

離散數學 - 筆記

Author: 國立臺北科技大學 109級資工系 黃漢軒

參考書籍：

- Kenneth H. Rosen, Discrete Mathematics and Its Applications Seventh Edition
- Kenneth H. Rosen, 謝良瑜, 陳志賢, 離散數學 第七版

若筆記侵權將會迅速移除，請聯繫 sigtuantw@gmail.com。

2021/05/27 筆記突破80000字

2021/06/05 筆記突破90000字

2021/06/09 筆記突破100000字

1. 基礎：邏輯與證明

1.1 邏輯命題

Introduce - 命題

命題通常都是一個明確的陳述句，只會有True或者False兩種結果，不會有兩種結果同時出現的可能。

Example 1

1. Washington, D.C., is the capital of the United States of America.
2. Toronto is the capital of Canada.
3. $1 + 1 = 2$.
4. $2 + 2 = 3$.

命題 1 和 3 是 True，而命題 2 和 4 是 False

Example 2

1. What time is it?
2. Read this carefully.
3. $x + 1 = 2$.
4. $x + y = z$.

1 和 2 不是命題，因為他們並不是陳述句

3 和 4 不是命題，因為他們並沒辦法用True或者False回答

我們習慣用一些英文字母來當作命題變數(例如 $p, q, r, s\dots$)，也就是用英文字母當作命題，就像數字的代數一樣。

如果命題是true，我們習慣用T來表示這個命題是true，而如果命題是false，則會使用F來表示這個命題是false。

Definition - 邏輯非

令 p 為一命題， p 的邏輯非，我們表示為 $\neg p$ ，或者也可以表示為 \overline{p} ，表示在 p 的條件下，結論不成立。

通常來說， $\neg p$ 讀做"not p "，而 $\neg p$ 的值與 p 的值互為反相，也就是若 p 是true，則 $\neg p$ 就是false，反之亦然。

Example 1

Find the negation of the proposition

"Michael's PC runs Linux."

and express this in simple English.

The negation of "Michael's PC runs Linux" is "It's not the case that Michael's PC runs Linux."

This negation can be more simply expressed as "Michael's PC doesn't run Linux."

Example 2

Find the negation of the proposition

"Vandana's smartphone has at least 32GB of memory"

and express this in simple English.

The negation of "Vandana's smartphone has at least 32GB of memory" is "It's not the case that Vandana's smartphone has at least 32GB of memory".

The negation can be more simply expressed as "Vandana's smartphone does not have at least 32GB of memory"

or even more simply as "Vandana's smartphone has less than 32GB of memory"

Table - 邏輯非的真值表

| TABLE 1 The Truth Table for the Negation of a Proposition. | |
|--|----------|
| p | $\neg p$ |
| T | F |
| F | T |

Definition - 邏輯與

令 p 與 q 為命題， p 與 q 的邏輯與，我們表示為 $p \wedge q$ ，念作" p and q "。

當 p 與 q 都為True時， $p \wedge q$ 才會true，反之為false。

Example

Find the conjunction of the propositions p and q where p is the proposition "Rebecca's PC has more than 16 GB free hard disk space" and q is the proposition "The processor in Rebecca's PC runs faster than 1 GHz."

The conjunction of the propositions p and q is

"Rebecca's PC has more than 16 GB free hard disk space, and the processor in Rebecca's PC runs faster than 1 GHz."

Table - 邏輯與的真值表

| TABLE 2 The Truth Table for the Conjunction of Two Propositions. | | |
|--|-----|--------------|
| p | q | $p \wedge q$ |
| T | T | T |
| T | F | F |
| F | T | F |
| F | F | F |

Definition - 邏輯或

令 p 與 q 為命題， p 與 q 的邏輯或，我們表示為 $p \vee q$ ，念作" p or q "。

當 p 與 q 都為false時， $p \vee q$ 為false，反之為true。

Example

What is the disjunction of the propositions p and q where p and q are the same propositions as in Example 5?

The disjunction of the propositions p and q is

"Rebecca's PC has more than 16 GB free hard disk space, or the processor in Rebecca's PC runs faster than 1 GHz."

Table - 邏輯或的真值表

| TABLE 3 The Truth Table for the Disjunction of Two Propositions. | | |
|--|-----|------------|
| p | q | $p \vee q$ |
| T | T | T |
| T | F | T |
| F | T | T |
| F | F | F |

Definition - 邏輯異或

令 p 與 q 為命題， p 與 q 的邏輯異或，我們表示為 $p \oplus q$ 。

當 p 或 q 都同為true或同為false時， $p \oplus q$ 為false，反之為true。

Definition - 實質蘊涵

令 p 與 q 為命題， p 與 q 的 實質蘊涵，我們表示為 $p \rightarrow q$ ，表示若 p 則 q 。

當 p 為true且 q 為false時，則 $p \rightarrow q$ 為false，否則為true。

在實質蘊涵中的 p ，我們稱作前件，而 q 我們稱作後件

Table - 實質蘊涵的真值表

| TABLE 5 The Truth Table for the Conditional Statement $p \rightarrow q$. | | |
|---|-----|-------------------|
| p | q | $p \rightarrow q$ |
| T | T | T |
| T | F | F |
| F | T | T |
| F | F | T |

Example

若期末考考100分，那你就會拿到A。

合理(T)

P：若期末考考100分(T)，則Q：你會拿到A(T)

P：若期末考沒考100分(F)，則Q：你不會拿到A(F)

P：若期末考沒考100分(F)，則Q：你依然可能拿到A(T)

不合理(F)

P：若期末考考了100分(T)，則Q：沒拿到A(F)

Definition - 換位命題

令 p 與 q 為命題， p 與 q 的換位命題，我們表示為 $q \rightarrow p$ 。

當 p 為false且 q 為true時， $q \rightarrow p$ 為false，否則為true。

Example

若期末考考100分，那你就會拿到A。

合理(T)

Q : 拿到A · 則P : 期末考考了100分

Q : 沒有拿到A · 則P : 期末考沒有考100分

Q : 拿到A · 則P : 期末考沒有考100分

不合理(F)

Q : 沒有拿到A · 則P : 期末考考100分

Table - 換位命題的真值表

| P | Q | $Q \rightarrow P$ |
|---|---|-------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Definition - 換質換位命題

令 p 與 q 為一個命題

則一條件敘述 $\neg q \rightarrow \neg p$ 稱作換質換位命題

質位互換命題與假設邏輯等價(也就是 $(\neg q \rightarrow \neg p) \leftrightarrow (p \rightarrow q)$) · 可以窮舉真值表來證明。

Table - 換質換位命題的真值表

| p | q | $\neg q \rightarrow \neg p$ | $p \rightarrow q$ |
|---|---|-----------------------------|-------------------|
| 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 |

Definition - 換質命題

令 p 與 q 為一個命題

則一條件敘述 $\neg p \rightarrow \neg q$ 稱作換質命題，與換位命題(也就是 $(q \rightarrow p) \leftrightarrow (\neg p \rightarrow \neg q)$)邏輯等價，可以窮舉真值表來證明。

Table - 換質命題的真值表

| p | q | $\neg p \rightarrow \neg q$ |
|-----|-----|-----------------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Definition - 若且為若

令 p 與 q 為一個命題

則一條件敘述 $p \leftrightarrow q$ 稱作若且為若

若 p 與 q 皆 True，或 p 與 q 皆 False，則 $p \leftrightarrow q$ 為 True，否則為 False。

Table - 若為且若的真值表

| p | q | $p \leftrightarrow q$ |
|-----|-----|-----------------------|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

1.2 邏輯命題應用

Example

How can this English sentence be translated into a logical expression?

"You can access the Internet from campus only if you are a computer science major or you are not a freshman."

Let A can access the internet from compus, B are a computer science major, and C are a freshman.

So that the English sentence be translated into $A \rightarrow (B \vee \neg C)$

1.3 命題等價

Introduce - 恒真式

恒真式代表複合命題恒為 true。

Example

$$(p \vee \neg p)$$

無論 p 是True或者False，他都恆為True，稱為tautology(恆真式)

Introduce - 矛盾式

矛盾式代表負和命題恆為false。

Example

$$(p \wedge \neg p)$$

無論 p 是True或者False，他都恆為False，稱為contradiction

Definition - 邏輯等價

令 p 和 q 為複合命題，邏輯等價的定義為 $p \leftrightarrow q$ 為恆等式，寫作 $p \equiv q$

Example

證明 $p \rightarrow q$ 和 $\neg p \vee q$ 為邏輯等價。

我們可以窮舉真值表，來證明兩者為邏輯等價

令 $A = p \rightarrow q$ ， $B = \neg p \vee q$ ，則

| p | q | $A = p \rightarrow q$ | $B = \neg p \vee q$ | $A \leftrightarrow B$ |
|-----|-----|-----------------------|---------------------|-----------------------|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Recall - 德摩根定律

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

Example 2

證明 $\neg(p \vee q)$ 與 $\neg p \wedge \neg q$ 邏輯等價。

設 $A = \neg(p \vee q)$ ， $B = \neg p \wedge \neg q$

我們可以窮舉真值表，來證明兩者為邏輯等價

| p | q | $A = \neg(p \vee q)$ | $B = \neg p \wedge \neg q$ | $A \leftrightarrow B$ |
|-----|-----|----------------------|----------------------------|-----------------------|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |

Example 3

證明 $p \rightarrow q$ 和 $\neg p \vee q$ 邏輯等價。

設 $A = p \rightarrow q$ · $B = \neg p \vee q$

我們可以窮舉真值表，來證明兩者為邏輯等價。

| p | q | $A = p \rightarrow q$ | $B = \neg p \vee q$ | $A \leftrightarrow B$ |
|-----|-----|-----------------------|---------------------|-----------------------|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Example 4

證明 $p \vee (q \wedge r)$ 和 $(p \vee q) \wedge (p \wedge r)$ 邏輯等價。

我們可以窮舉真值表，來證明兩者為邏輯等價。

| p | q | r | $(q \wedge r)$ | $(p \vee q)$ | $(p \wedge r)$ | $p \wedge (q \vee r)$ | $(p \wedge q) \wedge (p \wedge r)$ | $p \wedge (q \vee r) \leftrightarrow (p \wedge q) \wedge (p \wedge r)$ |
|-----|-----|-----|----------------|--------------|----------------|-----------------------|------------------------------------|--|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Recall - 衡等律

$$p \wedge T \equiv p$$

$$p \vee F \equiv p$$

Recall - 支配律

$$p \wedge F \equiv F$$

$$p \vee T \equiv T$$

Recall - 署等律

$$p \wedge p \equiv p$$

$$p \vee p \equiv p$$

Recall - 雙非律

$$\neg(\neg p) \equiv p$$

Recall - 交換律

$$p \wedge q \equiv q \wedge p$$

$$p \vee q \equiv q \vee p$$

Recall - 結合律

$$(p \vee q) \vee r \equiv p \vee (q \vee r)$$

$$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$$

Recall - 分配律

$$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$$

$$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$$

Recall - 吸收律

$$p \vee (p \wedge q) \equiv p$$

$$p \wedge (p \vee q) \equiv p$$

Recall - 否定律

$$p \wedge \neg p \equiv F$$

$$p \vee \neg p \equiv T$$

Recall - 一些有關實質蘊含的邏輯等價

$$p \rightarrow q \equiv \neg p \vee q$$

$$p \rightarrow q \equiv \neg q \rightarrow \neg p$$

$$p \vee q \equiv \neg p \rightarrow q$$

$$p \wedge q \equiv \neg(p \rightarrow \neg q)$$

$$\neg(p \rightarrow q) \equiv p \wedge \neg q$$

$$(p \rightarrow q) \wedge (p \rightarrow r) \equiv p \rightarrow (q \wedge r)$$

$$(p \rightarrow r) \wedge (q \rightarrow r) \equiv (p \vee q) \rightarrow r$$

$$(p \rightarrow q) \wedge (p \rightarrow r) \wedge p \rightarrow (q \vee r)$$

$$(p \rightarrow r) \vee (q \rightarrow r) \equiv (p \wedge q) \rightarrow r$$

Recall - 一些有關若為且若的邏輯等價

$$p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$$

$$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$$

$$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$$

$$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$$

Introduce - 建構新的邏輯等價

如果 p 與 q 邏輯等價，且 q 與 r 邏輯等價，那麼我們就可以說 p 與 r 邏輯等價。

Example 1

證明 $\neg(p \rightarrow q) \equiv p \wedge \neg q$ 邏輯等價

首先， $p \rightarrow q \equiv \neg p \vee q$ ，所以 $\neg(p \rightarrow q) \equiv \neg(\neg p \vee q) \equiv p \wedge \neg q$

因此 $\neg(p \rightarrow q) \equiv p \wedge \neg q$ ，證畢

Example 2

利用一連串的邏輯等價，證明 $\neg(p \vee (\neg p \wedge q)) \equiv \neg p \wedge \neg q$

利用德摩根定律，得到 $\neg(p \vee (\neg p \wedge q)) \equiv \neg p \wedge \neg(\neg p \wedge q)$

再次利用德摩根定律，得到 $\neg p \wedge \neg(\neg p \wedge q) \equiv \neg p \wedge (p \vee \neg q)$

利用分配律， $\neg p \wedge (p \vee \neg q) \equiv (p \wedge \neg p) \vee (\neg p \wedge \neg q) \equiv F \vee (\neg p \wedge \neg q) \equiv (\neg p \wedge \neg q)$

因此， $\neg(p \vee (\neg p \wedge q)) \equiv \neg p \wedge \neg q$ ，證畢

Example 3

證明 $(p \wedge q) \rightarrow (p \vee q)$ 為恆等式

利用 $p \rightarrow q \equiv \neg p \vee q$ 的特性，改寫為 $(p \wedge q) \rightarrow (p \vee q) \equiv \neg(p \wedge q) \vee (p \vee q)$

利用德摩根定律，改寫為 $\neg(p \wedge q) \vee (p \wedge q) \equiv (\neg p \vee \neg q) \vee (p \vee q)$

利用交換律， $(\neg p \vee p) \vee (\neg q \vee q) \equiv T \vee T \equiv T$

故 $(p \wedge q) \rightarrow (p \vee q)$ 為恆等式，證畢。

Introduce - 滿足命題

滿足命題的定義是，若有一個複合命題 P ，若能找到一組變數能夠使 P 為true，則 P 能夠被滿足。

Example

判斷以下三個複合命題 P 是否能夠被滿足。

1. $(p \wedge \neg q) \vee (q \wedge \neg r) \wedge (r \vee \neg p)$
2. $(p \wedge q \wedge r) \vee (\neg p \wedge \neg q \wedge \neg r)$
3. $(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p) \wedge (p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$

第一個命題：若 p, q, r 其中有一個是true，則可以滿足命題。

第二個命題：若 p, q, r 都為true或者都為false，則可以滿足命題。

第三個命題：

考慮到

$(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p)$ 必須要是true，則 p, q, r 都必須要是true，或者 p, q, r 都必須要是false。

$(p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$ 必須要是true，則 p, q, r 都不能是true，或者都不能是false。

因此，兩者矛盾，故 $(p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p) \wedge (p \vee q \vee r) \wedge (\neg p \vee \neg q \vee \neg r)$ 不能被滿足。

1.4 謂語和限定詞

Introduce - 謂語

命題是具有真假意義的陳述句，而陳述句是由主語與謂語所組成的。

例如我們可以說

1. 阿軒是北科大的學生
2. 阿哲不是北科大的學生

則我們假設 $P(x)$ 為： x 是北科大的學生

在這個範例中， $P(x)$ 為謂語， x 為主語，當 x 被賦予值，則 $P(x)$ 則成為一個命題。

因此則上面兩句分別可以寫成 $P(\text{阿軒})$ 與 $P(\text{阿哲})$ ，所得到的結果分別為true與false。

Example 1

若 $P(x)$ 代表 $x > 3$ ，求真值 $P(2)$ 與 $P(4)$

$P(2)$ 等於 $2 > 3$ ，則 $P(2)$ 的真值為false。

$P(4)$ 等於 $4 > 3$ ，則 $P(4)$ 的真值為true。

Example 2

若 $A(x)$ 代表「電腦 x 正在被入侵者攻擊」，且我們假設正在被入侵者攻擊的電腦為 $CS2$ 與 $MATH1$

則求真值 $A(CS1)$ 與 $A(CS2)$ ，以及 $A(MATH1)$

我們可以很清楚的知道， $CS2$ 與 $MATH1$ 正在被攻擊，因此我們可以知道 $A(CS2) = T$ ， $A(MATH1) = T$

而 $CS1$ 沒有被攻擊，故 $A(CS1) = F$

Example 3

若 $Q(x, y)$ 代表敘述 $x = y + 3$ ，則求真值 $Q(1, 2)$ 與 $Q(3, 0)$

$$Q(1, 2) \Rightarrow 1 \neq 2 + 3 \Rightarrow Q(1, 2) = F$$

$$Q(3, 0) \Rightarrow 3 = 0 + 3 \Rightarrow Q(3, 0) = T$$

Example 4

令 $A(c, n)$ 代表「電腦 c 連接著網路 n 」，其中 c 代表電腦而 n 代表著網路。假設 $MATH1$ 正在連接著網路 $CAMPUS2$ 而不是 $CAMPUS1$ 。求 $A(MATH1, CAMPUS1)$ 與 $A(MATH1, CAMPUS2)$ 的真值。

因為 $MATH1$ 沒有連接著網路 $CAMPUS1$ ，因此 $A(MATH1, CAMPUS1)$ 為false

而 $MATH1$ 連接著網路 $CAMPUS2$ 因此 $A(MATH1, CAMPUS2)$ 為true

Example 5

令 $R(x, y, z)$ 代表 $x + y = z$ ，求真值 $R(1, 2, 3)$ 與 $R(0, 0, 1)$

因為 $R(1, 2, 3)$ 代表 $1 + 2 = 3$ ，因此 $R(1, 2, 3)$ 為true。

因為 $R(0, 0, 1)$ 代表 $0 + 0 = 1$ ，因此 $R(0, 0, 1)$ 為false。

Introduce - 量化

量化用來決定一個謂詞，在一定的事物上成立的程度。

產生量化的語言叫作量詞。

舉個例子，「我所有的玻璃都破了」，「大量的人是聰明的」。

通常來說，有兩種量化的類型：全稱量化、存在量化

Definition - 全稱量化

對於 $P(x)$ 來說，全稱量化的代表對於所有在 P 的定義域內的 x ，我們可以寫作 $\forall x P(x)$

\forall 符號代表全域量詞，我們把 $\forall x P(x)$ 讀作"for all x , $P(x)$ " 或者"for every x , $P(x)$ "。

