Chapter 5: Induction and Recursion

Trần Hòa Phú

Ngày 6 tháng 6 năm 2023

5.1 Mathematical Induction

Quy nạp thường

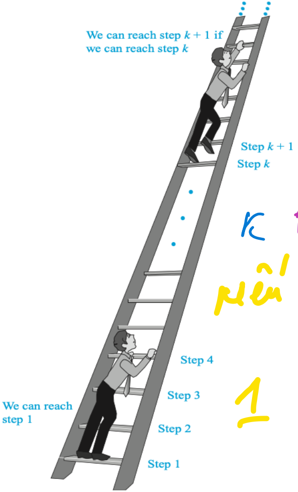


FIGURE 1 Climbing an Infinite Ladder.

nếu leo được bậc thứ k thì cũng leo được bậc k+1.
cm mđ cm mđ

nếu ông ta leo được bậc thứ 1 => leo được hết bậc.
cm mđ

Theorem

To prove that $P(n)$ is true for all positive integers n, where $P(n)$ is a propositional function, we complete two steps:

Basic Step we verify that $P(1)$ is true.

Inductive Step Assume that $P(k)$ is true for k is an arbitrary positive integer. We prove that $P(k+1)$ is also true.

Example Show that if n is a positive integer, then

$$1 + 2 + \ldots + n = \frac{n(n+1)}{2}$$

B1: cm mđ P(1) True

B2: Cho P(k) True. Cm P(k+1) True

⇒ Cm tất cả mđ P(n) ∀n ≥ 1

Solution

$P(n)$: "$1 + 2 + \ldots + n = \dfrac{n(n+1)}{2}$".

Basic step $P(1)$: "$1 = \dfrac{1.2}{2}$". So $P(1)$ is true.

Inductive step Assume that $P(k)$ is true for k arbitrary positive integer:

$$1 + 2 + \ldots + k = \frac{k(k+1)}{2}$$

And we need to prove that $P(k+1)$ is also true.

$$1 + 2 + \ldots + k + k + 1 = \frac{(k+1)(k+2)}{2}$$

In fact,

$$1 + 2 + \ldots + k + k + 1 = \frac{k(k+1)}{2} + k + 1 = \frac{(k+1)(k+2)}{2}$$

Therefore, $P(k+1)$ is also true.

In conclusion, by mathematical induction, $P(n)$ is true for all positive integers n.

Exercise

1. Show that $(n^3 - n) \vdots 3$ for every positive integer n $n \geq 0$

2. Use mathematical induction to show that

$$1 + 2 + 2^2 + \ldots + 2^n = 2^{n+1} - 1$$

for all nonnegative integers n. $n \geq 0 \rightarrow$ basis step: $P(0)$:

$P(0)$: $1 + 2 + 2^2 + \ldots + 1 = 2^1 - 1$ (True)

$P(k)$: $1 + 2 + 2^2 + \ldots + 2^k = 2^{k+1} - 1$ (GS:)

$P(k+1)$: $1 + 2 + 2^2 + \ldots + 2^k + 2^{k+1} = 2^{k+2} - 1$

$2^{k+1} - 1 + 2^{k+1} = 2^{k+2} - 1$ (True)

$2^1 \cdot 2^{k+1} - 1$

$2^{k+2} - 1$

Strong Induction

Quy nạp mạnh

Theorem

To prove that $P(n)$ is true for all positive integers n, we complete two steps:

$P(n)$ T $\forall n \geq 1$

Basic step *verify that $P(1)$ is true.* : $P(1)$ True.

Inductive step *Assume that $P(1), P(2), .., P(k)$ are true for k is an arbitrary positive integer. Then we need to prove $P(k+1)$ is also true.*

GS: $P(1)$, $P(2)$, .. $P(k)$ True.

Prove $P(k+1)$ True.

Example

Prove that every amount of postage of 12 cents or more can be formed using just 4-cent and 5-cent stamps. 'M... postage of n cents can be...' $\forall n \geq 12$

	4	5
12	3	
13	2	1
14	1	2
15		3
16	4	

basis step: P(12): 'postage 12 can be ....' True.

inductive step: P(12), P(13), ..., P(k) True

+1 tem 4 Cm: P(k+1) True.

" mọi postage có giá 12, 13, ... k cents can be ... "

+1 tem 4 → P(k+1) " —— (k+1) cent can be ... "

18 3 1 2 2 +1 Tem 4. : k+1 - 3 k - 3

4 5
k-3 a b
k+1 a+1 b

Exercise

Consider the problem:

Prove that P(n) = "for all n ≥ 12 we have n = 4a + 5b with a, b non-negative integers" is true.

