3 - (n+1)$

$= n^3 + 3n^2 + 3n + \cancel{1} - n - \cancel{1}$

$= (n^3 - n) + 3(n^2 + n)$ Proved

Solving by Master theorem

1] $T(n) = 2T(n/2) + n^4$

$\log_2^2 = 1 < k = 4$ Case 3

$T(n) = O(n^4)$

2] $T(n) = T(7n/10) + n$

$T(n) = O(n)$ $\quad \log_{10/7}^1 < 1$ Case 3

3] $T(n) = 16T(n/4) + n^2$

$\log_4^{4^2} = 2 = 2(k)$

$T(n) = O(n^2 \log n)$

Either select n^k or $n^{\log_b^a}$

or both (multiply with $\log n$)

Properties of log

1) $\log ab = \log a + \log b$

2) $\log\left(\frac{a}{b}\right) = \log a - \log b$

3) $\log a^b = b \log a$

4) $a^{\log_c^b} = b^{\log_c^a}$

5) $a^b = n \rightarrow b \log a = \log n \rightarrow b = \log_a^n$

Reviewing Asymptotic Notations

1. Big-Oh $\quad f(n) \leq c\,g(n)$ for all $n > 0$, $c > 0$

$0 \leq \lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty$ $\quad f(n) = O(g(n))$

imp

2. Big-theta $\quad c_1 g(n) \leq f(n) \leq c_2 g(n)$ for all $n > 0$, $c_1, c_2 > 0$

$0 < \lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty$

Equivalence function $\quad f(n) = \Theta(g(n))$

3. Big-Omega $\quad f(n) \geq c\,g(n)$ $\quad n > 0$, $c > 0$

$0 < \lim_{n\to\infty} \frac{f(n)}{g(n)} \leq \infty$ $\quad f(n) = \Omega(g(n))$

Master theorem for Decreasing functions

$T(n) = a\,T(n-b) + O(n^k)$

1] $a = 1$, $T(n) = O(n \cdot n^k)$

2] $a > 1$, $b \geq 1$, $T(n) = O(a^{n/b} \cdot n^k)$

3] $a < 1$, $T(n) = O(n^k)$

Master theorem for Dividing functions

$T(n) = aT(n/b) + \Theta(n^k \log^p n)$

Amt of work done inside the recurrence

Amt of work done by the recursion

$>$ 1] if $\log_b^a > k$, $T(n) = O(n^{\log_b^a})$

$=$ 2] if $\log_b^a = k$,

if $p > -1$, $O(n^k \cdot \log^{p+1} n)$

if $p = -1$, $O(n^k \log\log n)$

if $p < -1$, $O(n^k)$ ← log becomes insignificant

$<$ 3] if $\log_b^a < k$,

$T(n) = O(n^k \log^{\max(0,p)} n)$

we compare k & $\log_b^a$

who will take over

Simpler way to solve dividing functions

Case 2, directly take $O(n^k)$ & multiply by $\log n$

Case 3, directly take $f(n)$

Case 1, self explanatory.

How to approach a master theorem problem

1] calculate $\log_b^a$.

2) compare it with k

3] find out which one grows faster.

Regularity condition for case 3

$a f(n/b) \leq c \cdot f(n)$ for $c < 1$

Prove by assuming vals for c & n.