若存在一個 x ，使得 $P(x)$ 為false，則我們把他稱作 $\forall x P(x)$ 的反例。

Table - 量化類別

| 敘述 | 什麼時候為true | 什麼時候為false |
|------------------|--|---|
| $\forall x P(x)$ | 對於所有在 $P(x)$ 定義域中的 x ， $P(x)$ 都為 true。 | 存在一個 x 使得 $P(x)$ 為 false。 |
| $\exists x P(x)$ | 存在一個 x 使得 $P(x)$ 為 true | 對於所有在 $P(x)$ 定義域中的 x ， $P(x)$ 都為 false。 |

Example 1

令 $P(x)$ 代表 $x + 1 > x$ ，對於所有實數域中的 x ， $\forall x P(x)$ 的真值為何

我們可以清楚知道，對於所有實數域中的 x ， $x + 1$ 都大於 x ，找不到一個反例使 $x + 1 < x$ ，因此 $\forall x P(x)$ 的真值為 true。

Example 2

令 $Q(x)$ 代表 $x < 2$ ，對於所有實數域中的 x ， $\forall x P(x)$ 的真值為何

當 $Q(0), Q(1)$ 的時候 $Q(x)$ 為 true，但 $Q(2)$ 為 false，故 $\forall x P(x)$ 為 false。

Example 3

令 $P(x)$ 代表 $x^2 < 10$ ，求所有不超過 4 的正整數中， $\forall x P(x)$ 的真值為何

我們可以把真值表達成 $P(1) \wedge P(2) \wedge P(3) \wedge P(4)$

但是 $P(4)$ 為 false，因為 $4^2 < 10$

所以 $\forall x P(x)$ 為 false。

Example 4

令 $P(x)$ 代表 $x^2 \geq x$ ，若 x 的定義域為所有實數，則 $\forall x P(x)$ 的真值為何？

若 x 的定義域為所有整數，那麼 $\forall x P(x)$ 的真值又為何？

我們可以找到一個反例，若 $x = 0.5$ ，則 $0.5^2 < 0.5$ 。

故若 x 的定義域為所有實數，則 $\forall x P(x)$ 的真值為 false。

但若 x 的定義域為所有整數，我們找不到任何一個反例能夠證明 $\forall x P(x)$ 的真值為 false。

故若 x 的定義域為所有整數， $\forall x P(x)$ 的真值為 true。

Definition - 存在量化

對於 $P(x)$ 來說，存在量化的代表對於至少一個在 P 的定義域內的 x ，我們可以寫作 $\exists x P(x)$ 。 \exists 符號代表存在量詞

若不存在一個 x ，使得 $P(x)$ 為 true，則我們把他稱作 $\exists x P(x)$ 的反例。

Example 1

令 $P(x)$ 代表 $x > 3$ ，若 x 的定義域為所有實數，則 $\exists x P(x)$ 的真值為何？

我們可以找到 $x = 4$ 使得 $x > 3$ 成立，故 $\exists x P(x)$ 的真值為true。

Example 2

令 $Q(x)$ 代表 $x = x + 1$ ，若 x 的定義域為所有實數，則 $\exists x Q(x)$ 的真值為何？

我們找不到任何一個實數，使得 $x = x + 1$ ，故 $\exists x Q(x)$ 為false。

Example 3

令 $P(x)$ 代表 $x^2 > 10$ ，若 x 的定義域為不超過4的正整數，則 $\exists x P(x)$ 的真值為何？

我們可以知道 $x \in \{1, 2, 3, 4\}$

所以我們可以寫成 $P(1) \vee P(2) \vee P(3) \vee P(4)$

由於我們可以知道， $4^2 = 16 > 10$ ，因此 $P(4)$ 為true

故 $P(1) \vee P(2) \vee P(3) \vee P(4)$ 為true

因此 $\exists x P(x)$ 為true

Introduce - 唯一量化

我們可以使用 $\exists! x P(x)$ ，用來表示「唯一一個」、「正好一個」、「剛好一個」 x 使得 $P(x)$ 為真。

例如，令 $P(x)$ 代表 $x - 2 = 4$ ，則 $\exists! x P(x)$ 成立，因為我們可以找到 $x = 6$ ，使得 $P(6)$ 為true

除此之外我們找不到任何的 x ，使得 $P(x)$ 為true，故 $\exists! x P(x)$ 為true。

Introduce - 限定定義域的量詞

通常來說，我們可以透過縮寫，來表示定義域所需要符合的條件。

例如說 $\forall x > 0 (x^2 > x)$ ，且定義域為所有實數，則代表說，對於所有大於0的實數，使得 $x^2 > x$ 。

Example

若下列敘述的定義域皆為 $x \in \mathbb{R}$ ，求 $\forall x < 0 (x^2 > 0)$ ， $\forall y \neq 0 (y^3 \neq 0)$ ，還有 $\exists z > 0 (z^2 = 2)$ 的意義。

第一個例子，若 $x < 0$ ，則 $x^2 > 0$ ，則我們可以寫成 $(x < 0) \rightarrow (x^2 > 0)$

且所有的實數 x 都要符合這個敘述，因此 $\forall x [(x < 0) \rightarrow (x^2 > 0)]$

第二個例子，若 $y \neq 0$ ，則 $y^3 \neq 0$ ，則我們可以寫成 $(y \neq 0) \rightarrow (y^3 \neq 0)$

且所有的實數 y 都要符合這個敘述，因此 $\forall y [(y \neq 0) \rightarrow (y^3 \neq 0)]$

第三個例子，若 $z > 0$ ，則 $z^2 = 2$ ，則我們可以寫成 $(z > 0) \wedge (z^2 = 2)$

且至少一個 z 都要符合這個敘述，因此 $\exists z [(z > 0) \wedge (z^2 = 2)]$

Introduce - 量詞的優先級

\forall 與 \exists 是所有邏輯符號中優先級最高的，也就是若 $\forall xP(x) \vee Q(x)$ ，他代表著 $(\forall xP(x)) \vee Q(x)$ ，而非 $\forall x(P(x) \vee Q(x))$

Introduce - 綑綁變數

如果量詞用在變數 x 上，則我們說變數 x 為一個綑綁變數，否則他就是自由變數。

一個命題函數所有變數都必須為綑綁變數，則這個命題變數才會變成一命題。

無論是存在量化、全稱量化都可以使用

Example 1

在敘述 $\exists x(x + y = 1)$ 中，我們可以知道 x 是綑綁變數，因為前面的存在量詞是對 x 的

但是沒有任何對 y 的量化，因此 y 為一個自由變數。

Example 2

在敘述 $\exists x(P(x) \wedge Q(x)) \vee \forall xR(x)$ ，所有的變數都是綑綁變數

第一個存在量詞對括號內所有的 x 進行綑綁，而第二個存在量詞對命題函數 R 的變數 x 進行綑綁。

Example 3

在敘述 $\exists x(P(x) \wedge Q(x)) \vee \forall yR(y)$ ，所有變數都是綑綁變數

第一個存在量詞對括號內所有的 x 進行綑綁，而第二個存在量詞對命題函數 R 的變數 y 進行綑綁。

Introduce - 關於量化的邏輯等價

Definition - 量化的邏輯等價

關於量詞與謂語的邏輯等價，若兩邊敘述若為且若擁有相同的真值，則我們稱他為邏輯等價。

不用考慮謂語如何替代敘述，或者在命題函數中的定義域為何。

Example

證明 $\forall x(P(x) \wedge Q(x))$ 與 $\forall P(x) \wedge \forall Q(x)$ 邏輯等價。

若要證明 $\forall x(P(x) \wedge Q(x))$ 與 $\forall P(x) \wedge \forall Q(x)$ 邏輯等價

則我們假設 $a \in D(x)$ ，其中 $D(x)$ 代表 x 的定義域，那麼如果 $\forall x(P(x) \wedge Q(x))$ 是true，則 $P(a) \wedge Q(a)$ 都為true。

達成與邏輯為正的條件即為 $P(a)$ 與 $Q(a)$ 皆為true，則與邏輯才會成立。

接著，假設 $\forall xP(x) \wedge \forall xQ(x)$ 為true，且 $a \in D(x)$

那麼為了達成與邏輯的條件， $P(a)$ 應為true， $Q(a)$ 也應為true。

故 $P(a) \wedge Q(a)$ 應為true，而 $a \in D(x)$ ，則代表所有在 $D(x)$ 的變數 a 都可以使得 $P(a) \wedge Q(a)$ 為true

故我們可以把 a 改寫成 $\forall x(P(x) \wedge Q(x))$

因此，我們可以得知 $\forall x(P(x) \wedge Q(x))$ 與 $\forall xP(x) \wedge \forall xQ(x)$ 邏輯等價。

Introduce - 邏輯非的量化表達式

Introduce - 全稱量化的邏輯非

思考以下的敘述

「在這間教室所有人都修過微積分」

若我們利用謂語取代敘述的部分，則我們可以令 $P(x)$ 為「 x 修過微積分」，而 x 的定義域限定為在這間教室的人，故敘述可以表達成 $\forall xP(x)$

而若不是所有人都修過微積分，則必定在教室有一個人 x 使得 $P(x)$ 為 false。

因此，我們可以表達成，若不是所有人都修過微積分，則我們可以用存在量化來表示

也就是 $\exists x \neg P(x)$

因此，我們可以知道 $\neg(\exists x \neg P(x)) \equiv \forall xP(x)$

兩邊邏輯等價同取邏輯非，則 $\neg\neg(\exists x \neg P(x)) \equiv \neg\forall xP(x)$

也就是 $\exists x \neg P(x) \equiv \neg\forall xP(x)$

Introduce - 存在量化的邏輯非

思考以下敘述

「在這間教室，至少有一個人修過微積分」

若我們利用謂語取代敘述的部分，則我們可以令 $P(x)$ 為「 x 修過微積分」，而 x 的定義域限定為在這間教室的人，故敘述可以表達成 $\exists xP(x)$

而若不是至少有一個人修過微積分，則必定在定義域內所有的 x 都能使 $\exists xP(x)$ 為 false。

因此，我們可以表達成，若沒有人修過微積分，則我們可以用全稱量化來表示

也就是 $\forall x \neg P(x)$

因此我們可以知道 $\neg(\forall x \neg P(x)) \equiv \exists xP(x)$

兩邊邏輯等價同取邏輯非，則 $\neg\neg(\forall x \neg P(x)) \equiv \neg\exists xP(x)$

也就是 $\forall x \neg P(x) \equiv \neg\exists xP(x)$

Table - 量化表達式的德摩根定律

| 量化表達式 | 取邏輯非後的量化表達式 |
|-----------------|-----------------------|
| $\forall xP(x)$ | $\exists x \neg P(x)$ |
| $\exists xP(x)$ | $\forall x \neg P(x)$ |

Example

求 $\forall x(x^2 > x)$ 與 $\exists x(x^2 = 2)$ 的反邏輯。

對於所有的 $x \cdot x^2 > x$ 皆成立

而他的反邏輯即為存在一個 x 使得 $x^2 > x$ 不成立

故我們可以寫成存在一個 x 使得 $x^2 \leq x$

因此 $\neg\forall x(x^2 > x) \equiv \exists x(x^2 \leq x)$

存在一個 x 使得 $x^2 = 2$ 成立

而他的反邏輯即為使所有的 x 讓 $x^2 = 2$ 不成立

故我們可以寫成讓所有的 x 使得 $x^2 \neq 2$

因此 $\neg\exists x(x^2 = 2) \equiv \forall x(x^2 \neq 2)$

x 的值取決於定義域。

1.5 嵌套限定詞

Introduce - 嵌套量詞

在1.4的章節，我們通常都只會用到一個量詞，而量詞是可以被嵌套的。

舉個例子，就像這樣： $\forall x \exists y(x + y = 0)$ ，一層一層套上的量詞

而我們也可以改個表達方式，令 $Q(x)$ 為 $\exists y P(x, y)$ ，而 $P(x, y)$ 為 $x + y = 0$

則利用 $\forall x Q(x)$ ，就能表達 $\forall x \exists y(x + y = 0)$

也就能表達在定義域 $D(x)$ 內的所有 x ，都存在一個 $y \in D(y)$ 使得 $x + y = 0$

Introduce - 嵌套量詞的順序

Example 1

若我們考慮 $\forall x \forall y(x + y = y + x)$ ，則他唸起來會像

「對於每一對的 (x, y) ， $\forall x \forall y(x + y = y + x)$ 均成立。」

而若我們變換一下順序，寫作 $\forall y \forall x(x + y = y + x)$ ，則他唸起來會像

「對於每一對的 (y, x) ， $\forall y \forall x(x + y = y + x)$ 均成立。」

因此，我們可以知道，兩個不同的全稱量詞對換是不會影響命題本身的。

Example 2

1. 若我們考慮 $\exists x \forall y(x + y = 0)$ ，且 x, y 的定義域為所有實數

則他唸起來會像，存在一個 x ，使得每一種 y 都能符合 $x + y = 0$

這個命題很明顯是false，因為不存在任何一個 x ，使得每一種 y 都能符合 $x + y = 0$ 。

2. 若我們考慮 $\forall x \exists y(x + y = 0)$ ，且 x, y 的定義域為所有實數

則他唸起來會像，對於每一個屬於實數的 x ，存在一種 y 能符合 $x + y = 0$

這個命題就會是true，因為只要使 $y = -x$ ，就能使敘述成立。

因此，兩者對調是會影響命題本身的。

Example 3

令 $Q(x, y, z)$ 代表敘述 $x + y = z$ 。若 x, y, z 的定義域為所有實數，試求 $\forall x \forall y \exists z Q(x, y, z)$ 和 $\exists z \forall x \forall y Q(x, y, z)$ 的真值。

1. 若我們考慮 $\forall x \forall y \exists z Q(x, y, z)$ ，則他唸起來會像，對於所有的 x 與所有的 y ，存在一個 z ，使得 $x + y = z$ 。這樣是合理的，無論 x 與 y 的值為多少，兩個實數相加必定為另一個實數，故 $\forall x \forall y \exists z Q(x, y, z)$ 的真值為true。
2. 若我們考慮 $\exists z \forall x \forall y Q(x, y, z)$ ，則他唸起來會像，存在一個 z 使得對於所有的 (x, y) ，都能符合 $x + y = z$ 。這是不合理的，因為找不到一種 z ，使得任意的 (x, y) 對符合 $x + y = z$ ，故 $\exists z \forall x \forall y Q(x, y, z)$ 的真值為false。

綜合上述，兩者的量詞互換，會影響到命題的結果。

Table - 兩個嵌套量詞的意義

| <i>Statement</i> | <i>When True?</i> | <i>When False?</i> |
|--|---|--|
| $\forall x \forall y P(x, y)$ $\forall y \forall x P(x, y)$ | $P(x, y)$ is true for every pair x, y . | There is a pair x, y for which $P(x, y)$ is false. |
| $\forall x \exists y P(x, y)$ | For every x there is a y for which $P(x, y)$ is true. | There is an x such that $P(x, y)$ is false for every y . |
| $\exists x \forall y P(x, y)$ | There is an x for which $P(x, y)$ is true for every y . | For every x there is a y for which $P(x, y)$ is false. |
| $\exists x \exists y P(x, y)$ $\exists y \exists x P(x, y)$ | There is a pair x, y for which $P(x, y)$ is true. | $P(x, y)$ is false for every pair x, y . |

Introduce - 嵌套量詞的負邏輯

負邏輯一樣可以用在嵌套量詞上。

Example

請找出 $\forall x \exists y(xy = 1)$ 的負邏輯命題

命題翻譯會變成，對於所有的 x ，存在一個 y 使得 $xy = 1$

那麼對於這個命題加上負邏輯，則變成了存在一個 x ，使得所有可能的 y ，讓 $xy = 1$ 不成立。

故我們可以寫成 $\exists x \forall y \neg(xy = 1)$

而 $\neg(xy = 1)$ 又可以寫成 $(xy \neq 1)$

故整個式子可以寫成 $\exists x \forall y(xy \neq 1)$

1.6 推論規則

Introduce - 肯定前件

若 $P \equiv T$ · 且 $P \rightarrow Q \equiv T$ · 則 $Q \equiv T$

Example

你考到了100分。

若你考到100分，就會得到A。

所以，你會得到A。

Introduce - 否定後件

若 $\neg Q \equiv T$ · 且 $P \rightarrow Q \equiv T$ · 則 $\neg P \equiv T$

Example

你沒有得到A。

若你考到100分，就會得到A。

所以你沒有考到100分。

Introduce - 三段論證

若 $P \rightarrow Q \equiv T$ · 且 $Q \rightarrow R \equiv T$ · 則 $P \rightarrow R \equiv T$