In the strong induction proof, assuming that P(k) is true for some k, in order to prove P(k+1) is true, we should ________

A. use P(k - 3) = "k - 3 = 4x + 5y, (x, y non-negative integers)" is true and k + 1 = (k - 3) + 4. → k - 3 + 4

4x+5y + 4

= 4(x+1) + 5y.

B. use P(k - 1) = "k - 1 = 4x + 5y, (x, y non-negative integers)" is true and k + 1 = (k - 1) + 2.

4x+5y + 2

C. use P(k - 2) = "k - 2 = 4x + 5y, (x, y non-negative integers)" is true and k + 1 = (k - 2) + 3.

4x+5y + 3

gom thành

dạng 4x + 5y đc

5.3 Recursive Definitions and Structural Induction

là cách định nghĩa đối tượng thông qua chính đối tượng đó

Definition

Recursion is a way to define an object in terms of itself.

Example Define a sequence $a_n = 2^n \quad n = 1, 2, \ldots$ by recursion.

Solution

$a_n = 2^n, n \geq 1.$ để viết đệ quy

$a_n = 2.a_{(n-1)}$: quy luật

$a_0 = 1$ → số ban đầu

$$a_n = 2.2^{n-1} = 2.a_{n-1} \qquad n = 1, 2, 3, \ldots$$

$$a_0 = 1$$

$a_1 = 2$

$a_2 = 4$

$a_3 = 8.$

$a_n = 2^n$

$a_{n-1} = 2^{n-1}$

$\frac{a_n}{a_{n-1}} = \frac{2^n}{2^{n-1}} = 2$

$a_1 = 2a_0 = 2$

$a_2 = 2a_1 = 4$

$a_3 = 2a_2 = 8.$

$\begin{cases} a_n = 2 \cdot a_{n-1} \\ a_0 = 1 \end{cases}$

Exercise

Give a recursive definition of the sequence $\{a_n\}$, $n = 1, 2, 3, \dots$ if

a) $a_n = 6n$

$a_{n-1} = 6(n-1)$

$a_n - a_{n-1} = 6$.

$\begin{cases} a_n = a_{n-1} + 6 \\ a_0 = 0. \end{cases}$

b) $a_n = 2n + 1$

(b) $a_n = 2n+1$.

$a_{n-1} = 2(n-1)+1 = 2n-1$

$\to \begin{cases} a_0 = 1. \\ a_n = a_{n-1} + [(2n+1) - (2n-1)] \end{cases}$

$= a_{n-1} + 2.$

c) $a_n = 10n$

c). $a_n = 10n$

$a_{n-1} = 10(n-1) = 10n - 10$

$\to \begin{cases} a_0 = 0. \\ a_n = a_{n-1} + 10 \end{cases}$

d) $a_n = n(n+1)$

d) $a_n = n(n+1)$

$a_{n-1} = (n-1)(n-1+1) = n(n-1)$

$\Rightarrow \begin{cases} a_0 = 0 \\ a_n = a_{n-1} + [n(n+1) - n(n-1)] \end{cases}$

$= a_{n-1} + 2n.$

f) $a_n = 1 + (-1)^n$

$a_{n-2} = 1 + (-1)^{n-2}$

$\{\to\ a_n = a_{n-2} + (-1)^n - \frac{(-1)^n}{(-1)^2} = a_{n-2}.$ $\quad a_0 = 2$, $a_1 = 0$

(f) $a_n = 1 + (-1)^n$

$a_{n-1} = 1 + (-1)^{n-1} = 1 + \frac{(-1)^n}{(-1)^1}$

$\to \begin{cases} a_0 = 2 \\ a_n = a_{n-1} + (-1)^n - \frac{(-1)^n}{(-1)} \end{cases}$

$= a_{n-1} + (-1)^n + (-1)^n$

$= a_{n-1} + 2 \cdot (-1)^n$

1) $F(n) = 1 + 2 + \ldots + n$

$F(n-1) = 1 + 2 + \ldots + n - 1$

$F(n) - F(n-1) = n$

$\Rightarrow F(n) = F(n-1) + n$

$F(1) = 1.$

2) $F(n) = 1 \cdot 2 \cdot 3 \ldots n$

$F(n-1) = 1 \cdot 2 \cdot 3 \ldots (n-1)$

$\Rightarrow \frac{F(n)}{F(n-1)} = n.$

$\Rightarrow F(n) = n \cdot F(n-1)$

$F(1) = 1 \ldots$

Exercise

1) Let $F(n)$ be the sum of the first n positive integer. Give a recursive definition of $F(n)$.

