The principle of mathematical induction (generalized weak form)

Choose two constant integers $n_0 \in \mathbb{N}_0$ and $k \in \mathbb{Z}^+$.

$S(n)$ is a statement about an integer $n \geq n_0$.

(1) $S(n_0), S(n_0+1), S(n_0+2), \ldots, S(n_0+k-1)$ are true

(2) $\forall n \geq n_0$, if $S(n), S(n+1), S(n+2), \ldots, S(n+k-1)$ are true, then $S(n+k)$ is also true.

Then $S(n)$ is true for all $n \geq n_0$.

Last class: $k = 1$.

Proof: Exercise (use the well-ordering principle)

__Example 1__: Lucas numbers L_n

$$L_0 = 2, \quad L_1 = 1,$$

$$L_n = L_{n-1} + L_{n-2} \text{ for all } n \geq 2.$$

$$L_n = F_{n-1} + F_{n+1} \text{ for all } n \geq 1.$$

__Proof__ $k = 2$

[Basis] $n = 1$. $L_1 = 1$ $\quad F_0 + F_2 = 0 + 1 = 1$

two $\quad n = 2$ $\quad L_2 = L_0 + L_1 = 3, \quad F_1 + F_3 = 1 + 2 = 3$

[Induction] Take $n \geq 1$. Assume that

$$\left.\begin{aligned} L_n &= F_{n-1} + F_{n+1} \\ L_{n+1} &= F_n + F_{n+2} \end{aligned}\right\} \text{two}$$

$$\begin{aligned} L_{n+2} = L_n + L_{n+1} &= (F_{n-1} + F_n) + (F_{n+1} + F_{n+2}) \\ &= F_{n+1} + F_{n+3} \\ &= F_{(n+2)-1} + F_{(n+2)+1} \end{aligned}$$

Example 2 : 4-Re and 7-Re coins.

$n \in \mathbb{Z}^+$. If $n \geq 18$, then n Rs can be exchanged by coins of the given denominations.

Proof: $n_0 = 18$, $k = 4$.

[Basis]

$$\left.\begin{aligned} 18 &= 4+7+7 \\ 19 &= 4+4+4+7 \\ 20 &= 4+4+4+4+4 \\ 21 &= 7+7+7 \end{aligned}\right\} \text{4 cases}$$

[Induction] $n \geq 18$ and $\overbrace{n, n+1, n+2, n+3}^{\text{4 previous values}}$ Rs can be exchanged.

$n+4$ (just add another 4-Re coin)

Frobenius coin problem

$a, b \to$ two denominations

What is the largest n that cannot be exchanged by coins of these denominations?

$d = \gcd(a, b) > 1$

$\gcd(a, b) = 1.$

$a = 4, b = 7$

17 Rs cannot be exchanged.

Sylvester proved that this largest amount is

$$(a-1)(b-1) - 1.$$

Only super-interested students may try to prove.

Example 3

$a_0 = 0,\ a_1 = 1,\ a_2 = 2,$

$a_n = a_{n-1} + a_{n-2} + a_{n-3} \quad \forall n \geq 3.$

Then, $a_n < 3^n$ for all $n \geq 0$.

Proof: $n_0 = 0,\ k = 3$

[Basis]

$n = 0 \qquad a_0 = 0 < 1 = 3^0$

$n = 1 \qquad a_1 = 1 < 3 = 3^1$

$n = 2 \qquad a_2 = 2 < 9 = 3^2$

[Induction] $n \geq 0$. Assume

$a_n < 3^n$

$a_{n+1} < 3^{n+1}$

$a_{n+2} < 3^{n+2}$

Then $a_{n+3} = a_{n+2} + a_{n+1} + a_n$

$< 3^{n+2} + 3^{n+1} + 3^n < 3^{n+2} + 3^{n+2} + 3^{n+2}$

$= 3 \times 3^{n+2} = 3^{n+3}$

Principle of mathematical induction
[Strong form]

Let $S(n)$ be a statement about $n \in \mathbb{Z}^+$ such that

(1) $S(1)$ is true

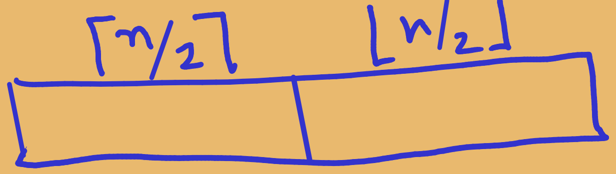
(2) $\forall\, n \geq 1$, if all of $S(1), S(2), S(3), \ldots, S(n)$ are true, then $S(n+1)$ is true.

Then $S(n)$ is true for all $n \in \mathbb{Z}^+$.

Proof: Exercise.

Note: We may start from any $n_0 \in \mathbb{N}_0$.

Merge sort



n integers

$n = 0, 1$ return.

two recursive calls

merging process

$$T(0) = T(1) = 1$$

$$T(n) = T(\lceil n/2 \rceil) + T\left(\left\lfloor \tfrac{n}{2} \right\rfloor\right) + n \quad \text{For } n \geq 2.$$

$T(n)$ is an increasing (non-decreasing) function of n.

$$T(n) \leq T(n+1) \quad \forall\, n \geq 0.$$

Proof :

[Basis] $n = 0$ $T(0) = T(1) = 1$

$$T(0) \leq T(1)$$

[induction] $T(0) \leq T(1) \leq T(2) \leq \cdots \leq T(n) \leq T(n+1)$

To prove $T(n+1) \leq T(n+2)$

n is even $n = 2k$ $n+1 = 2k+1$ $T(n+2) = T(k+1) + T(k+1) + n+2$

$n+2 = 2k+2$ $T(n+1) = T(k+1) + T(k) + n+1$

$$T(n+2) - T(n+1) = T(k+1) - T(k) + 1$$

$$T(n+1) \leq T(n+2) \quad \geq 0 + 1 = 1 \geq 0$$

n is odd

$n = 2k+1$ $T(n+2) - T(n+1) = [T(k+2) + T(k+1) + n+2]$

$- [T(k+1) + T(k+1) + n+1]$

$$= T(k+2) - T(k+1) + 1 \geq 0$$