Example

若你難過，就吃東西

若你吃東西，就可能會變胖

所以你難過就可能會變胖。

Introduce - 選言三段論

若 $P \vee Q \equiv T$ · 且 $\neg P \equiv T$ · 則 $Q \equiv T$

Example

我要嘛選擇睡覺，要嘛選擇讀書。

我沒在睡覺。

所以我在讀書。

Introduce - 添加律

若 $P \equiv T$ · 則 $P \vee Q \equiv T$

Example

若我在睡覺。

所以我可能在睡覺或者在讀書。

Introduce - 簡化律

若 $P \wedge Q \equiv T$ ，則 $P \equiv T$

Example

外面是晚上而且外面在下雨

所以外面是晚上。

Introduce - 連言

若 $P \equiv T$ ，且 $Q \equiv T$ ，則 $P \wedge Q \equiv T$

Example

外面是晚上。

外面在下雨。

所以外面是晚上，而且外面在下雨。

Introduce - 預解律

若 $P \vee Q \equiv T$ ，且 $\neg P \vee R \equiv T$ ，則 $Q \vee R \equiv T$

Example

外面是晚上或者外面在下雨。

且外面是白天或者外面在放晴。

則外面在下雨或者外面在放晴。

Introduce - 全稱實例化

若 $\forall x P(x) \equiv T$ ，則若 $c \in D(x)$ ，則 $P(c) \equiv T$

Example

所有資工系的學生都修過微積分

若小碩是資工系的學生

則小碩修過微積分

Introduce - 全稱普遍化

若 $c \in D(x)$ · 且 $P(c)$ for an arbitrary c · 則 $\forall x P(x) \equiv T$

Example

若小碩是資工系的學生

所有資工系的學生都修過微積分

則小碩修過微積分

Introduce - 存在實例化

若 $\exists x P(x) \equiv T$ · 則 $P(c) \equiv T$ for some element c

Introduce - 存在普遍化

若 $P(c) \equiv T$ for some element c · 則 $\exists x P(x) \equiv T$

1.7 Introduce to proof

Introduce - 定理

定理是一個可以被證明的敘述。

Introduce - 公理

公理是一個不需要證明 · 假設正確的敘述。

Example

$\forall x, y \in \mathbb{R} \cdot x + y \in \mathbb{R}$ 是一個公理。

Introduce - 引理

引理是較不重要的敘述 · 用來協助證明其他的結果。

Introduce - 系理

系理是一個定理 · 由另一個定理推導出另一個顯而易見的定理。

Introduce - 猜想/假說

猜想(假說)是一個定理 · 被提出但沒有人能夠證明正確與否。

Example

費馬大定理： $x^n + y^n = z^n$

當 $n \in \mathbb{Z}$ 且 $n > 2$ 時， (x, y, z) 沒有整數解。

Example

試證明，若 $n \in \text{odd}$ ，則 $n^2 \in \text{odd}$

我們可以寫成 $\forall n(P(n) \rightarrow Q(n))$ ， $P(n)$ 代表 n 是奇數，而 $Q(n)$ 代表 n^2 是奇數。

假設 $n \in \text{odd}$ ，則 $n = 2k + 1, k \in \mathbb{Z}$

$$\text{則 } n^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$$

則我們可以知道 $2k^2 + 2k$ 必定是某個整數

由於偶數乘以任何整數均為偶數，故偶數+1必為奇數

故 n^2 必為奇數。

Introduce - 反證法

若 $P \rightarrow Q$ 證明較為困難，則證明與 $P \rightarrow Q$ 等價的 $\neg Q \rightarrow \neg P$ 。

Example

試證明，若 $n \in \mathbb{Z}$ ，且 $3n + 2$ 是奇數，則 n 是奇數。

我們可以寫成 $\forall n(P(n) \rightarrow Q(n))$ ， $P(n)$ 代表 $3n + 2$ 是奇數， $Q(n)$ 代表 n 是奇數。

則我們利用反證法，證明 $\forall n(\neg Q(n) \rightarrow \neg P(n))$ ，也就是證明對於所有的整數 n ，若 n 是偶數，則 $3n + 2$ 是偶數。

則 n 可以寫成 $2k$ 的形式，因此 $3n + 2 = 6k + 2 \cdot k \in \mathbb{Z}$

由於偶數乘以任何整數均為偶數，且偶數加上任何偶數均為偶數

故我們可以證明，若 n 是偶數，則 $3n + 2$ 是偶數。

故若 $n \in \mathbb{Z}$ ，且 $3n + 2$ 是奇數，則 n 是奇數。

Introduce - 空泛證明

若 $P \equiv F$ ，則證明完成。

Example

試證明，如果 $0 > 1$ ，則 $0^2 > 0$

我們可以寫成 $P \rightarrow Q$ ，其中 P 為 $0 > 1$ ，且 Q 為 $0^2 > 0$

但 $P \equiv F$ ，則 Q 不可能發生，證畢。

Introduce - 平庸證明

若 $Q \equiv T$ · 則證明完成。

Example

設 $a, b \in \mathbb{Z}$ · 若 $a \geq b$ · 則 $a^0 \geq b^0$

我們可以寫成 $\forall a \forall b (P(a, b) \rightarrow Q(a, b))$ · 其中 $P(a, b)$ 為 $a \geq b$ · 且 $Q(a, b)$ 為 $a^0 \geq b^0$

則由於無論 a, b 為何 · $a^0 = b^0$ · 故 $\forall a \forall b Q(a, b) \equiv T$ · 故 $\forall a \forall b (P(a, b) \rightarrow Q(a, b)) \equiv T$

Introduce - 歸謬證明法

有兩種用途

| 用途 | 假設 | 矛盾 |
|---------------------------|-------------------------------|-----------------------------------|
| P is true | $\neg P$ is true | $\neg P \rightarrow F$ |
| $P \rightarrow Q$ is true | P is true, $\neg Q$ is true | $(P \wedge \neg Q) \rightarrow F$ |

1.8 Exhaustive Proof

Example 1

$(n + 1)^3 \geq 3 \cdot n \in \mathbb{Z}^+, n \leq 4$

若 $n = 1$ · 則 $2^3 = 8 \geq 3$ ·

若 $n = 2$ · 則 $3^3 = 27 \geq 3$

若 $n = 3$ · 則 $4^3 = 64 \geq 3$

若 $n = 4$ · 則 $5^3 = 125 \geq 3$

故 $(n + 1)^3 \geq 3$ 成立。

Example 2

若 $n \in \mathbb{Z}$ · 則 $n^2 \geq n$

設 $\forall n P(n)$ 代表對於所有實數 n · $n^2 \geq n$

分成三個case 分開做處理。

$n = 0 \cdot 0 = 0 \cdot$ 故 $P(0)$ 成立。

$n \geq 1 \cdot$ 已知 $n \geq 1 \cdot$ 則兩邊同乘以 $n^2 \geq n \cdot$ 故 $n^2 \geq n$ 成立。

$n \leq 1 \cdot$ 已知 $n^2 \geq 0 \cdot$ 故 $n^2 \geq n$

故 $\forall n P(n)$ 成立

Introduce - 建構式證明

建構式證明可以分成存在證明或不存在證明。

Example 1

請證明可以找到一個整數，這個整數可以用兩個以上不同的立方和所組成。

我們可以找到一個數字 1729 ，且 $1729 = 10^3 + 9^3$ 且 $1729 = 12^3 + 1^3$ ，故證明完畢。

Example 2

請證明可以找到一個無理數 x 與 y ，使得 x^y 為有理數。

分成兩個case

若 $\sqrt{2}^{\sqrt{2}}$ 是有理數，則 $x = \sqrt{2}, y = \sqrt{2}$

若 $\sqrt{2}^{\sqrt{2}}$ 是無理數，則 $x = \sqrt{2}^{\sqrt{2}}, y = \sqrt{2} \cdot x^y = \sqrt{2}^{\sqrt{2}\sqrt{2}} = \sqrt{2}^2 = 2$

則這兩者其中之一可以符合命題。

Introduce - 存在證明

存在一個 x ，使得 $P(x)$ 為 true，且除了 x 以外的 y ，使得 $P(y)$ 為 false。

$$\exists x(P(x) \wedge \forall y(y \neq x \rightarrow \neg P(y)))$$

Introduce - 反例

Example

試證明，每個正整數，是三個整數的平方和。

若正整數為 1 ，則我們找不到任意一組 $(x, y, z) \in \mathbb{Z}$ ，使得 $x^2 + y^2 + z^2 = 1$ ，故假說錯誤。

2. 基礎結構: 集合、函數、序列、總和、與矩陣

2.1 集合

Definition - 集合

集合是一個不需按照順序排列，且集合內部所有元素均不相等的物件。

$a \in A$, a 是集合 A 的元素之一。

$a \notin A$, a 不是集合 A 的元素之一。

Example

$$\{a, b\} = \{b, a\}$$

$$\{a, a, b\} = \{a, b\}$$

$$(a, b) \neq (b, a)$$

Introduce - 窮舉法

如果窮舉出集合內的所有物件是可能的，我們可以窮舉出集合內的所有物件。

Example 1

所有母音的集合。

$$V = \{a, e, i, o, u\}$$

Example 2

所有小於100的正整數的集合。

$$O = \{1, 2, 3, \dots, 99, 100\}$$

Introduce - 集合建構式符號

$\{x \mid x \text{ has property } P\}$ · 念作「所有符合條件 P 元素 x 的集合」。

Example 1

所有小於10的正整數奇數集合。

$$O = \{1, 3, 5, 7, 9, \dots\}$$

$$O = \{x \mid x \text{ is an odd positive integers less than } 10\}$$

$$O = \{2x + 1 \mid 0 \leq x \leq 4\}$$

Example 2

所有正的有理數，且為整數的集合 \mathbb{Q}^+ 。

$$\mathbb{Q}^+ = \{x \in \mathbb{R} \mid x = \frac{a}{b} \text{ for some positive integers } p \text{ and } q.\}$$

Example 3

所有自然數的整數集合。

$$\mathbb{N} = \{0, 1, 2, 3, \dots\}$$

Example 4

所有整數的集合。

$$\mathbb{Z} = \{\dots, -2, -1, 0, 1, 2, \dots\}$$

Example 5

所有正整數的集合。

$$\mathbb{Z}^+ = \{0, 1, 2, \dots\}$$

Example 6

所有有理數的集合。

$$\mathbb{Q} = \left\{ \frac{p}{q} \mid p \in \mathbb{Z}, q \in \mathbb{Z}, q \neq 0 \right\}$$

Definition - 集合相等

若兩個集合的所有元素相等，則我們說這兩個集合相等。

$$A = B \iff (x \in A \rightarrow x \in B)$$

Example

$$\{1, 3, 5\} = \{3, 5, 1\}$$

$$\{1, 3, 3, 3, 5, 5, 5, 5\} = \{1, 3, 5\}$$

Definition - 空集合

空集合沒有任何元素，寫作 \emptyset 。

Definition - 單元素集合

單元素集合只有一個元素。

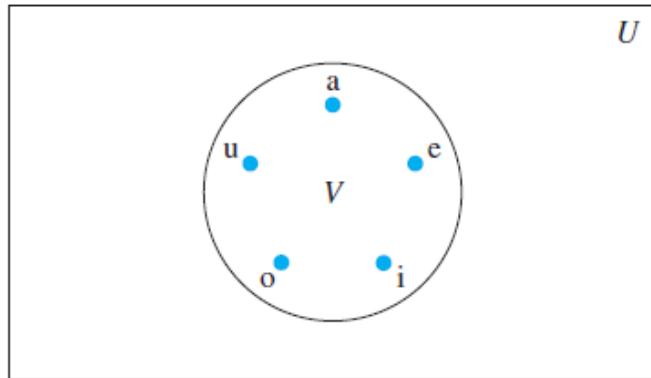
Example

$\{\emptyset\}$ 是一個單元素集合。

Introduce - 文氏圖

我們可以用文氏圖來表示一個集合。

Example



$$U = \{a, b, c, d, e, f, g, \dots\}$$

$$V = \{a, e, i, o, u\}$$

Definition - 子集合

若為且若集合 A 的每個元素都為集合 B 的元素，則我們說 A 是 B 的子集合，且 B 為 A 的父集合。

$$\text{可以表示成 } (A \subseteq B \wedge B \supseteq A) \iff \forall x(x \in A \rightarrow x \in B)$$

Theorem - 1

對於每一個集合 S , $\emptyset \subseteq S$, $S \subseteq S$

$$\{\} \subseteq \{\}$$

$$\{\} \subseteq \{a\}$$

$$\{a\} \subseteq \{a\}$$

$$\{\} \subseteq \{a, b\}$$

$$\{a\} \subseteq \{a, b\}$$

$$\{b\} \subseteq \{a, b\}$$

$$\{a, b\} \subseteq \{a, b\}$$

如果一個集合有 n 個元素，則他會有 $n!$ 種不同的子集合。

Introduce - 真子集

若 A 是 B 的真子集，則對於所有在集合 A 的 x 也都在集合 B ，且 B 存在一個元素不在集合 A 。

$$\text{可以寫作 } A \subset B \iff \forall x(x \in A \rightarrow x \in B) \wedge \exists x(x \in B \wedge x \notin A)。$$

Definition - 有限集合與無限集合

如果一個集合有剛好 n 個元素，且 n 存在，則我們說這個集合是有限的，否則這個集合是無限的。

Definition - 集合的勢

若有一個集合 A ，我們定義「集合的勢」為集合內部的元素數量，寫作 $|A|$ 。

Example 1

$$|\emptyset| = 0$$

Example 2

令集合 S 為擁有所有字母的集合，則 $|S| = 26$

Example 3

$$|\{1, 2, 3\}| = 3$$

Example 4

$$|\{\emptyset\}| = 1$$

Example 5

若 S 為所有整數的集合，則 $|S|$ 為無限。

Introduce - 罩集

若存在一個集合有集合 A 的所有子集，我們寫作 $\wp(A)$ 。

如果集合 A 有 N 個元素，則 $|\wp(A)| = 2^N$ 。

Example

若 $A = \{a, b\}$ ，則 $\wp(A) = \{\{\emptyset\}, \{a\}, \{b\}, \{a, b\}\}$

Introduce - 多元組

- 一個有序且長度為 n 的多元組包含了元素 $(a_1, a_2, a_3, \dots, a_n)$ ，且 a_1 是第一個元素， a_n 是最後一個元素。
- 兩個長度為 n 的多元組 A, B ，我們先用 A_i 表示多元組 A 的第 i 個元素。
若兩個多元組的每一項都相同，也就是 $A_1 = B_1, A_2 = B_2, \dots, A_n = B_n$ ，則兩個多元組就是相同的。
- 長度為2的多元組我們稱作有序對。
- 若有兩個有序對 (a, b) 與 (c, d) ，則若 $a = c$ 且 $b = d$ ，則兩個有序對才相同。

Example

若有兩個長度為 n 的多元組 A, B 。

$$A = (1, 2, 3, 4, 5) \cdot B = (5, 4, 3, 2, 1) \cdot \text{則 } A \neq B$$

$$A = (1, 2, 3, 4, 5) \cdot B = (1, 2, 3, 4) \cdot \text{則 } A \neq B$$

$$A = (1, 2, 3, 4, 5) \cdot B = (1, 2, 3, 4, 5) \cdot \text{則 } A = B$$

Introduce - 笛卡兒積

兩個集合 A, B 相乘，我們稱作笛卡爾積，寫作 $A \times B$ 。

$A \times B$ 是一個集合，包含了所有不同的有序對 (a, b) ，其中 $a \in A \wedge b \in B$

我們可以寫成這樣： $A \times B = \{(a, b) | a \in A \wedge b \in B\}$

Example

若集合 $A = \{a, b\}$ ，集合 $B = \{1, 2\}$

則 $A \times B = \{(a, 1), (a, 2), (b, 1), (b, 2)\}$

Introduce - 笛卡兒積的子集合

在笛卡爾積的子集合 R ，我們可以說與集合 A 和集合 B 都有關係。

Introduce - 多個集合的笛卡兒積

若我們有 m 個集合 $A_1, A_2, A_3, \dots, A_M$ ，則笛卡爾積寫作 $A_1 \times A_2 \times \dots \times A_M$ 。

則 $A_1 \times A_2 \times \dots \times A_M = (a_1, a_2, \dots, a_n)$ 為一個有序多元組的集合，其中 $a_i \in A_i, i = 1, 2, \dots, n$ 。

Example

若 $A = \{0, 1\}$ ， $B = \{1, 2\}$ ， $C = \{0, 1, 2\}$ ，求 $A \times B \times C$

$$A \times B \times C = \{(0, 1, 0), (0, 1, 1), (0, 1, 2), (0, 2, 0), (0, 2, 1), (0, 2, 2), (1, 1, 0), (1, 1, 1), (1, 1, 2), (1, 2, 0), (1, 2, 1), (1, 2, 2)\}$$

Introduce - 量詞的真值集

給定一個量詞 P 與定義域 D ，我們定義量詞 P 的真值集為所有使得 P 為true且在定義域 D 的元素。

我們可以把真值集寫作 $\{x \in D | P(x)\}$ 。

Example

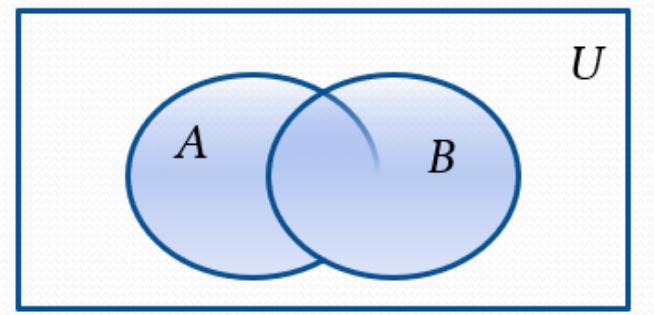
給定定義域 D 為所有整數與 $P(x)$ 為 $|x| = 1$ ，找出 $P(x)$ 的真值集。