2) Let $G(n)$ be the product of the first n positive integer. Give a recursive definition of $G(n)$.

Recursively Defined Sets and Structures

Example

Consider the subset S of the set of integers recursively defined

$3 \in S$

$x, y \in S \rightarrow x + y \in S$

Basic step: $3 \in S$

Recursive step: If $x \in S, y \in S$, then $x + y \in S$

$x = 3;\ y = 6.$

$\rightarrow S = \{3, 6, 9, 12, 15 \ldots\}$

$3 \in S$

$6 \in S$

$\rightarrow 9 \in S \xrightarrow{+3} 12 \in S \xrightarrow{+3} 15 \in S$

Exercise

Give a recursive definition of each of these sets

a) $A = \{2, 5, 8, 11, 14, \dots\}$ $\begin{cases} 2 \in A \\ x \in A \to x + 3 \in A. \end{cases}$

b) $B = \{\dots, -5, -1, 3, 7, 10, \dots\}$ $\begin{cases} x \in B \\ x \in B \to x + 4 \in B \end{cases}$

c) $C = \{3, 12, 48, 192, 768, \dots\}$ $\begin{cases} 3 \in C \\ x \in C \to x \cdot 4 \in C \end{cases}$

index 1 2 3 4 5 6

d) $D = \{1, 2, 4, 7, 11, 16, \dots\}$ $\begin{cases} 1 \in D \\ x \in D \to x + i(x) \in D \end{cases}$

vị trí của x trong dãy

The set Σ^* of string

Definition

The set Σ^* of strings over the finite set (alphabet) Σ is defined recursively by

- Basis step $\lambda \in \Sigma^*$ (λ is the empty string containing no symbols)
- Inductive step If $w \in \Sigma^*$ and $x \in \Sigma$, then $wx \in \Sigma^*$.

$\lambda \in \Sigma^*$

$\left.\begin{array}{l} x \in \Sigma \\ w \in \Sigma^* \end{array}\right\} \rightarrow wx \in \Sigma^*$

Example

$\Sigma = \{1\} \Rightarrow \Sigma^* = \{\lambda, 1, 11, 111, \ldots\}$: is set of string made by 1 with arbitrary length.

$\Sigma = \{1\} \to \Sigma^*$?

$11, 1, 1, \lambda \in \Sigma^*$ $\quad 1111, 111, 11, 1 \in \Sigma^*$

$1 \in \Sigma$

Exercise

Find Σ^* if $\Sigma = \{0, 1\}$.

$00, 01$

$, 10, 11, 0, 1, \lambda \quad \Sigma^*$

$0, 1 \in \Sigma$

Exercise

Let S be the subset of the set of ordered pairs of integers defined recursively by

$(0,0) \in S.$

$\bar{a}\ \bar{b}.$

(2 3)
(4 6)
(6 9)
(8 12)
(10 15)

(0 0) ∈ S
a b
(3 2)
(6 4)
(9 6)
(12 8)

Basis step: $(0,0) \in S$

Recursive step: If $(a,b) \in S$, then

$(a+2, b+3) \in S$ and $(a+3, b+2) \in S$

A.(8,15) B.(9,15) C.(10,15) D.(12,15)

Exercise

Let S be the set defined recursively as follows:

Basis step: $2 \in S$ → phần tử đầu $2 \in S.$

Recursive step: If $x \in S$, then $2x \in S$

2, 4, 8, 16, 32. 2^n

What is S?

A. $S = \{2^n | n = 1, 2, \ldots\}$

B. $S = \{2^{2n} | n = 1, 2, \ldots\}$

C. $S = \{2n | n = 1, 2, \ldots\}$

Recursive Algorithms

Definition

An algorithm is called recursive if it solves a problem by reducing it to an instance of the same problem with smaller input.

Algorithm 1: A Recursive Algorithm for Computing $n!$

procedure factorial(n: nonnegative integer)

if $n = 0$ **then return** 1
else return n. factorial($n - 1$)

{ output is $n!$}

factorial(4)

= 4. factorial(3)

3. factorial(2)

2. factorial(1)

~~2~~ 1.

4!