則 $P(x)$ 的真值集為 $\{-1, 1\}$ 。

2.2 集合運算子

Introduce - 聯集

A 與 B 為集合，若 A 與 B 取聯集，則我們可以表示成 $A \cup B = \{x | x \in A \vee x \in B\}$ 。

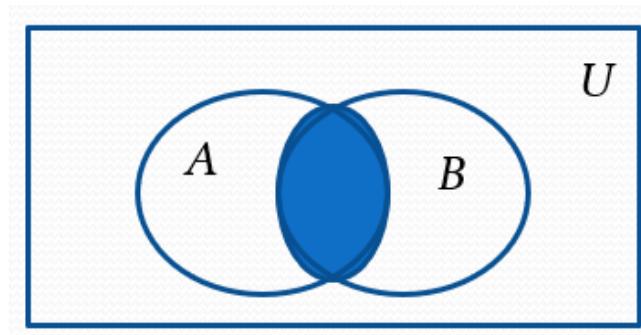


Example

若 $A = \{1, 2, 3\}$ · 且 $B = \{3, 4, 5\}$ · 則 $A \cup B = \{1, 2, 3, 4, 5\}$

Introduce - 交集

A 與 B 為集合 · 若 A 與 B 取交集 · 則我們可以表示成 $A \cap B = \{x | x \in A \wedge x \in B\}$



如果 $A \cap B = \emptyset$ · 則 A 與 B 的關係為互斥的。

Example 1

若 $A = \{1, 2, 3\}$ · 且 $B = \{3, 4, 5\}$ · 則 $A \cap B = \{3\}$

Example 2

若 $A = \{1, 2, 3\}$ · 且 $B = \{4, 5, 6\}$ · 則 $A \cap B = \emptyset$

Introduce - 補集

令 A 是一個集合 · 則 A 的補集(通常叫做宇集 U) · 寫作 \overline{A} · 為 $U - \overline{A}$ 的集合。

定義為 $\overline{A} = \{x \in U | x \notin A\}$

A 的補集有時表示成 A^c 。

Example

如果字集 U 代表小於100的正整數，則求 $\{x|x > 70\}$ 的補集。

$$\{x|x \leq 70\}$$

####

Introduce - 差集

令 A 與 B 為一個集合。

A 與 B 的差集，可以表示成 $A - B$ ，代表集合 A 不包含集合 B 的東西。

可以被定義為 $A - B = \{x|x \in A \wedge x \notin B\} = A \cap \overline{B}$

Example

令 $A = \{1, 2, 3\}$ ， $B = \{3, 4, 5\}$ ，求 $A - B$

$$A - B = \{1, 2\}$$

Introduce - 兩個集合交集的勢

利用排容原理。

$$|A \cup B| = |A| + |B| - |A \cap B|$$

Example

令 $A = \{1, 2, 3\}$ ， $B = \{3, 4, 5\}$ ，求 $|A \cup B|$

$$|A \cup B| = |A| + |B| - |A \cap B| = |3| + |3| - |5| = 1$$

Introduce - 對稱差

若有兩個集合 A 與 B ，則 A 與 B 的對稱差寫作 $A \oplus B$ 。

定義為 $A \oplus B = (A - B) \cup (B - A)$

Example

若 $U = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ ， $A = \{1, 2, 3, 4, 5\}$ ， $B = \{3, 4, 5, 6, 7\}$ ，求 $A \oplus B$

$$A \oplus B = (A - B) \cup (B - A) = \{1, 2\} \cup \{6, 7\} = \{1, 2, 6, 7\}$$

Introduce - 集合特徵

- 恒等律
 - $A \cup \emptyset = A \cdot A \cap U = A$
- 支配律
 - $A \cup U = U \cdot A \cap \emptyset = \emptyset$
- 罢等律
 - $A \cup A = A \cdot A \cap A = A$
- 補餘律
 - $\overline{(\overline{A})} = A$
- 交換律
 - $A \cup B = B \cup A \cdot A \cap B = B \cap A$
- 連鎖律
 - $A \cup (B \cup C) = (A \cup B) \cup C$
 - $A \cap (B \cap C) = (A \cap B) \cap C$
- 分配律
 - $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
 - $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
- 德摩根定律
 - $\overline{A \cup B} = \overline{A} \cap \overline{B} \cdot \overline{A \cap B} = \overline{A} \cup \overline{B}$
- 吸收律
 - $A \cap (A \cup B) = A \cdot A \cap (A \cup B) = A$
- Complement laws
 - $A \cup \overline{A} = U \cdot A \cap \overline{A} = \emptyset$

2.3 函數

Definition - 函數

令 A 與 B 為一個非空集合，一個函數從 A 映射 B ，寫作 $A \rightarrow B$ 。

代表每一個集合 A 的元素都剛好指向一個集合 B 的元素，寫作 $f(a) = b$ 。

其中 b 為集合 B 的相異元素，被集合 A 的元素所映射。

Introduce - 笛卡耳積的函數

一個 $A \rightarrow B$ ，可以用來表示 $A \times B$ 的子集合，寫作

$$\forall x(x \in A \rightarrow \exists y(y \in B \wedge (x, y) \in f))$$

以及

$$\forall x, y_1, y_2 [(x, y_1) \in f \wedge (x, y_2) \in f] \rightarrow y_1 = y_2$$

Introduce - 映射、像與原像

給你一個集合 A 與集合 B ，我們說 f 是由 A 映射 B 所組成，則

A 被稱為 f 的定義域

B 被稱為 f 的值域

如果 $f(a) = b$ ，則 b 被稱為 f 在 a 的像， a 被稱為 b 的像原

當兩個函數有相同的定義域，相同的域值，還有兩個函數的像與像原映射相同，則兩個函數相同。

Introduce - 單射

函數 f 被稱做一對一函數，或者稱做單射，也就是對於所有在定義域的 a, b ，若為且若 $f(a) = f(b)$ ，則 $a = b$ 。

函數 f 如果是一對一函數，則這個函數是個單射函數。

Introduce - 滿射

若有兩集合 A, B ，若為且若所有元素 $b \in B$ ，存在一個 $a \in A$ ，使得 $f(a) = b$ ，則稱做這個函數為滿射函數。

Introduce - 對射

若一個函數是一對一函數，且函數滿射，則我們稱作這個函數是一對一對應函數或叫做對射函數。

Introduce - 反函數

令 f 是一個集合 A 對集合 B 的對射函數， f 的反函數寫作 f^{-1} 。

反函數 f^{-1} 代表集合 B 對集合 A 的函數，定義為若為且若 $f^{-1}(y) = x$ 則 $f(x) = y$ 。

Introduce - 複合函數

令 f 為集合 B 對集合 C 的函數，且 g 為集合 A 對集合 B 的函數， f 與 g 的複合函數，寫作 $f \circ g$ ，代表一個集合 A 對集合 C 的函數，定義為 $(f \circ g)(x) = f(g(x))$ 。

Introduce - 函數的圖形

令 f 為一個集合 A 對集合 B 函數，函數 f 的圖形即為每一對的 (a, b) ，即為

$$\{(a, b) | a \in A \wedge f(a) = b\}$$

Introduce - 一些重要的函數

底函數代表將小於等於 x 之最大整數指派給實數 x ，記為 $\lfloor x \rfloor$

頂函數代表將大於等於 x 之最小整數指派給實數 x ，記為 $\lceil x \rceil$

階乘函數 $f : \mathbb{N} \rightarrow \mathbb{Z}^+$ ，記為 $f(n) = n! = 1 \times 2 \times \dots \times n$ ，其中 $n \in \mathbb{Z}^+$ 。

Table - 頂函數與升函數

TABLE 1 Useful Properties of the Floor and Ceiling Functions.

(n is an integer, x is a real number)

(1a) $\lfloor x \rfloor = n$ if and only if $n \leq x < n + 1$

(1b) $\lceil x \rceil = n$ if and only if $n - 1 < x \leq n$

(1c) $\lfloor x \rfloor = n$ if and only if $x - 1 < n \leq x$

(1d) $\lceil x \rceil = n$ if and only if $x \leq n < x + 1$

(2) $x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1$

(3a) $\lfloor -x \rfloor = -\lceil x \rceil$

(3b) $\lceil -x \rceil = -\lfloor x \rfloor$

(4a) $\lfloor x + n \rfloor = \lfloor x \rfloor + n$

(4b) $\lceil x + n \rceil = \lceil x \rceil + n$

Introduce - 部分函數

令 f 為一個集合 A 對集合 B 函數。

若 f 的定義域 $D(f) \subseteq A$ · 且 f 的值域 $R(f) \subseteq B$ · 則我們稱 f 為一部分函數。

2.4 數列與總和

Definition - 序列

- 序列是有序的表列 · 例如 $1, 3, 5, 7, 9$ · 或者 $1, 4, 9, 16, 25$ ·
- 序列是一個整數子集的函數。
- 我們使用符號 $\{a_n\}$ 來描述序列。

Example

考慮序列 a_n · 其中 $a_n = \frac{1}{n}$

由 a_1 開始 · 可記為 $a_1, a_2, a_3, a_4, \dots$

前幾項為 $1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \dots$

Introduce - 幾何數列

一個幾何數列可以表示成 : $a, ar, ar^2, ar^3, \dots, ar^n$ · 其中 a 為首項 · r 為公比。

Example

若有一序列 $a_n = (-1)^n$ ，則我們說這是一個幾何數列，因為 a_n 的前幾項為 $1, -1, 1, -1 \dots$ ，即為 $a = 1 \cdot r = -1$ 。

Introduce - 算術數列

一個算術數列可以表示成： $a, a + d, a + 2d, a + 3d, \dots, a + nd$ ，其中 a 為首項， d 為公差

Example

若有一序列 $a_n = 1 + 3n$ ，則我們說這是一個算術數列，因為 a_n 的前幾項為 $1, 4, 7, 10 \dots$ ，即為 $a = 1 \cdot r = 3$ 。

Definition - 字串

- 字串為有限字元數所組成的序列。
- 一個不含任何項的字串稱作空字串。

Introduce - 遞迴關係式

遞迴關係，也就是差分方程式，是一種遞推地定義一個序列的方程式：序列的每一項目是定義為前一項的函數。

Example - 斐波納契數列

一個Fibonacci Sequence可以由以下的性質定義

$$F_0 = 0$$

$$F_1 = 1$$

$$F_n = F_{n-2} + F_{n-1}$$

因此，斐波納契數列的序列為

$$\{0, 1, 1, 2, 3, 5, 8, 13, 21 \dots\}$$

2.5 集合的勢

Definition - 勢

- 令A與B為兩個集合，若為且若A與B對射，則集合A與集合B的勢是相同的，寫作 $|A| = |B|$ 。
- 令A與B為兩個集合，若為且若A與B單射，則集合A的勢小於等於集合B的勢，寫作 $|A| \leq |B|$ 。
如果集合A的勢與集合B的勢不同，則 $|A| < |B|$ 。
- 一個集合如果是有限的，或者與正整數集合的勢相同，則我們說這個集合是可數的，否則就是不可數的。
例如一個實數集合是個不可數的集合。
- 如果一個無限的集合是可數的，他的勢為 \aleph_0 ，寫作 $|S| = \aleph_0$ ，且唸做"aleph null"。
- 一個無限集合的勢，若為且若可以將所有的元素列成一個數列，則我們說這個無限集合的勢是可數的。

原因是一個無限集合的勢，可以寫作一個對射函數，且函數的index是正整數，故我們可以寫成這樣的形式：

$$f(1) = a_1, f(2) = a_2, \dots, f(n) = a_n$$

故根據「一個集合如果是有限的，或者與正整數集合的勢相同，則我們說這個集合是可數的」，我們可以知道無限集合的勢，若為且若可以將所有的元素列成一個數列，則我們說這個集合是無限的。

Introduce - 希爾伯特旅館悖論

一個有無限間房間的旅館，每一個房間均住滿人，我們要怎麼樣能夠再容納一個旅客？

假設我們有無限多個客人，我們將每個客人 j 編號成 a_j ，新的客人我們表示成 a_0 ，則我們可以寫成這樣的形式

$$f(1) = a_1, f(2) = a_2, f(3) = a_3, f(4) = a_4, \dots, f(n) = a_n, \dots$$

我們可以將第 i 房的人，請他移駕至第 $i + 1$ 房，這樣就會使第一間房間空出來。

$$g(1) = a_0, g(2) = a_1, g(3) = a_2, g(4) = a_3, \dots, g(n) = a_{n-1}, \dots$$

可以顯而易見的知道這樣分配不會使房間編號撞號。

Example - 證明集合是可數的(1)

證明正偶數整數集合 E 是可數的。

令 $f(x) = 2x$ ，則我們可以將其列成序列，也就是

$$f(1) = 2, f(2) = 4, f(3) = 6, f(4) = 8, \dots, f(n) = 2n$$

我們可以證明這個函數是對射，假設 t 為偶正整數，則它可以寫成 $2k$ 的形式，故每一個 t 存在唯一一個 k ，使得 $f(k) = t$

我們可以證明這個函數是一對一函數，假設存在 $f(n) = f(m)$ ，必定存在 $2n = 2m$ ，也就是存在 $n = m$

故若 $n = m$ ，才能使得 $f(n) = f(m)$

根據勢的定義，「一個無限集合的勢，若為且若可以將所有的元素列成一個數列，則我們說這個無限集合的勢是可數的。」

因此 $f(x) = 2x$ 是可數的。

Example - 證明集合是可數的(2)

證明整數集合 Z 是可數的。

將集合 Z 列成： $0, 1, -1, 2, -2, 3, -3, \dots$

我們可以寫成以下的部分函數： $f(x) = \begin{cases} f(x) = -(x - 1)/2 & x \in \text{odd} \\ f(x) = x/2 & x \in \text{even} \end{cases}$

4. 數論與密碼學

4.1 可除性與模算術

Definition - 除法

- 如果 a 與 b 是整數，且 $a \neq 0$ ，則我們說 b 能夠被 a 整除，如果存在一個整數 c ，使得 $b = ac$ 。
- 如果 b 能夠被 a 整除，則我們說 a 是 b 的因數或者被除數，且 b 為 a 的倍數。
- 我們用 $a|b$ 來表示 b 能夠被 a 整除。
- 如果 $a|b$ ，則 b/a 為一個整數。
- 如果 a 沒辦法整除 b ，則我們表示為 $a \nmid b$

Theorem 1

令 a, b, c 為整數，且 $a \neq 0$ ，則如果 $a|b$ 和 $a|c$ ，則 $a|(b + c)$

Proof

如果 $a|b$ ，則我們可以說存在一個整數 s ，使得 $as = b$

如果 $a|c$ ，則我們可以說存在一個整數 t ，使得 $at = c$

因此 $b + c = as + at = a(s + t)$ ，因為 $s + t$ 為整數，由此可證如果 $a|b$ 和 $a|c$ ，則 $a|(b + c)$

Theorem 2

令 a, b, c 為整數，且 $a \neq 0$ ，則如果 $a|b$ ，那麼對於任意 c ，符合 $a|bc$

Proof

如果 $a|b$ ，則我們可以說存在一個整數 s ，使得 $as = b$

將等號兩邊同乘以 c ，得到 $asc = bc$

由於兩個整數相乘為整數，故我們依然可以找到一個整數 $d = sc$ ，使得 $ad = bc$

由此可證如果 $a|b$ ，那麼對於任意 c ，符合 $a|bc$ 。

Theorem 3

令 a, b, c 為整數，且 $a \neq 0$ ，則如果 $a|b$ 且 $b|c$ ，那麼 $a|c$

Proof

如果 $a|b$ ，則我們可以說存在一個整數 s ，使得 $as = b$

如果 $b|c$ ，則我們可以說存在一個整數 t ，使得 $bt = c$ ，也就是 $ast = c$

由於兩個整數相乘為整數，故我們依然可以找到一個整數 $d = st$ ，使得 $ad = c$

由此可證如果如果 $a|b$ 且 $b|c$ ，那麼 $a|c$ 。

Corollary 1

如果 a, b, c 為整數，且 $a \neq 0$ ， $a|b$ 和 $a|c$ ，那麼 $a|mb + nc$ ，其中 $n, m \in \mathbb{Z}$

Introduce - 除法算法

如果 $a \in \mathbb{Z}$ · 且 $d \in \mathbb{Z}^+$ · 那麼存在唯一 $q, r \in \mathbb{Z}$ · 其中 $0 \leq r < d$ · 使得 $a = dq + r$ ·

其中 a 被稱為被除數 · d 被稱為除數 · q 被稱做商 · r 被稱做餘數。

我們可以用 mod 與 div 來取得商與餘數 · 定義

$$q = a \text{ div } d$$

$$r = a \text{ mod } d$$

Introduce - 同餘關係

如果 a, b 為整數 · 且 m 為一正整數 · 則如果 m 可以整除 $a - b$ · 我們說 a 與 b 同餘模 m · 可以用 $a \equiv b \pmod{m}$ 來表示。

兩個整數 a, b 能夠同餘模 m · 若為且若 a 除以 m 與 b 除以 m 的餘數相同。

如果 a 除以 m 與 b 除以 m 的餘數不同 · 那麼我們表示為 $a \not\equiv b \pmod{m}$

Theorem 4

令 m 為一正整數 · 整數 a, b 能夠同餘於 m · 若為且若存在一個 k 使得 $a = b + km$

Proof

若 $a \equiv b \pmod{m}$ · 那麼 $m | (a - b)$

因此 · 會存在一個 k · 使得 $mk = a - b$ · 因此 $a = mk + b$

反之 · 若存在一個 k 使得 $a = b + km$ · 那麼 $km = a - b$ · 因此可以得知 $m | (a - b)$ 和 $a \equiv b \pmod{m}$

Introduce - 兩個不同的表示法($\text{mod } m$) 與 $\text{mod } m$

- $a \equiv b \pmod{m}$ 與 $a \text{ mod } m = b$ 是不同的東西
 - $a \text{ mod } m = b$, 代表函數的關係。
 - $a \equiv b \pmod{m}$, 代表一個整數集合的關係。

Theorem 5

令 a, b 為整數 · 令 m 為正整數 · 則 $a \equiv b \pmod{m}$ 若為且若 $a \text{ mod } m = b \text{ mod } m$ 。

Proof

如果 $a \equiv b \pmod{m}$ · 則我們可以說 $a - b = me, e \in \mathbb{Z}$ · 因此 $a = me + b$

那麼令 $d = b \text{ mod } m$ · 我們可以表示成 $b = mr + d, r \in \mathbb{Z}, 0 \leq d < m$

因此 $a = me + mr + d = m(e + r) + d$

我們可以說 $(e + r)$ 是 a/m 的商 · d 為 a/m 的餘。

故 $a \text{ mod } m = d = b \text{ mod } m$

Theorem 6

令 m 為正整數 · 如果 $a \equiv b \pmod{m}$ 和 $c \equiv d \pmod{m}$ · 那麼 $(a + c) \equiv (b + d) \pmod{m}$ 且 $ac \equiv bd \pmod{m}$

Proof

如果 $a \equiv b \pmod{m}$, $c \equiv d \pmod{m}$ · 那麼存在 $s, t \in \mathbb{Z}$ · 使得 $a = sm + b$ · 且 $c = tm + d$

故 $a + c = (s + t)m + (b + d)$ · 且 $ac = stm^2 + sdm + tbm + bd = m(stm + sd + tb) + bd$

因此 $(a + c) \equiv (b + d) \pmod{m}$ · $ac \equiv bd \pmod{m}$

Introduce - 同餘的代數運算

- 令 a, b 為整數 · 若 $a \equiv b \pmod{m}$ · 則 $ac \equiv bc \pmod{m}$
※可以根據Theorem 6 · 令 $d = c$ 。
- 令 a, b 為整數 · 若 $a \equiv b \pmod{m}$ · 則 $a + c \equiv b + c \pmod{m}$
※可以根據Theorem 6 · 令 $d = c$ 。
- 除法並不保證能夠用在同餘上。

Corollary 2

令 m 為正整數 · 令 a, b 為整數 · 則

$$(a + b) \pmod{m} = ((a \pmod{m}) + (b \pmod{m})) \pmod{m}$$

$$(ab) \pmod{m} = ((a \pmod{m})(b \pmod{m})) \pmod{m}$$

Proof

根據模的定義 · 則 $a \equiv (a \pmod{m}) \pmod{m}$ · $b \equiv (b \pmod{m}) \pmod{m}$

By Theorem 6 · 可知 $a + b \equiv (a \pmod{m}) + (b \pmod{m}) \pmod{m}$

以及 $ab \equiv (a \pmod{m})(b \pmod{m}) \pmod{m}$

Definition - 模的算術運算元

令 \mathbb{Z}_m 為所有小於 m 的非負整數集合 · 則

- $a +_m b$ · 用來表示 $(a + b) \pmod{m}$
- $a \cdot_m b$ · 用來表示 $(ab) \pmod{m}$
- 模的算術運算元有許多性質
 - 封閉性：如果 $a, b \in \mathbb{Z}_m$ · 則 $a +_m b \in \mathbb{Z}_m$ 和 $a \cdot_m b \in \mathbb{Z}_m$ 。
 - 結合律：如果 $a, b, c \in \mathbb{Z}_m$ · 則 $(a +_m b) +_m c = a +_m (b +_m c)$ · 且 $(a \cdot_m b) \cdot_m c = a \cdot_m (b \cdot_m c)$
 - 交換律：如果 $a, b \in \mathbb{Z}_m$ · 則 $a +_m b = b +_m a$ · 且 $a \cdot_m b = b \cdot_m a$
 - 單位元素：如果 $a \in \mathbb{Z}_m$ · 則 $a +_m 0 = a$ · 且 $a \cdot_m 1 = a$
 - 加法反元素：如果 $a \neq 0$ 且 $a \in \mathbb{Z}_m$ · 則 $a +_m (m - a) = 0$ · 且 $0 +_m 0 = 0$
 - 分配律：如果 $a, b, c \in \mathbb{Z}_m$ · 那麼 $a \cdot_m (b +_m c) = (a \cdot_m b) +_m (a \cdot_m c)$ ·
 $(a +_m b) \cdot_m c = (a \cdot_m c) + (b \cdot_m c)$

4.2 整數表示與演算法

Theorem 1

令 b 為一個大於1的正整數，如果 n 為一正整數，則他可以表示成 $n = a_k b^k + a_{k-1} b^{k-1} + \dots + a_1 b + a_0 b^0$

其中 k 為非負整數， $a_0, a_1 \dots$ 為一非負整數，且小於 b ，且 $a_k \neq 0$ 。

$a_j \cdot j = 0, 1, \dots, k$ 為以 b 為底數的各個位數。

4.3 質數與最大公因數

Definition - 質數

一個正整數 $p > 1$ ，若 p 的因數只有它自己與1，則我們說這個數字 p 是質數。

若 $p > 1$ ，且 p 的因數不只有它自己與1即為合數。

Theorem 1

任何大於1的正整數都可以用質數的乘積來分解，且分解的結果為唯一。

且結果我們通常會用非遞減的形式呈現。

Example

$$100 = 2 \times 2 \times 5 \times 5 = 2^2 \times 5^2$$

$$641 = 641$$

$$999 = 3 \times 3 \times 3 \times 37 = 3^3 \times 37$$

$$1024 = 2 \times 2 = 2^{10}$$

Introduce - 厄拉托西尼篩法

厄拉托西尼篩法可以在一定的範圍內找到所有質數，舉個例子，我們可以用厄拉托西尼篩法找1~100的所有質數。

1. 刪除所有2的倍數的數字，除了2。
2. 刪除所有3的倍數的數字，除了3。
3. 刪除所有5的倍數的數字，除了5。
4. 刪除所有7的倍數的數字，除了7。
5. 這樣一來，所有剩下的數字都不能被2、3、5、7給整除，因此質數為
 $\{2, 3, 5, 7, 11, 15, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97\}$

Theorem 2

質數的個數是無限的。

Proof

令 $P = p_1 p_2 p_3 p_4 \dots p_n$ · 且 $q = P + 1$ · 則 q 是質數或不是質數 · 兩者必居其一。

如果 q 是質數 · 那麼至少有一個質數不在有限質數集 $p_1, p_2, p_3 \dots p_n$ 中。

如果 q 是合數 · 那麼存在一個質數因子 p 整除 q · 如果 p 在我們的質數集合 P 中 · 則 p 必然整除 P 。

但是 · 已知 p 整除 $P + 1$ · 如果 p 同時整除 P 和 q · 則 P 必然整除 P 與 q 之差 · 也就是 $(P + 1) - P = 1$

但是沒有質數能整除 1 · 即有 p 整除 q · 就不存在 p 整除 P · 因此 p 不在有限質數集合 P 中。

這就證明了：對於任何一個有限的質數集 · 總有一個質數不在其中 · 所以質數一定是無限的。

Theorem 3

定義 $\pi(x) = \frac{x}{\ln(x)}$ 為質數計數函數 · 也就是小於 x 的質數個數。

$$\text{則若 } x \rightarrow \infty \cdot \lim_{x \rightarrow \infty} \frac{\pi(x)}{x} \approx 1$$

這個定理告訴我們 · 若要尋找所有不超過 x 的質數 · 則他的數量會趨近於 $\frac{x}{\ln(x)}$

若我們要從所有小於 x 的正整數中挑中質數 · 則他的機率為 $\frac{x}{\ln(x)} \div x = \frac{1}{\ln(x)}$

Definition - 最大公因數

令 a, b 為兩整數 · 且 $a, b \neq 0$ · 若存在一個最大的整數 d 使得 $d|a$ 且 $d|b$ · 則 d 被稱做 a, b 的最大公因數。

最大公因數 a, b 被寫作 $gcd(a, b)$ 。

Definition - 互質

令 a, b 為兩整數 · 若 $gcd(a, b) = 1$ · 則我們說 a, b 互質。

Definition - 兩兩互質

若有一整數數列 $a_1, a_2, a_3, a_4, \dots, a_n$ · 若 $gcd(a_i, a_j) = 1, 1 \leq i < j \leq n$ · 則我們說這個數列兩兩互質。

Introduce - 利用質因數分解尋找最大公因數

假設 $a = p_1^{a_1} p_2^{a_2} p_3^{a_3} \dots p_n^{a_n}$ · 且 $b = p_1^{b_1} p_2^{b_2} p_3^{b_3} \dots p_n^{b_n}$

則兩數的 $gcd(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} p_3^{\min(a_3, b_3)} \dots p_n^{\min(a_n, b_n)}$

Definition - 最小公倍數

令 a, b 為兩正整數 · 若存在一個 d 使得 $a|d$ 且 $b|d$ · 則我們說 d 為 a, b 的最小公倍數。

最小公倍數 a, b 被寫作 $lcm(a, b)$

我們可以用質因數分解來尋找最小公倍數 · 也就是

假設 $a = p_1^{a_1} p_2^{a_2} p_3^{a_3} \dots p_n^{a_n}$ · 且 $b = p_1^{b_1} p_2^{b_2} p_3^{b_3} \dots p_n^{b_n}$

則兩數的 $lcm(a, b) = p_1^{\max(a_1, b_1)} p_2^{\max(a_2, b_2)} p_3^{\max(a_3, b_3)} \dots p_n^{\max(a_n, b_n)}$

Theorem 4

令 a, b 為正整數 · 則 $ab = \gcd(a, b) \times \text{lcm}(a, b)$

Proof

待補

Introduce - 歐幾里得演算法

我們可以用歐幾里得演算法來計算 a, b 的最大公因數

論點建立在 $\gcd(a, b) = \gcd(b, c)$, $a > b$ · 其中 c 為 a 除以 b 的餘數。

Introduce - 歐幾里得演算法的正確性

引理1

令 $a = bq + r$ · 其中 $a, b, q, r \in \mathbb{Z}$ · 則 $\gcd(a, b) = \gcd(b, r)$ 。

證明

假設 d 能夠整除 a, b · 則 d 也可以整除 $a - bq = r$ · 因此 · 若 $\gcd(a, b) = d_1$ · 則 $\gcd(b, r) = d_1$

假設 d 能夠整除 b, r · 則 d 也可以整除 $bq + r = a$ · 因此 · 若 $\gcd(b, r) = d_2$ · 則 $\gcd(a, b) = d_2$

因此 · $\gcd(a, b) = \gcd(b, r)$ 。

Introduce - 歐幾里得演算法

歐幾里得演算法可以用來計算兩個整數 (a, b) 的最大公因數。

概念建立在 $\gcd(a, b) = \gcd(b, c)$ · 其中 $a > b$ 且 c 為 a 除 b 的餘數。

Example

Find $\gcd(91, 287)$

$$287 = 91 \times 3 + 14$$

$$91 = 14 \times 6 + 7$$

$$14 = 7 \times 2 + 0$$

因此 $\gcd(91, 287) = \gcd(91, 14) = \gcd(14, 7) = 7$

Introduce - 引理 1

令 $a = bq + r$ · 其中 $a, b, q, r \in \mathbb{R}$ · 則 $\gcd(a, b) = \gcd(b, r)$

Proof

假設 d 可以整除 a, b · 則 d 也可以整除 $a - bq = r$ ·

因此若 a, b 存在一公因數 · 則此公因數也是 b, r 的公因數 ·

假設 d 可以整除 b, r · 則 d 也可以整除 $bq + r = a$ ·

因此若 a, b 存在一公因數 · 則此公因數也是 b, r 的公因數 ·

因此 · $\gcd(a, b) = \gcd(b, r)$

Introduce - 歐幾里得演算法的正確

假設 a, b 為正整數 · 且 $a \geq b$ · 令 $r_0 = a, r_1 = b$ · 則我們可以透過除法定理得到以下步驟 ·

$$r_0 = r_1 q_1 + r_2$$

$$r_1 = r_2 q_2 + r_3$$

$$r_2 = r_3 q_3 + r_4$$

重複多次後

$$r_{n-2} = r_{n-1} q_{n-1} + r_n$$

$$r_{n-1} = r_n q_n$$

則我們可以知道 · 餘數隨著步驟越來越多 · 得到以下結果 ·

$$r_0 > r_1 > r_2 > r_3 \dots \geq 0$$

則根據引理 1 · 得到

$$\gcd(a, b) = \gcd(r_0, r_1) = \gcd(r_1, r_2) = \dots = \gcd(r_{n-1}, r_n) = \gcd(r_n, 0) = r_n$$

因此最大公因數發生在最後一個非零的餘數 ·

Introduce - 貝祖定理

如果 a, b 為正整數 · 則存在一組 s, t 使得 $\gcd(a, b) = as + bt$

Example

$$\gcd(6, 14) = 2 = 6 \times (-2) + 14 \times 1$$

Example

利用歐幾里得演算法 · 將 $\gcd(252, 198) = 18$ 利用線性組合找出 s, t · 使得 $252s + 198t = \gcd(252, 198) = 18$

$$252 = 198 \times 1 + 54$$

$$198 = 54 \times 3 + 36$$

$$54 = 36 \times 1 + 18$$

$$36 = 18 \times 2 + 0$$

又

$$18 = 54 - 36$$

$$= 54 - (198 - 54 \times 3) = 54 \times 4 - 198$$

$$= (252 - 198) \times 4 - 198 = 4 \times 252 - 5 \times 198$$

故我們可以找到 $s = 4, t = -5$ · 使得 $\gcd(252, 198) = 4 \times 252 - 5 \times 198$

我們可以整理成以下的表格

| j | r_j | r_{j+1} | q_{j+1} | r_{j+2} |
|-----|-------|-----------|-----------|-----------|
| 0 | 252 | 198 | 1 | 54 |
| 1 | 198 | 54 | 3 | 36 |
| 2 | 54 | 36 | 1 | 18 |
| 3 | 36 | 18 | 2 | 0 |

Introduce - 歐幾里得擴展演算法

若我們已經找到足夠的 q · 則我們可以將 s, t 擴展成以下的遞迴式。

$$s_j = s_{j-2} - q_{j-1}s_{j-1} \text{ 和 } t_j = t_{j-2} - q_{j-1}t_{j-1}, j \geq 2$$

其中 $s_0 = 1, s_1 = 0, t_0 = 0, t_1 = 1$

Example

利用歐幾里得擴展 · 尋找出 s, t 使得 $\gcd(252, 198) = 252 \times s + 198 \times t$

$$s_2 = s_0 - q_1 s_1 = 1 - 1 \times 0 = 1$$

$$t_2 = t_0 - q_1 t_1 = 0 - 1 \times 1 = -1$$

$$s_3 = s_1 - q_2 s_2 = 0 - 3 \times 1 = -3$$

$$t_3 = t_1 - q_2 t_2 = 1 - 3 \times (-1) = 4$$

$$s_4 = s_2 - q_3 s_3 = 1 - 1 \times (-3) = 4$$

$$t_4 = t_2 - q_3 t_3 = -1 - 1 \times 4 = -5$$

因此 · 我們可以知道 $s = 4, t = -5$ 時符合 $\gcd(252, 198) = 4 \times 252 - 5 \times 198 = 18$

Introduce - 引理 2

如果 a, b, c 為正整數 · 且 $\gcd(a, b) = 1$ 而且 $a|bc$ · 則 $a|c$

Proof

假設 $\gcd(a, b) = 1$ · 且 $a|bc$ · 則

根據貝祖定理 · 存在一組 (s, t) 使得 $\gcd(a, b) = as + bt = 1$

將等號兩邊同乘以 c · 則得到 $asc + btc = c$

已知 $a|bc$ · 代表存在一個 r 使得 $bc = ar$ · 將等號兩邊同乘 t · 可以得到 $btc = art \iff btc = a(rt)$ · 故 $a|btc$

又已知 asc 可被 a 整除 ($asc/a = sc, sc \in \mathbb{R}$) · 故 $a|asc, a|btc \Rightarrow a|c$

Introduce - 引理3

假設 p 是質數 · 且 $p|a_1 a_2 a_3 \dots a_n$ · 則對於某些 i · $p|a_i$

Introduce - 質數分解的唯一性

若一個數字可被質數分解 · 且分解的結果為非遞增的方式排列 · 則這個結果為唯一。

Proof

假設有些大於1的自然數可以用多種方式寫成多個質數的乘積

則假設 n 為最小可以用多種方式寫成多個質數的乘積的數字。

因此我們可以將 n 寫成 $n = p_1 p_2 p_3 p_4 \dots p_s = q_1 q_2 q_3 q_4 \dots q_t$ · p, q 為質數。

根據引理3 · 假設 p 是質數 · 且 $p|a_1 a_2 a_3 \dots a_n$ · 則對於某些 i · $p|a_i$

因此 $q_1 q_2 q_3 \dots q_t$ 存在一個數字使得 p_1 可以整除 · 假設為 q_1

又若要使得 $p_1|q_1$ · 只存在於 $p_1 = q_1$ 。

所以 $n' = p_2 p_3 p_4 p_5 \dots p_s = q_2 q_3 q_4 q_5 \dots q_t$

但 n 是最小的一個 · 不應該存在比 n 更小的數字 n' 能夠用多種方式寫成多個質數的乘積

故與 n 的最小性矛盾 · 因此唯一性得證。

Theorem 5

在先前有介紹過同餘的除法代數運算並不適用於每一種結果 · 在這邊會介紹同餘的除法代數運算。

令 m 為一正整數 · 且 a, b, c 為整數 · 則若 $ac \equiv bc \pmod{m}$ 和 $\gcd(c, m) = 1$ · 則 $a \equiv b \pmod{m}$ 。

Proof

若 $ac \equiv bc \pmod{m}$ · 則 $ac = mr + bc$ · $ac - bc = mr \cdot r = \frac{(ac - bc)}{m} = \frac{ac}{m} - \frac{bc}{m}$

利用引理2可知

若 $m|ac$ 且 $\gcd(m, c) = 1$ · 則 $m|a$

若 $m|bc$ 且 $\gcd(m, c) = 1$ · 則 $m|b$

又因 $m|a, m|b$ · 則 $a \equiv b \pmod{m}$

4.4 求解同餘方程式

Introduce - 線性同餘

當 m 為正整數， a, b 為整數，而 x 為變數時， $ax \equiv b \pmod{m}$ 稱為線性同餘。

Definition - 反元素

若存在一個整數 \bar{a} ，使得 $\bar{a}a \equiv 1 \pmod{m}$ ，則我們說 \bar{a} 為模 m 的反元素。

Theorem 1

若 a 與 m 為互質整數，且 $m > 1$ ，則 a 在模 m 下的反元素存在，此外，在模 m 下，此反元素是唯一的。

Proof (存在性)

利用定理：若 $\gcd(a, m) = 1$ ，則存在兩整數 s, t 使得 $as + mt = 1$ ，也可以看成 $as + mt \equiv 1 \pmod{m}$