Algorithm 2: A Recursive Algorithm for Computing a^n

procedure power (a : nonzero real number, n: nonnegative integer)

$a \in \mathbb{R} \setminus \{0\}$ $n \geq 0$.

if $n = 0$ **then return 1**
else return a. power$(a, n-1)$
{ output is a^n}

power$(3, 4)$
$= 3 \cdot$ power$(3, 3)$
$\cdot\ 3 \cdot$ pow$(3, 2)$
$\cdot\ 3$ pow$(3, 1)$
3 pow$(3, 0)$
$= 1.$

$\rightarrow 3^4$

Algorithm 3: A Recursive algorithm for computing $\gcd(a, b)$

procedure $\gcd(a, b$: nonnegative integers with $a < b)$

$0 \le a < b$

if $a = 0$ **then return** b

else return $\gcd(b \textbf{ mod } a, a)$

$\{$ output is $\gcd(a, b)\}$ a

$\gcd(\underset{a}{24}, \underset{b}{36})$.

$= \gcd(\underbrace{36 \bmod 24}_{12}, 24)$ $\gcd = 12$

$= \gcd(0, 12)$

cho đến khi $a = 0$ → $\gcd = b$.

Algorithm 4: A Recursive Linear Search Algorithm

search(i,j,x) is the procedure that search for first occurrence of x in the sequence $a_i, a_{i+1}, \ldots, a_j$.

procedure search(i, j, x: i, j, x , $1 \leq i \leq j \leq n$)

if $a_i = x$ **then**

  return i

else if $i = j$ **then**

  return 0

else

  return search($i + 1, j, x$)

{ output is the location of x in $a_1, a_2, \ldots, a_n$ if it appears, otherwise it is 0}

tìm số x trong dãy từ $i \to j$

$x = 4$

a_1	a_2	a_3	a_4	a_5
1	4	2	3	7

Search(i, j, x) : $a_i \to a_j$

Search$(1, 5, 4)$

If $(a_i = x)$ F

else if $i = j$ F

search$(2, 5, 4)$

If $(a_i = x)$ T

$\to i = 2$

Algorithm 5: A Recursive Binary Search Algorithm

procedure *binary search*$(i, j, x$: i, j, x integers, $1 \le i \le j \le n)$
$m := \lfloor (i+j)/2 \rfloor$
if $x = a_m$ **then**
  return m
else if $(x < a_m$ **and** $i < m)$ **then**
  return *binary search*$(i, m-1, x)$
else if $(x > a_m$ **and** $j > m)$ **then**
  return *binary search*$(m+1, j, x)$
else return 0
{output is location of x in $a_1, a_2, \ldots, a_n$ if it appears; otherwise it is 0}

$x = 7$:

a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
1	2	3	5	6	7	8	10

search(1, 8, 7) $m = \left\lfloor \frac{i+j}{2} \right\rfloor = \lfloor 4,5 \rfloor = 4.$

If $x = a_m$ False.

else if: $(x < a_m$ and $i < m)$ False.

else if $(x > a_m\ (a_4)$ and $j > m)$

return. binarysearch$(m+1, j, x)$ → 5, 8, 7

$m = \left\lfloor \frac{5+8}{2} \right\rfloor = 6$

If $(x = a_m)$ ⇒ return $m = 6$

$a_6 = 7.$

Example: To search for 8 in the list
1, 2, 3, 5, 6, 7, 8, 10

Proving Recursive Algorithms Correct

Example

Show that the following algorithm is correct by using mathematical induction.

"factorial(n) = n!" $\forall n \geq 0$

procedure factorial(n: nonnegative integer)

P(n): "factorial(n) = n!"

if $n = 0$ **then return** 1

B1: Check: P(0): "factorial(0) = 0!"

True.

else return n. factorial($n - 1$)

B2: GS: P(k): "fac(k) = k!" True.

{ output is n!}

Cm: fac(k+1) = (k+1)!

(k+1) . fac(k)

= (k+1)! k! (GS bước 2)

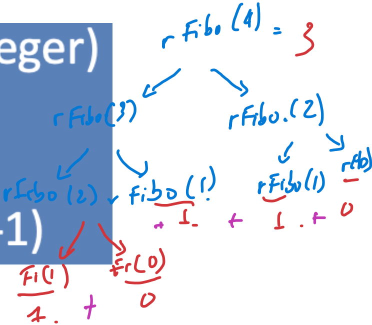
Recursion and Iteration

A Fibonacci sequence is

$$f_n = f_{n-1} + f_{n-2} \; \forall n \geq 2, \quad f_0 = 0, f_1 = 1 \tag{1}$$

```
procedure rfibo (n: nonnegative integer)
If n=0 then rFibo(0)=0
Else if n=1 then rFibo(1)=1
Else rFibo(n) := rFibo(n-2) + rFibo(n-1)
```

recursion



```
procedure  iFibo (n: nonnegative integer)
If n=0 then y:=0
Else if  n=1 then y:=1
Else Begin
        x:=0 ;  y:=1
        for i:= 2 to n
            Begin
                z:= x+y;   x:= y;  y:=z
            End
    End  { iFibo(n) = z}
```

iFibo(4)