由於 $mt \equiv 0 \pmod{m}$ ，因此 $as \equiv 1 \pmod{m}$

因此， s 是 a 在模 m 的反元素。

Introduce - 找出反元素

我們可以用歐幾里得演算法與貝祖定理來找出反元素。

首先我們必須要證明 $\gcd(a, b) = 1$ ，再利用貝祖定理做回推。

Example 1

尋找3模7的反元素。

根據歐幾里得演算法：

$$7 = 3 \times 2 + 1$$

$$3 = 1 \times 3 + 0$$

因此我們能夠證明 $\gcd(7, 3) = 1$

接著利用貝祖定理進行回推，可得 $-2 \times 3 + 1 \times 7 = 1$

因此可以得到貝祖係數 $s = -2, t = 1$

因此， -2 是3模7的反元素，而所有結果為 -2 模7同餘的整數皆為3模7的反元素。

Example 2

尋找101模4620的反元素。

根據歐幾里得演算法：

$$4620 = 101 \times 45 + 75$$

$$101 = 75 \times 1 + 26$$

$$75 = 26 \times 2 + 23$$

$$26 = 23 \times 1 + 3$$

$$23 = 3 \times 7 + 2$$

$$3 = 2 \times 1 + 1$$

$$2 = 1 \times 2 + 0$$

故我們可以證明 $\gcd(101, 4620) = 1$

利用這個來反推貝祖係數

$$\begin{aligned} 1 &= 3 - 2 \times 1 \\ &= 3 - (23 - 3 \times 7) = 3 \times 8 - 23 \\ &= (26 - 23) \times 8 - 23 = 8 \times 26 - 9 \times 23 \\ &= 8 \times 26 - 9 \times (75 - 26 \times 2) = 26 \times 26 - 9 \times 75 \\ &= (101 - 75) \times 26 - 9 \times 75 = 101 \times 26 - 35 \times 75 \\ &= 101 \times 26 - 35 \times (4620 - 101 \times 45) \\ &= 101 \times 1601 - 35 \times 4620 \end{aligned}$$

故我們可以得到 $s = 1601, t = -35$

故我們說 1601 為 101 模 4620 的反元素。

Introduce - 中國剩餘定理

令 $m_1, m_2, m_3, \dots, m_n$ 為兩兩互質的正整數。而 $a_1, a_2, a_3, \dots, a_n$ 為任意正整數。則下列系統

$$\left\{ \begin{array}{l} x \equiv a_1 \pmod{m_1} \\ x \equiv a_2 \pmod{m_2} \\ x \equiv a_3 \pmod{m_3} \\ \vdots \\ \vdots \\ x \equiv a_n \pmod{m_n} \end{array} \right.$$

在 $m = m_1 m_2 m_3 m_4 \dots m_n$ 有唯一解，其中 $0 \leq x < m$ 。而其他解都在 x 模 m 的情況下同餘。

Proof

我們設 $M_k = m/m_{k'}$ ，其中 $k = 1, 2, 3, \dots, n$ 。也就是說 M_k 為除了 m_k 以外的所有兩兩互質正整數乘積。

當 $i \neq k$ 時， M_k 與 m_i 沒有公因數。故 $\gcd(M_k, m_i) = 1$

根據 Theorem 1，可知存在一個反元素使得 $M_k y_k \equiv 1 \pmod{m_k}$

令 $x = a_1 M_1 y_1 + a_2 M_2 y_2 + \dots + a_n M_n y_n$ 。因為 $M_j \equiv 0 \pmod{m_k}$ ，其中 $j \neq k$

因此，對所有的 $k = 1, 2, 3, \dots, n$ ，得到 $x \equiv a_k M_k y_k \equiv a_k \pmod{m_k}$ 。因此 x 即為方程式系統的解。

Example

求解下列同餘方程式系統。

$$\left\{ \begin{array}{l} x \equiv 2 \pmod{3} \\ x \equiv 3 \pmod{5} \\ x \equiv 2 \pmod{7} \end{array} \right.$$

$$m = 3 \times 5 \times 7$$

則

$$M_1 = 5 \times 7 = 35 \cdot 35y_1 \equiv 1 \pmod{3} \cdot 得到 y_1 = 2$$

$$M_2 = 3 \times 7 = 21 \cdot 21y_2 \equiv 1 \pmod{5} \cdot 得到 y_2 = 1$$

$$M_3 = 3 \times 5 = 15 \cdot 15y_3 \equiv 1 \pmod{7} \cdot 得到 y_3 = 1$$

$$\text{因此 } x = 35 \times 2 \times 1 + 21 \times 1 \times 3 + 15 \times 1 \times 2 = 233 \equiv 23 \pmod{105}$$

故 $x = 23$ 為這個系統的最小正整數解。

Introduce - 回朔代換

我們可以用回朔代換來解中國剩餘問題。

已知 $a \equiv b \pmod{m}$ · 則我們可以知道存在一個 k 使得 $a = k \times m + b \cdot k \in \mathbb{Z}$

因此我們可以假設並且代換。

Example

利用回朔代換的方法來找到所有的整數 x · 使得 $x \equiv 1 \pmod{5}$ · $x \equiv 2 \pmod{6}$ · $x \equiv 3 \pmod{7}$

假設 $x = 5t + 1, t \in \mathbb{Z}$ · 則我們可知 $5t + 1 \equiv 2 \pmod{6} \iff 5t \equiv 1 \pmod{6}$

- 利用尋找反元素的方法求得貝祖係數

$$\gcd(5, 6) = 1 \cdot \text{因此我們可以寫成}$$

$$6 = 5 \times 1 + 1$$

$$\text{所以 } 6 \times 1 - 5 \times 1 = 1$$

因此我們可以知道貝祖係數為 $(-1, 1)$ · 可知 $t \equiv -1 \pmod{6}$ · 因此我們假設 $t \equiv 5$ 。

我們假設 $t = 6u + 5$ · 利用代換回 $5t + 1$ 可得 $x = 30u + 26$

接著我們假設 $30u + 26 \equiv 3 \pmod{7} \iff 30u \equiv 4 \pmod{7}$

- 利用尋找反元素的方法求得貝祖係數

$$\gcd(30, 7) = 1 \cdot \text{因此我們可以寫成}$$

$$30 = 7 \times 4 + 2$$

$$7 = 2 \times 3 + 1$$

$$\text{所以 } 7 - 2 \times 3 = 1$$

$$\text{再代換得到 } 7 - (30 - 7 \times 4) \times 3 = 1 \iff 7 \times 13 - 30 \times 3 = 1$$

因此我們可以知道貝祖係數為 $(13, -3)$ · 則可知 $u \equiv 13 \equiv 6 \pmod{7}$

我們假設 $u = 7v + 6$ · 利用代換可知 $x = 30(7v + 6) + 26 = 210v + 206$

因此我們可以知道 $x \equiv 206 \pmod{210}$

Introduce - 費馬小定理

如果 p 是質數，且 a 是整數， $p \nmid a$ ，那麼 $a^{p-1} \equiv 1 \pmod{p}$

費馬小定理可以幫助我們求得指數非常大的數字模 p 的結果，見以下範例。

Example

計算出 $7^{222} \pmod{11}$

$$7^{222} \pmod{11} = [(7^{22})^{10} \times 7^2] \pmod{11} = [1^{10} \times 7^2] \pmod{11} = 5$$

Introduce - 底數為2偽質數

根據費馬小定理，因為 $2^{n-1} \equiv 1 \pmod{n}$ ，則 $n > 2$ 皆為質數。

但根據這個模的結果， n 未必會是質數（下面會給一個反例）。

若存在一個合數使得 $2^{n-1} \equiv 1 \pmod{n}$ 成立，則我們說 n 為底數為 2 的偽質數。

Example

341 是底數為 2 的偽質數。

$341 = 11 \times 13$ ，因此 $\gcd(2, 341) = 1$ 。

因此根據費馬小定理 $2^{340} \equiv 1 \pmod{341}$ 。

Introduce - 底數為 b 的偽質數

令 b 為一正整數，如果 n 為合數且使得 $b^{n-1} \equiv 1 \pmod{n}$ 成立，則我們說 n 為底數為 b 的偽質數。

Introduce - 卡邁爾數 (Optional)

令 n 為一正合數，若對於所有符合 $\gcd(b, n) = 1$ 的 b ，使得 $b^{n-1} \equiv 1 \pmod{n}$ ，則我們說 n 為卡邁爾數。

Example

561 是一個卡邁爾數

$561 = 3 \times 11 \times 17$ ，且若 $\gcd(b, 561) = 1$ ，則 $\gcd(b, 3) = 1, \gcd(b, 11) = 1, \gcd(b, 17) = 1$

因此我們使用費馬小定理，得到 $b^2 \equiv 1 \pmod{3}, b^{10} \equiv 1 \pmod{11}, b^{16} \equiv 1 \pmod{17}$

得到 $b^{560} = (b^2)^{280} \equiv 1 \pmod{3}, b^{560} = (b^{10})^{56} \equiv 1 \pmod{11}, (b^{16})^{35} \equiv 1 \pmod{17}$

因此 $b^{560} \equiv 1 \pmod{561}$ 成立，所以 561 是卡邁爾數。

Introduce - 原根

對於一個質數 p , 定義模 p 下的原根如下:

若 r 為 \mathbb{Z}_p 中的一個元素 · 設有一集合 $A = \{r^k \bmod p | 0 < k < p\}$ · 且集合 A 與集合 \mathbb{Z}_p 中的非零整數一一對應 · 則我們說 r 為模 p 的原根。

Example

2為模11的原根。

$$2^1 \bmod 11 = 2 \cdot 2^2 \bmod 11 = 4 \cdot 2^3 \bmod 11 = 8 \cdot 2^4 \bmod 11 = 5$$

$$2^5 \bmod 11 = 10 \cdot 2^6 \bmod 11 = 9 \cdot 2^7 \bmod 11 = 7 \cdot 2^8 \bmod 11 = 3$$

$$2^9 \bmod 11 = 6 \cdot 2^{10} \bmod 11 = 1$$

Introduce - 離散對數

假設 p 是一個質數 · r 是一個模 p 下的原根 · 而 a 是介於1和 $p - 1$ 之間的一個整數。

如果 $r^e \bmod p = a$ · 其中 $0 \leq e < p$ · 則我們說 e 是以 r 為底 a 模 p 的離散對數 · 寫作 $\log_r a = e$
(p 為已知的質數)

Example 1

試找出以2為底的3模11的離散對數。

$$\because 2^8 \bmod 11 = 3$$

$$\therefore \log_2 3 = 8$$

Example 2

試找出以2為底的5模11的離散對數。

$$\because 2^4 \bmod 11 = 5$$

$$\therefore \log_2 5 = 4$$

4.5 同餘應用

Introduce - 雜湊函數

雜湊函數的概念是 · 使用者丟入key後 · 會透過雜湊函數產生出一個不可逆的值(value)。

且雜湊函數是蓋射 · 故對於一個key · 對應到一個value · 但有可能多個key對應到同一個value · 這就是雜湊碰撞。

我們可以用這樣的函數來表示雜湊函數 · 也就是

$$h(k) = k \bmod m$$

當發生碰撞時，我們可以用線性探測來排除，也就是

$$h(k, i) = (h(k) + i) \bmod m \cdot \text{而 } i \text{ 從 } 0 \text{ 跑到 } m - 1$$

Introduce - 偽隨機數

我們用線性同餘法來製作偽隨機數。

我們會需要三個要素：模數 m ，乘數 a ，增加的數字 c ，以及種子 x_0

則我們可以用以下的遞迴式來製作偽隨機數。

$$x_{n+1} \equiv (ax_n + c) \bmod m$$

產出來的偽隨機數範圍會在 $0 \sim m - 1$ 。

如果我們需要介於 0 到 1 的偽隨機數，我們可以將產出來的偽隨機數除以模數 m ，也就是 x_i / m

Example

假設 $m = 9 \cdot a = 7 \cdot c = 4 \cdot x_0 = 3$ ，找出偽隨機數的數列。

$$\text{則 } x_1 \equiv (7 \times 3 + 4) \pmod{9} \equiv 7 \pmod{9}$$

$$x_2 \equiv (7 \times 7 + 4) \pmod{9} \equiv 8 \pmod{9}$$

$$x_3 \equiv (7 \times 8 + 4) \pmod{9} \equiv 6 \pmod{9}$$

$$x_4 \equiv (7 \times 6 + 4) \pmod{9} \equiv 1 \pmod{9}$$

$$x_5 \equiv (7 \times 1 + 4) \pmod{9} \equiv 2 \pmod{9}$$

$$x_6 \equiv (7 \times 2 + 4) \pmod{9} \equiv 0 \pmod{9}$$

$$x_7 \equiv (7 \times 0 + 4) \pmod{9} \equiv 4 \pmod{9}$$

$$x_8 \equiv (7 \times 4 + 4) \pmod{9} \equiv 5 \pmod{9}$$

$$x_9 \equiv (7 \times 5 + 4) \pmod{9} \equiv 3 \pmod{9}$$

因此偽隨機數的序列會是 $\{7, 8, 6, 1, 2, 0, 4, 5, 3, 7, 8, 6, 1, 2, 0, 4, 5, 3, \dots\}$

4.6 密碼學

Introduce - 凱薩密碼

凱薩密碼是指將一串英文的每個單字遞增或遞減 m 次（例如 A->B, B->C，或者 B->A, A->Z），作法如下。

1. 把所有英文字母用 Z_{26} 作替代，也就是用 0~25 替代每一個英文字母
2. 加密函數就是 $f(p) = (p + m) \bmod 26$ ，因此 $f(p)$ 的值域也為 0~25
3. 接著把所有的整數轉回英文字母。

如果凱薩密碼要進行還原，則我們只需要將所有字母位移的部分移回去。

因此我們可以得到解密函數 $f^{-1}(p) = (p - m) \bmod 26$

對於凱薩密碼來說， m 就是解開凱薩密碼的鑰匙。

Example

嘗試用凱薩加密法加密「MEET YOU IN THE PARK」。

先將「MEET YOU IN THE PARK」轉成數字，也就是「12 4 4 19 24 14 20 8 13 19 7 4 15 0 17 10」

接著假設 $m = 3$ ，那我們將每個數字套用加密函數，得到加密後的訊息「15 7 7 22 27 17 23 11 16 22 10 7 18 3 20 13」

接著將訊息還原成英文字母，也就是「PHHW BRX LQ WKH SDUN」

Introduce - 仿射密碼法

我們可以用 $f(p) = (ap + b) \pmod{26}$ 來增加其安全性。

設定整數 a, b 讓 $f(p)$ 變成雙射函數，函數 $f(p)$ 為雙射函數若為且若 $\gcd(a, 26) = 1$

這樣的函數叫做仿射轉換，而這樣的加密方式被稱為仿射密碼法。

解密這樣的密碼法，首先我們設有一整數 c ，且 $c = (ap + b) \pmod{26}$

已知 $\gcd(a, 26) = 1$ ，則我們可以利用同餘方程式，得到 $c \equiv (ap + b) \pmod{26}$

接下來的目標是解出 p ，所以我們將等式兩端減去 b ，得到 $(c - b) \equiv ap \pmod{26}$

則由於 $\gcd(a, 26) = 1$ ，所以可以找到一個反元素 \bar{a} 使得 $a\bar{a} \equiv 1 \pmod{26}$

我們在等式兩端乘上 \bar{a} ，得到 $\bar{a}(c - b) \equiv a\bar{a}p \pmod{26}$

得到 $\bar{a}(c - b) \equiv p \pmod{26}$

仿設密碼法的密碼分析

分析位移密碼法最主要的工具是計算密文中字母出現的頻率。

若出現的次數最多的字母為E，則有可能就是從E去做位移的，因此我們可以考慮從E下手。

若還原出來的東西毫無意義，可以選擇其他的字母嘗試進行還原。

Introduce - 區塊密碼法

我們可以把字串分成一個一個區塊，針對於每個區塊去進行加密，見以下的轉換密碼法。

Introduce - 轉換密碼法

轉換密碼法是區塊密碼法的其中一種。

利用一個集合 $\{1, 2, 3, \dots, m\}$ 的排列函數 σ 當成密鑰。

首先先將訊息字母分成 m 個字的區塊，如果不滿 m 個字可以隨意在尾端加上幾個字。

接下來對於每一個區塊中的字母 $p_1 p_2 p_3 \dots p_m$ ，編碼成 $c_1 c_2 c_3 \dots c_m = p_{\sigma(1)} p_{\sigma(2)} p_{\sigma(3)} \dots p_{\sigma(m)}$

解碼時則必須要找到 σ^{-1} 。

Example

根據集合 $\{1, 2, 3, 4\}$ 的排列函數 $\sigma(1) = 3, \sigma(2) = 1, \sigma(3) = 4$ 以及 $\sigma(4) = 2$ 執行轉換密碼法。