$x=0; y=1$

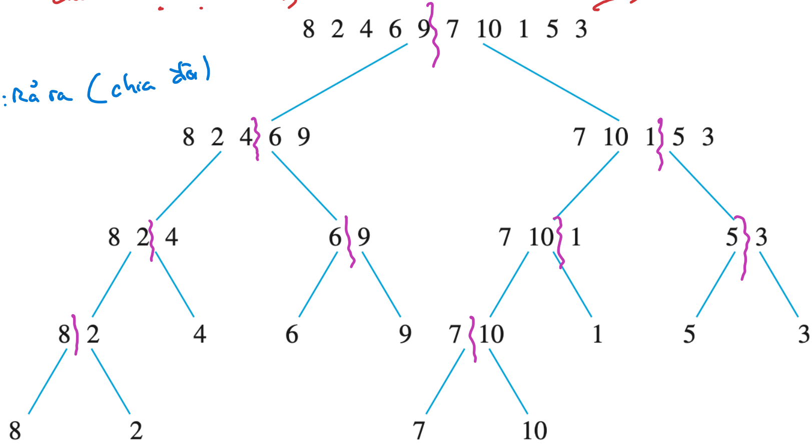
$i=2$	$i=3$	$i=4$
$z=1$	$z=2$	$z=3$
$x=1$	$x=1$	$x=2$
$y=1$	$y=2$	$y=3$

Merge Sort

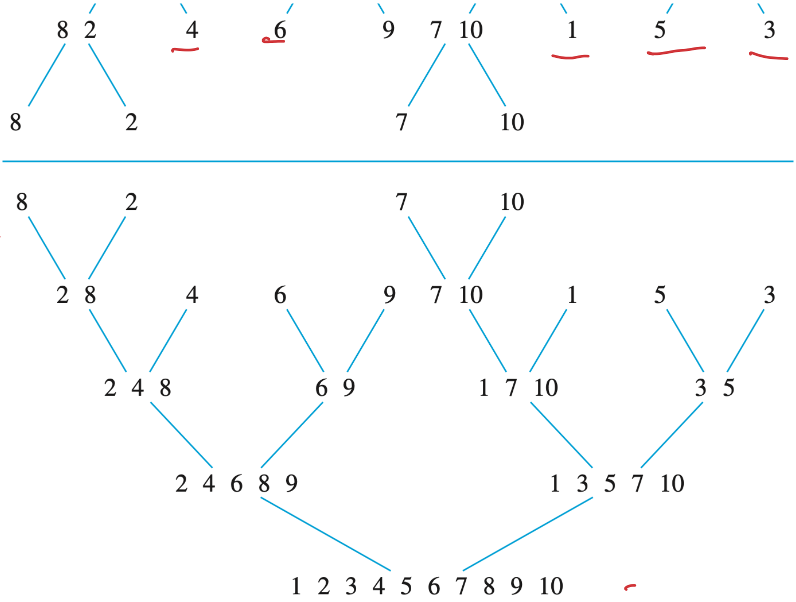
đầu vào lộn xộn

đầu ra tăng dần ↑

B1: Rã ra (chia đôi)



Trộn lại:


8 2
4
6
9
7 10
1
5
3
8
2
7
10
8
2
7
10
2 8
4
6
9
7 10
1
5
3
2 4 8
6 9
1 7 10
3 5
2 4 6 8 9
1 3 5 7 10
1 2 3 4 5 6 7 8 9 10

ALGORITHM 9 A Recursive Merge Sort.

procedure *mergesort*$(L = a_1, \ldots, a_n)$
if $n > 1$ **then**
    $m := \lfloor n/2 \rfloor$
    $L_1 := a_1, a_2, \ldots, a_m$
    $L_2 := a_{m+1}, a_{m+2}, \ldots, a_n$
    $L := merge(mergesort(L_1),\ mergesort(L_2))$
$\{L$ is now sorted into elements in nondecreasing order$\}$

TABLE 1 Merging the Two Sorted Lists 2, 3, 5, 6 and 1, 4.

First List	*Second List*	*Merged List*	*Comparison*
2 3 5 6	1 4		$1 < 2$
2 3 5 6	4	1	$2 < 4$
3 5 6	4	1 2	$3 < 4$
5 6	4	1 2 3	$4 < 5$
5 6		1 2 3 4	
		1 2 3 4 5 6	

ALGORITHM 10 Merging Two Lists.

procedure *merge*(L_1, L_2: sorted lists)
L := empty list
while L_1 and L_2 are both nonempty
  remove smaller of first elements of L_1 and L_2 from its list; put it at the right end of L
  if this removal makes one list empty **then** remove all elements from the other list and
    append them to L
return L {L is the merged list with elements in increasing order}

Exercise

Consider the following algorithm

procedure $F(a_1, a_2, \dots, a_n : \text{integers})$

if $n = 0$ then return 0

else return $a_n + F(a_1, a_2, \dots, a_{n-1})$ → tổng của tất cả các số

Find

a) $F(\overset{a_1}{1}, \overset{a_2}{3}, \overset{a_3}{5})$ b) $F(1, 2, 3, 5, 6) = 1 + 2 + 3 + 5 + 6 = 17$

$= a_3 + F(a_1, a_2)$

$= a_3 + a_2 + F(a_1)$

$= a_3 + a_2 + a_1 + \underbrace{F(a_0)}_{=0} = 1 + 3 + 5 = 9$

Exercise

Let $P(n)$ be the statement that $n! < n^n$, where n is an integer greater than 1. What do you need to prove in the basis step if using induction method?

A. Show that $P(2)$ is true

B. Show that $P(3)$ is true

C. Show that $P(4)$ is true

D. Show that $P(1)$ is true

Exercise

Which of the following algorithms are recursive?

```
(i) procedure A(n: nonnegative even integer)
    if n = 0 then A(n):= 1;
   else A(n):=A(n-2)*3

(ii) procedure A(n: nonnegative even integer)
     if n = 0 then y: = 1;
     else
     begin
          y: = 1; m = n div 2;
          for i:=1 to m
               y: = y*3;
     end
```

A. Only (i)

B. Both of them

C. Only (ii)

D. None of them

GS: $A(5)$

$A(5) = A(3)*3$ → Recursive

$A(1)*3$.

Exercise

Find a recursive definition for the set of all integers divisible by 3.

A. $3, -3 \in S$ and if $a \in S$ then $3a \in S$
B. $3 \in S$ and if $a, b \in S$ then $a - b \in S$
C. $3 \in S$ and if $a, b \in S$ then $a + b \in S$
D. $3 \in S$ and if $a \in S$ then $3a \in S$.

Exercise

Give a recursive definition of the sequence

$$a_n = 5n, n = 1, 2, \ldots$$

i. $a_0 = 0, a_n = a_{n-1} + 5$ for $n = 1, 2, 3, \ldots$

ii. $a_1 = 1, a_n = a_{n-1} + 5$ for $n = 2, 3, \ldots$

iii. $a_1 = 5, a_{n-1} = a_n + 5$ for $n = 2, 3, \ldots$

iv. $a_1 = 5, a_n = a_{n-1} + 5$ for $n = 2, 3, ..$

Exercise

Find a recursive definition for the set of all positive integers NOT divisible by 4

i. If $a \in S$ then $a + 4 \in S$

ii. $1, 2, 3 \in S$. If $a \in S$, then $a + 4, a - 4 \in S$

iii. If $a \in S$, then $a - 4, a + 4 \in S$

iv. $1, 2, 3 \in S$. If $a \in S$, then $a + 4 \in S$

A. i B. ii C. iii D. iv

Exercise

Given the recursive algorithm that computes the n-th Fibonacci numbers

Procedure F(n: natural numbers)

If $n = 0$ then $F(n) := 0$
else
    If $n = 1$ then $F(n) := 1$
    else $Fn) := F(n-1) + F(n-2)$

How many additions are used if $n = 6$?

A. 7   B. 8   C. 9   D. 12

Exercise

Let S be the set defined recursively by:

$5 \in S$

If $x \in S$ then $x + 5 \in S$. What is S?

$A.$ $S = \{5, 10, 15, 20, \ldots\}$

$B.$ $S = \{0, 5, 10, 15, \ldots\}$

$C.$ $S = \{0, 1, 2, 3, 4, \ldots\}$

$D.$ $S = \{\ldots, -10, -5, 0, 5, 10, \ldots\}$