(a) 將明文PIRATE ATTACK編碼

(b) 將明文SWUE TRAE OEHS解碼

將PIRATE ATTACK分成 m 個字的區塊，得到PIRE TEAT TACK

接著利用排列函數做替換，得到IEPR ETTA AKTC，即為加密後的密文。

利用排列函數來反向替換，把明文SWUE TRAE OEHS進行解密，得到USEU ATER HOSE

Introduce - 密碼系統

一個密碼系統包含了五個集合(P, C, K, E, D)，其中 P 為明文字串所形成的集合， C 為密文字串所形成的集合。

K 為密鑰空間， E 為編碼函數所形成的集合，而 D 為解碼函數所形成的集合。

E_k 表示在 E 中與密鑰 k 相關的編碼函數

對於所有的明文字串 p ， $D_k(E_k(p)) = p$ 。

Introduce - 私密金鑰密碼系統

在私密金鑰密碼系統中，如果知道加密的密鑰，則便能很快地找到解密的密鑰。

例如，使用平移密碼，只要找出平移的數量 k ，反向操作一次就能找到解密的訊息。

在使用私密金鑰系統時，秘密互通的兩方都必須各自擁有密鑰，且擁有密鑰的人都能解開訊息。

因此秘密通訊的兩人必須要私下交換對方的加密金鑰，而類似平移密碼法或者仿射密碼法皆很容易被解開。

Introduce - 公開金鑰密碼系統

為了避免互通訊息的每一方都需要私下交換對方的加密金鑰，而衍伸出了公開金鑰密碼系統。

在公開金鑰密碼系統，知道訊息如何加密並無助於解密，且在此系統下，每個人都知道加密的金鑰，但是解密金鑰則是秘密的。

只有指定的接收者能獲得解密金鑰，並且用其解開訊息。

而若非指定的接收者獲得解密金鑰，則需要經過非常複雜的運算，才能得到解密金鑰，光是用電腦運算就需要10億年的時間。

Introduce - RSA加密系統

在RSA加密系統中，每個人都能使用公開的金鑰 (n, e) 來編碼，其中 $n = pq$ ， p, q 為兩個長達200位的質數。

而使用來做底數的 e 與 $(p - 1)(q - 1)$ 互質，為了產生一個有效的金鑰，則需要兩個很大的質數來進行加密。

經過這個加密的 n 近乎400位數，因此要在有限的時間內質因數分解這個 n 近乎不可能。

因此若沒有解密金鑰，則要快速獲得金鑰基本上是很困難的事情。

Introduce - RSA加密法

在使用公開金鑰 (n, e) 的RSA加密法中，首先訊息 M 被轉換成一個整數字串，每個字母轉換成一個兩位數的整數。

將這些數字字串分割成 $2N$ 的區塊 m ，形成較大的整數，其中 $2N$ 是一個小於 n 的偶數。

如果 M 的長度不足 $2N$ ，則可以在尾端加入許多餘字母。

經過以上的步驟後，我們有數字區塊 $m_1 m_2 m_3 m_4 \dots m_k$ ，其中 k 為正整數。

接下來對於每一個區塊，我們利用以下的函數加密：

$$C = m^e \pmod{n}$$

將密文以數字區塊的方式丟給指定的接收者，由於RSA把一個一個區塊加密，所以RSA也是一種區塊加密法。

Introduce - RSA解密法

如果知道 $(p - 1)(q - 1)$ 下 e 的反元素，也就是解密金鑰 d ，那麼就能很快地找出明文。

d 必定會存在，因為 $\gcd(e, (p - 1)(q - 1)) = 1$ ，因此反元素存在。

也就是若 $de \equiv 1 \pmod{(p - 1)(q - 1)}$ ，則存在一個 k ，使得 $de = 1 + k(p - 1)(q - 1)$

因此 $C^d \equiv (M^e)^d = M^{de} = M^{1+k(p-1)(q-1)} \pmod{n}$

根據費馬小定理（若 $\gcd(M, p) = \gcd(M, q)$ 均成立，則在大部分的情況下都會成立）

則有 $M^{p-1} \equiv 1 \pmod{p}$ ，與 $M^{q-1} \equiv 1 \pmod{q}$

則 $C^d \equiv M \cdot (M^{(p-1)})^{k(q-1)} \pmod{p} \equiv M \cdot 1 = M \pmod{p}$

$C^d \equiv M \cdot (M^{(q-1)})^{k(p-1)} \pmod{q} \equiv M \cdot 1 = M \pmod{q}$

因為 $\gcd(p, q) = 1$ ，根據中國剩餘定理，存在唯一一個 M ，使得 $C^d \equiv M \pmod{pq}$

Introduce - 密鑰交換

在這邊討論一種方法，能夠使得接收方與傳送方不需要分享任何資訊，便能透過公開管道來交換密鑰。

假設接收方與傳送方要分享一個共同的金鑰，則他們的交換方法有以下幾個步驟，均在集合 \mathbb{Z}_p 中。

1. 接收方與傳送方決定好一個質數 p 與一個原根 a
2. 傳送方選擇一個數字 k_1 ，計算 $a^{k_1} \pmod{p}$ ，然後把值傳送給接收方
3. 接收方選擇一個數字 k_2 ，計算 $a^{k_2} \pmod{p}$ ，然後把值傳送給傳送方
4. 接收方計算 $(a^{k_2})^{k_1}$
5. 傳送方計算 $(a^{k_1})^{k_2}$
6. 最後，他們能夠擁有共同的金鑰，因為 $(a^{k_2})^{k_1} \pmod{p} = (a^{k_1})^{k_2} \pmod{p}$

要從 $a^{k_2} \pmod{p}$ 推出 k_2 ，與從 $a^{k_1} \pmod{p}$ 推出 k_1 ，必須反推解出離散對數問題。

在 p 與 a 非常大的情況下，計算基本上無法執行，因此利用公開訊息推出私密訊息是近乎不可能的。

Introduce - 數位簽章

假設傳送方的RSA公開金鑰為 (n, e) ，而私密金鑰為 d ，則他能使用函數 $E_{(n,e)}(x) = x^e \pmod{n}$ 為明文 x 加密。

而收到密文 y 後，便可以使用 $D_{(n,e)}(y) = y^d \pmod{n}$ 解密。

當傳送方要將訊息 M 傳送給接收方，則他可以使用像RSA的分塊方式。

先將訊息轉成數字後，分塊成數個區塊 $m_1, m_2, m_3, \dots, m_k$ ，其中 $0 \leq m_i \leq n$ ， $i = 1, 2, 3, \dots, k$ 。

接下來將他的解密函數運作在所有的區塊上，得到 $D_{(n,e)}(m_i)$ ，傳送給接收方。

當接收方接收到加密後的訊息後，只需要利用公開的加密函數來解密，得到 $E_{(n,e)}(D_{(n,e)}(x)) = x$

因此這樣就能確定訊息來自傳送方了。

5. 歸納與遞迴

5.1 數學歸納法

Introduce - 數學歸納法的原理

為了證明對於所有的 $P(n)$ 為真， $n \in \mathbb{Z}, n > 0$ ，則我們需要完成兩個步驟：

1. 基礎步驟：證明 $P(1)$ 為真
2. 歸納步驟：證明 $P(k) \rightarrow P(k + 1)$ ， $k \in \mathbb{Z}, k > 0$ 為真。

在歸納證明中， $p(k)$ 為歸納假設，我們必須要假設 $P(k)$ 為真，接著證明 $P(k + 1)$ 也為真。

數學歸納法可以被轉成以下的推論：

$(P(1) \wedge \forall k(P(k) \rightarrow P(k + 1))) \rightarrow \forall n P(n)$ ，其中這個推論的定義域為正整數。

在數學歸納的證明，我們並不會假設所有的 $P(k)$ 皆為真，我們會假設 $P(k)$ 為真，來證明 $P(k + 1)$ 也為真。

在數學歸納的證明，起始的數字也不一定會是 1，可以是某個整數 b 。

Example 1

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

基礎步驟：證明 $P(1)$ 為真，把 1 帶進去後得到 $\frac{1(1 + 1)}{2} = 1$

歸納步驟：假設 $P(k)$ 為真，則 $1 + 2 + \dots + k + (k + 1) = \frac{k(k + 1)}{2} + (k + 1)$
 $= \frac{k(k + 1) + 2(k + 1)}{2} = \frac{(k + 1)(k + 2)}{2}$

因此，對於所有的正整數 n ， $P(n)$ 為真。

Example 2

猜出一個式子能夠證明前 n 個正整數奇數的和，然後證明式子是正確的

$$1 = 1 \cdot 1 + 3 = 4 \cdot 1 + 3 + 5 = 9 \cdot 1 + 3 + 5 + 7 = 16$$

因此我們可以猜，若 n 代表前 n 個奇數的和，則 $P(n) = n^2$

利用數學歸納法來證明式子是正確的

基礎步驟： $P(1) = 1$

歸納步驟：假設 $P(k)$ 為真

$$1 + 3 + 5 + 7 + \dots + (2k - 1) + (2k + 1) = [1 + 3 + 5 + \dots + (2k - 1)] + (2k + 1)$$

$$= k^2 + (2k + 1) \text{ (因為歸納假設)}$$

$$= k^2 + 2k + 1 \\ = (k + 1)^2$$

因此，對於所有正整數 k ， $P(k)$ 皆為真，因此前 n 個正整數奇數的和為 n^2 。

Example 3

利用數學歸納法來證明對於所有的正整數 n ， $n < 2^n$

令 $P(n)$ 為 $n < 2^n$ 的命題，則

基礎步驟： $P(1)$ 為真，因為 $1 < 2$

歸納步驟：假設 $P(k)$ 為真，則對於任意的 k ，使得 $k < 2^k$

證明 $P(k+1)$ ，則 $k+1 < 2^k + 1 \leq 2^k + 2^k = 2 \times 2^k = 2^{k+1}$

因此對於所有的正整數 n ， $n < 2^n$ 。

Example 4

利用數學歸納法證明，對於所有 $n \geq 4$ ， $2^n < n!$

令 $P(n)$ 為 $2^n < n!$ 的命題，則

基礎步驟： $P(4)$ 為真，因為 $2^4 = 16 < 4! = 24$

歸納步驟：假設 $P(k)$ 為真，得到任意的 $k \geq 4$ 使得 $2^k < k!$

為了證明 $P(k+1)$ 為真，我們得到 $2^{k+1} = 2 \times 2^k$

根據歸納假設，得到 $2^{k+1} < 2 \times k! < (k+1) \times k! < (k+1)!$

因此，對於任意的 $k \geq 4$ ， $2^n < n!$ 均成立。

Example 5

利用數學歸納法證明對於任意的正整數 n ， $n^3 - n$ 可被 3 整除。

令 $P(n)$ 為 $n^3 - n$ 的命題，則

基礎步驟： $P(1)$ 為真，因為 $1^3 - 1 = 0 \cdot 0 \mod 3 = 0$

歸納步驟：假設 $P(k)$ 為真，那麼 $k^3 - k$ 可被 3 整除

為了證明 $P(k+1)$ 為真，有

$$(k+1)^3 - (k+1) = k^3 + 3k^2 + 3k + 1 - k - 1 = k^3 + 3k^2 + 2k \\ = (k^3 - k) + 3k^2 + 3k = (k^3 - k) + 3(k^2 + k)$$

根據歸納假設，已知 $(k^3 - k)$ 可被 3 整除，第二項為三的倍數，因此也必定能被 3 整除。

因此，對於任意的正整數 n ， $P(n)$ 均成立。

Example 6

利用數學歸納法來證明如果 S 是有限集合，有 n 個元素，且 n 為非負整數，則 S 有 2^n 個子集合。

令 $P(n)$ 為一命題，代表如果 S 是有限集合，有 n 個元素，且 n 為非負整數，則 S 有 2^n 個子集合。

基礎步驟： $P(0)$ 為真，因為 $2^0 = 1$ ，空集合的子集合只有自己本身。

歸納步驟：假設 $P(k)$ 為真，那麼令 T 為一集合，有 $k + 1$ 個元素，那麼 $T = S \cup \{a\}$

其中 $a \in T$ ，且 $S = T - \{a\}$ ，那麼 $|S| = k$

如果 $T = S \cup \{a\}$ ，那麼 T 的子集有兩種可能：包含 a 或者不包含 a

透過之前的歸納假設已知 S 的子集合有 2^k 個，則我們可以知道 T 的子集合數量為 $2 \times 2^k = 2^{k+1}$

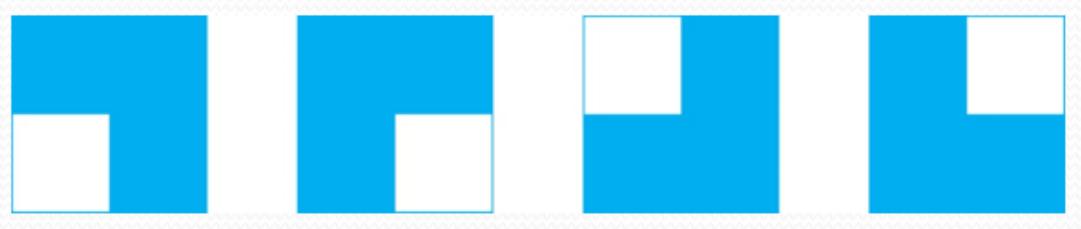
因此，對於任意的非負整數 n ， $P(n)$ 均成立。

Example 7

證明在 $2^n \times 2^n$ 的棋盤上，刪去一個方格後可以用三格板填滿棋盤。

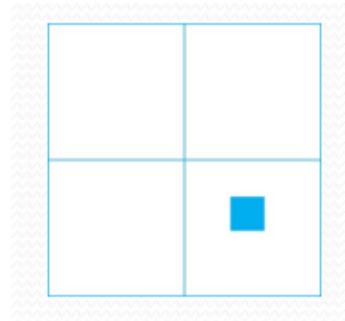
設 $P(n)$ 的命題為在 $2^n \times 2^n$ 的棋盤上，刪去一個方格後可以用三格板填滿棋盤。

基礎步驟： $P(1)$ 為真，列出所有可能的三格板填滿方式，均多一格



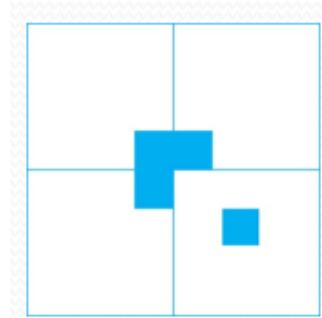
歸納步驟：假設一個任意的正整數 k ， $P(k)$ 為真，則

我們考慮 $2^{k+1} \times 2^{k+1}$ 其中一個方格被刪去了，我們分割棋盤變成四個 $2^k \times 2^k$ 的棋盤



如果我們能夠刪去這四個棋盤的其中一格，那麼根據 $P(k)$ ， $2^{k+1} \times 2^{k+1}$ 能被填滿

一個棋盤已經被刪去了，我們可以用一塊三格板來占走這三個棋盤，如下



這樣，四個棋盤都至少被占走一格，根據歸納假設可以知道這四個棋盤被占走一格，因此能夠被填滿
因此， $2^{k+1} \times 2^{k+1}$ 可以被三格板填滿

Example 8

利用數學歸納法，證明對於所有大於等於12的數字n，可以用4與5來湊成。

令 $P(n)$ 為一命題，代表對於數字n，可以用4和5來湊成。

基礎步驟： $P(12)$ 為真，因為可以用3個4來湊成。

歸納步驟：假設 $P(k)$ 為真，為了要證明 $P(k + 1)$ 為真，我們考慮兩種情況：

1. 如果 $k + 1$ 是偶數，那麼當我們全部都用4來湊時，會剛好或者多2。

由於數字n大於等於12，因此若全部用4來湊，則4的數量必定大於等於3

因此我們可以把其中兩個4替換成兩個5，得到多2的結果。

2. 如果 $k + 1$ 是奇數，那麼當我們全部都用4來湊時，會多1或者多3

由於數字n大於等於12，因此若全部用4來湊，則4的數量必定大於等於3

因此若是多1，只需把其中一個4換成5即可。

若是多3，只需把其中3個4換成5即可。

綜合以上，可以得到無論 $k + 1$ 是奇數或是偶數， $P(k + 1)$ 均成立。

因此，若 $n \geq 12$ ， $P(n)$ 皆成立。

Introduce - 數學歸納法的正確性

數學歸納法能夠正確，主要來自於良序規則，也就是對於所有的非空正整數集合，至少會存在一個最小元素。

數學歸納法的證明如下：

1. 假設知道 $P(1)$ 為真，且對於所有正整數來說， $P(k) \rightarrow P(k + 1)$ 為真。
2. 假設這裡有至少一個正整數n能使 $P(n)$ 為假，則我們假設有一正整數非空集合S能夠使 $P(n)$ 為假。
3. 根據良序定理，S至少有一個最小元素，叫做m
4. 我們知道m不可能為1，因為 $P(1)$ 為真
5. 因為m為正整數且大於1，所以 $m - 1$ 也必定為正整數，因為 $m - 1 < m$ ，所以 $m - 1$ 必定不在S內
所以 $P(m - 1)$ 必定為真
6. 但是對於所有的正整數k， $P(k) \rightarrow P(k + 1)$ 為真，所以 $P(m)$ 必定為真，會與 $P(m)$ 為假的結論矛盾。
7. 因此，對於所有的正整數n， $P(n)$ 必定為真。

Introduce - 數學歸納證明的模板

1. 將需要證明的命題表示為「對於所有的 $n \geq b$ ， $P(n)$ 」的形式，b為一個固定的整數
2. 寫下基礎步驟，證明 $P(b)$ 為真，注意選擇正確的b，這樣就完成了證明的第一步
3. 寫下歸納步驟
4. 明確列出歸納假設，形式是「假設 $P(k)$ 為真，對於任意固定的整數 $k \geq b$ 」
5. 列出在歸納假設的前提下需要證明的命題，即寫出 $P(k + 1)$ 的含意
6. 採用 $P(k)$ 證明 $P(k + 1)$ ，確保對於所有 k ， $k \geq b$ ， $P(k)$ 均為真，k也有可能等於b。

5.2 強歸納法

Introduce - 強歸納法

證明對於所有的正整數 n ， $P(n)$ 均為真

其中 $P(n)$ 是一命題，需要完成以下兩個步驟

基礎步驟：證明 $P(1)$ 是真的

歸納步驟：證明對於所有的正整數 k ， $[P(1) \wedge P(2) \wedge \dots \wedge P(k)] \rightarrow P(k+1)$ 為真

Example 1

假設有一個無限階梯，我們已知我們可以抵達第一階，就能抵達第二階，利用數學證明法來證明我們可以抵達每一階。

設 $P(n)$ 為一命題，代表我們可以踏上第 n 階階梯

基礎步驟： $P(1)$ 為真，因為我們可以踏上第一階。

歸納步驟：我們假設 $P(k)$ 為真，代表我們可以踏上前 k 階階梯，那麼對於所有大於等於2的正整數 k ，我們可以抵達 $k+1$ 階，只要我們能夠踏上 $k-1$ 階，根據歸納假設可知我們可以抵達 k 階，因此我們可以抵達 $k+1$ 階。

因此，我們可以抵達所有的階梯。

Example 2

證明對於大於1的整數 n 可以寫成質數的乘積。

令 $P(n)$ 為一命題，代表整數 n 可以寫成質數的乘積。

基礎步驟： $P(2)$ 為真，因為 $2 = 2 \cdot 2$ 為質數

歸納步驟：假設 $P(k)$ 為真，則代表 $P(j)$ 皆為真，其中 $2 \leq j \leq k$

為了要證明 $P(k+1)$ 為真，則

如果 $k+1$ 是質數，那 $P(k+1)$ 必為真

如果 $k+1$ 不是質數，那麼 $k+1$ 可以寫成兩個數 a, b 乘積的和，也就是 $k+1 = a \times b$ ，也就是 $2 \leq a, b \leq k+1$

根據歸納假設，可以知道 $P(a), P(b)$ 皆為真，因此 $k+1$ 可以被寫成質數的乘積。

因此， $P(k+1)$ 為真

Introduce - 歸納法的使用時機

強歸納法可以取代數學歸納法，但是有的時候很難用強歸納法來證明，用數學歸納法歸納比較簡單的話，就用數學歸納法。

Axiom - 良序公理

任何一個非空的非負整數集合都有最小元素。

Example

利用良序關係證明除法定理，也就是假設 a 為整數， d 為一正整數，則存在唯一 q, r 使得 $a = dq + r$ ，其中 $0 \leq r < d$

設 S 是 $a - dq$ 的非負整數集合，其中 q 是整數。

這個集合非空，因為 $-dq$ 可以很大，只要 $q < 0$ 時取絕對值是很大的整數即可

根據良序關係， S 有最小元素 $r = a - dq_0$ ，其中 r 是非負整數，並且 r 必定小於 d

為了證明 r 必定小於 d ，我們可以先假設 $r \geq d$ 的情況，所以可以知道 $a = dq_0 + r = dq_0 + d = d(q_0 + 1)$

所以 $a - d(q_0 + 1) = a - dq_0 - d = r - d \geq 0$ ，由於 $r - d \geq 0$ ，因此不符合 r 是最小元素。

因此 r 必定小於 d ，也因此存在滿足 $0 \leq r < d$ 的整數 r 和 q 。

5.3 遞迴定義和結構歸納法

Definition - 遞迴

遞迴分成兩個步驟

1. 基礎步驟：定值 $f(0)$
2. 紿定一個規則，用之前的結果來計算出這個值

遞迴函數 $f(n)$ 也可以寫成序列 $a_0, a_1, a_2, \dots, a_n$ ，其中 $f(i) = a_i$

Example 1

假設 $f(0) = 3 \cdot f(n + 1) = 2f(n) + 3$ ，找出 $f(1), f(2), f(3), f(4)$

$$f(1) = 2f(0) + 3 = 9$$

$$f(2) = 2f(1) + 3 = 21$$

$$f(3) = 2f(2) + 3 = 45$$

$$f(4) = 2f(3) + 3 = 93$$

Example 2

寫出一個遞迴式，用來找到 $f(n) = n!$

1. 定值 $f(0) = 1$
2. 我們定義， $f(n + 1) = (n + 1) \times f(n)$

因此， $f(0) = 1 \cdot f(1) = (0 + 1) \times f(0) = 1 \cdot f(2) = (1 + 1) \times f(1) = 2 \cdot f(3) = (2 + 1) \times f(2) = 6 \dots$

Introduce - 斐波納契數

斐波納契數的定義如下： $f(0) = 0 \cdot f(1) = 1 \cdot$ 則 $f(n) = f(n - 1) + f(n - 2)$

Example 1

找出斐波納契數的 $f(2), f(3), f(4), f(5)$

$$f(2) = f(1) + f(0) = 1$$

$$f(3) = f(2) + f(1) = 2$$

$$f(4) = f(3) + f(2) = 3$$

$$f(5) = f(4) + f(3) = 5$$

Example 2

證明當 $n \geq 3$ 時， $f_n \geq \alpha^{n-2}$ ，其中 $\alpha = \frac{1 + \sqrt{5}}{2}$

設 $P(n)$ 為一命題，代表 $f_n \geq \alpha^{n-2}$ 。欲證明 $n \geq 3$ 時 $P(n)$ 皆為真。

$$\text{基礎步驟：已知 } \alpha < 2 = f_3 \text{，則 } \alpha^2 = \frac{(3 + \sqrt{5})}{2} < 3 = f_4$$

因此 $P(3)$ 與 $P(4)$ 都為真。

歸納步驟：假設 $P(k)$ 為真，則 $P(j)$ 皆為真，其中 $3 \leq j \leq k$ ， $k \geq 4$ 。

欲證明 $P(k+1)$ 為真，則 $f_{k+1} > \alpha^{k-1}$ 。已知 α 為 $x^2 - x - 1 = 0$ 的解，則 $\alpha^2 - \alpha - 1 = 0$ 。

$$\text{因此 } a^{k-1} = a^2 \times a^{k-3} = (a+1)a^{k-3} = a^{k-2} + a^{k-3}$$

$$\text{根據歸納假設，得知 } f_{k-1} > a^{k-3} \text{，} f_k > a^{k-2}$$

$$\text{因此 } f_{k+1} > f_k + f_{k-1} > \alpha^{k-2} + \alpha^{k-3} = \alpha^{k-1} \text{，得證。}$$

Introduce - 拉密定理

令 a, b 為正整數，滿足 $a \geq b$ ，則歐幾里得演算法求出 $\gcd(a, b)$ 使用的除法將會小於等於 b 的十進位位數的5倍。

利用歐幾里得求出 $\gcd(a, b)$ 時，會使用以下的等式：

$$r_0 = r_1 q_1 + r_2 \text{，} 0 \leq r_2 < r_1$$

$$r_1 = r_2 q_2 + r_3 \text{，} 0 \leq r_3 < r_2$$

重複多次後

$$r_{n-2} = r_{n-1} q_{n-1} + r_n \text{，} 0 \leq r_n < r_{n-1}$$

$$r_{n-1} = r_n q_n$$

我們使用了 n 次除法定理，我們可以保證商數 $q_1, q_2, q_3 \dots q_{n-1}$ 都至少為1。

又因為 $r_{n-1} > r_n$ ，故我們可以保證 $q_n \geq 2$ 。

則可以得到以下的結論

$$r_n \geq 1 = f_2$$

$$r_{n-1} \geq 2r_n \geq 2f_2 = f_3$$

$$r_{n-2} \geq r_{n-1} + r_n \geq f_3 + f_2 = f_4$$

以此類推

$r_2 \geq r_3 + r_4 \geq f_{n-1} + f_{n-2} = f_n$

設 $b = r_1$ · 得到 $b = r_1 \geq r_2 + r_3 \geq f_n + f_{n-1} = f_{n+1}$

從斐波納契數Example2可以知道 · 當 $n > 2$ 時 · $f_{n+1} > \alpha^{n-1}$ · 其中 $\alpha = (1 + \sqrt{5})/2$ · 因此得出 $b \geq \alpha^{n-1}$

因為 $\log_{10} \alpha \approx 0.208 > 1/5$ · 所以可以得到 $\log_{10} b > (n - 1) \log_{10} \alpha > (n - 1)/5$

因此 · $5 \times \log_{10} b > n - 1$ 。假設 b 有 k 個 10 進制位數 · 則 $b < 10^5$ · 也就是 $\log_{10} b < k$ 。

得出 $n - 1 < 5k$ · 因為 k 是整數 · 則 $n \leq 5k$ · 得證。

因為 b 的十進位數為 $\lfloor \log_{10} b \rfloor + 1$ · 故我們需要除小於等於 $5(\log_{10} b + 1)$ 次才能得到結果

因為 $O(5(\log_{10} b + 1)) = O(\log b)$ · 可知當 $a > b$ 時 · 歐幾里得演算法需用 $O(\log b)$ 次除法求出 $\gcd(a, b)$ 的結果。

Introduce - 以遞迴的方式定義的集合與結構

集合的遞迴定義也有兩個部分 · 即基礎步驟與遞迴步驟。

在基礎步驟裡 · 必須指定初始的集合元素 · 在遞迴步驟裡必須提供已知元素形成新元素的規則。

遞迴定義也可能包含排除規則 : 以遞迴方式所定義的集合 · 除了基礎步驟所指定的元素以及應用遞迴步驟所產生的元素以外 · 不包含任何其他的元素。

因此除非元素從基礎步驟或者遞迴步驟中弄出來 · 否則他並不會出現在集合中。

Example 1

令 S 是正整數的子集合 · 遞迴定義為

基礎步驟 : $3 \in S$

遞迴步驟 : 若 $x \in S$ 且 $y \in S$ · 則 $x + y \in S$

根據基礎步驟 · 在 S 裡找到的元素是 3 · 而第一次應用遞迴步驟得到 $3 + 3 = 6 \in S$

第二次得 $3 + 6 = 6 + 3 = 9 \in S$ · $6 + 6 = 12 \in S$ · 以此類推。

Introduce - 字串

由字母集 Σ 所遞迴建構出的字串集 Σ^* 如下

1. 基礎步驟 : $\lambda \in \Sigma^*$ · 其中 λ 是空字串
2. 遞迴步驟 : 如果 $w \in \Sigma^*$ 而且 $x \in \Sigma$ · 則 $wx \in \Sigma^*$

Example

設 $\Sigma = \{0, 1\}$ · 則字串集 Σ^* 有這些字串 · $0, 1, 00, 01, 10, 11 \dots$

Introduce - 字串串接

兩個字串可以透過串接運算子來做串接。

設有一字母集 Σ 與接下來由遞迴建構出的字串集 Σ^*

則我們可以定義運算元，如下：

1. 基礎步驟：如果 $w \in \Sigma^*$ ，則 $w \cdot \lambda = w$
2. 遞迴步驟：如果 $w_1 \in \Sigma^*$ 、 $w_2 \in \Sigma^*$ 且 $x \in \Sigma$ ，則 $w_1 \cdot (w_2 x) = (w_1 \cdot w_2)x$

通常來說，我們會把 $w_1 \cdot w_2$ 寫作 $w_1 w_2$

如果 $w_1 = abcd$, $w_2 = e f g h$ ，則串接 w_1 與 w_2 的結果 $w_1 \cdot w_2 = w_1 w_2 = abcdefgh$

Introduce - 字串長度

給定一個遞迴定義 $l(w)$ ，代表字串 w 的長度，定義如下：

1. 基礎步驟： $l(\lambda) = 0$
2. 遞迴步驟：如果 $w \in \Sigma^*$ 且 $x \in \Sigma$ ，則 $l(wx) = l(w) + 1$

Introduce - 平衡括號

給定一個遞迴定義，用來產生平衡括號集 P

平衡括號的定義是每個左括號一定有一個對應的右括號

例如 $()()$ 是合法的平衡括號，或者 $((())()$ 也是，但是 $)()$ 不是

平衡括號集的定義如下：

1. 基礎步驟： $() \in P$
2. 遞迴步驟：如果 $w \in P$ ，則 $w(), (w), ()w \in P$

Introduce - 命題邏輯的合式公式

定義包含 T 、 F 、命題變數，以及運算符號形成的集合 $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$ 中的運算符號之命題邏輯的合式公式集合，來代表複合命題。

基礎步驟： T 、 F 和 s 都是合式公式，其中 s 是命題變數

遞迴步驟：如果 E 、 F 都是合式公式，則 $(\neg E)$ 、 $(E \wedge F)$ 、 $(E \vee F)$ 、 $(E \rightarrow F)$ 、 $(E \leftrightarrow F)$ 都是合式公式。

例如 $((p \vee q) \rightarrow (q \wedge F))$ 是合式公式，但 $p q \wedge$ 不是合式公式。

Introduce - 有根樹圖

有根樹圖的集合可以依照下列的步驟以遞迴的方式加以定義。

一個有根樹圖的組成包含稱為根點的特殊頂點，以及其他頂點和連接這些頂點的邊所形成的集合。

基礎步驟：單一頂點 r 是有根樹圖。

遞迴步驟：假設 T_1 、 T_2 、 \dots 、 T_n 是有根樹圖，其根點分別為 r_1 、 r_2 、 \dots 、 r_n

則從某個不屬於有根樹圖 T_1 、 T_2 、 \dots 、 T_n 的根 r 所開始的圖，再加上每一個從 r 連到頂點 r_1 、 r_2 、 \dots 、 r_n 的邊

所得的結果仍然是有根樹圖。

Introduce - 二元樹

二元樹圖是一種特殊形態的有根樹圖。

在二元樹圖定義的遞迴步驟中，新的樹圖都是由兩個二元樹圖結合而成的，原先的兩個樹圖分別標示為左子樹圖與右子樹圖。

Introduce - 延伸二元樹圖

延伸二元樹圖的集合可以用遞迴的方式定義。

基礎步驟：單個節點是延伸二元樹

遞迴步驟：若存在兩個互斥延伸二元樹 T_1, T_2 ，頂點各為 R_1, R_2

則存在一個新的延伸二元樹 T ，頂點為 R ，連接左子樹 T_1 的頂點 R_1 與右子樹 T_2 的頂點 R_2 ，且兩個延伸二元樹非空。

Introduce - 滿二元樹圖

滿二元樹圖的集合可依下列步驟以遞迴的方式定義。

基礎步驟：存在僅含單一頂點 r 的完全二元樹。

遞迴步驟：如果 T_1, T_2 是滿二元樹圖，則存在滿二元樹圖

其組成是根點 r ，再加上連接這個根點與左子樹圖 T_1 和右子樹圖 T_2 所含每個根點的邊。

5.4 遞迴演算法

Definition - 遞迴演算法

藉由把問題簡化到更小輸入的相同問題來解決問題的演算法，稱為遞迴演算法。

Example

求出計算 $n!$ 的遞迴演算法，其中 n 為非負整數。

基礎步驟： $0! = 1$

遞迴步驟：如果 $f(n) = n!$ ，則 $f(n) = (n) \times f(n - 1)$

因此我們可以寫成以下的遞迴程式

```
int factorial(int n){  
    if(n == 0) return 1;  
    else return n * factorial(n-1);  
}
```

Introduce - 證明遞迴演算法的正確性

數學歸納法與由其變化而得的強歸納法，皆都能用來證明遞迴演算法的正確性，即是對所有可能的輸入，都能得到想要的輸出。

Example

給定以下演算法，證明以下的遞迴演算法能夠正確的算出實數之幕次。

```
procedure power(a: nonzero real number, n: nonnegative integer){  
    if n = 0 then return 1  
    else return a * power(a, n-1);  
    {output is a^n}  
}
```

基礎步驟：若 $n = 0$ ，則結果為 1。這個步驟是正確的因為對於任意的實數 a ， $a^0 = 1$ 。

歸納步驟：假設對所有的 $a \neq 0$ ，和任意的非負整數 k 而言， $\text{power}(a, k) = a^k$ ，也就是說演算法執行出來的 a^k 是正確的。

現在證明 a^{k+1} 是正確的，因為 $k + 1$ 是正整數，演算法執行 $\text{power}(a, k + 1) = a \cdot \text{power}(a, k)$ 。

根據歸納甲說， $\text{power}(a, k) = a^k$ ，因此 $\text{power}(a, k + 1) = a \cdot a^k = a^{k+1}$ ，得證。

6. 計數

6.1 計數的基礎

Introduce - 乘法法則

假設一個程序可以分解成兩個連續的任務。如果完成第一個任務有 n 種方法，在完成第一個任務後，有 m 種方法完成第二個任務，則完成這整個程序共有 mn 種方法。

Example 1

一間只有兩位員工的新公司，租下一個有 12 間辦公室的房子，有幾種不同的方式能將這兩個雇員安排於不同的辦公室？

考慮第一個員工，他有 12 間辦公室可以選擇，則當他選擇其中一間後，第二個員工只剩下 11 間辦公室可以選擇。

根據乘法法則，可以知道共有 132 種選擇將辦公室分配給這兩位員工。

Example 2

長度為 7 的二進位字串有多少個？

考慮到每個 bit 的選擇不是 1 就是 0，因此七個 bit，每個 bit 可以選擇 1 或者 0。

根據乘法定理，數量為 $2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 2^7 = 128$

Introduce - 加法法則

如果一種任務有兩種方式可以完成，一個方式有 n 種方法，另一個方式有 m 種方法，則完成此任務的方法有 $n + m$ 種。

Example 1

假設要從數學教師或主修數學的學生中選一個人擔任校委會代表。

如果有37位數學教師與83位主修數學的學生，則有多少種不同的選擇？

我們有37種方式可以從數學教師中挑出一個擔任校委會代表，以及有83種方式可以從主修數學的學生挑出一個擔任校委會的代表。

因此共有 $37 + 83 = 120$ 種方法。

Example 2

一名學生可以從三份清單中選擇一個計劃來完成任務，這三份清單分別包含23、15、19種計畫，則可以選擇的計畫有多少種？

這名學生可以從第一份清單中選擇23種可能的計畫來完成，從第二份清單中選擇15種可能的計畫來完成，以及從第三份清單選擇19種可能的計畫來完成。

因此共有 $23+15+19=57$ 種計畫可以選擇。

Introduce - 更複雜的計數問題

有很多的複雜計數問題無法單獨用加法法則或乘法法則來解決，見以下。

Example

一個註冊表單的要求是一個6到8字元構成的登入密碼。

其中密碼由大寫字母與數字所組成，且要求至少要包含一個數字，求密碼的所有組合。

考慮所有的可能，減去沒有含數字的可能，就代表至少會含一個數字。

因此，考慮所有合法密碼 P_n 的數量，其中 n 代表字串的長度。

$$P_6 = 36^6 - 26^6 = 1867866560$$

$$P_7 = 36^7 - 26^7 = 70332353920$$

$$P_8 = 36^8 - 26^8 = 2612282842880$$

因此所有的可能 $P = P_6 + P_7 + P_8 = 2684483063360$

Introduce - 減法法則 (兩個集合的排容原理)

若有項任務能以兩種方式完成，分別有 n 種與 m 種方法數，則完成此項任務的所有方法數量共有 $n + m$ 再減去兩種方法中相同的方法。

Example 1

以字元1開始或者以字串00結束的8位元字串有多少個?

考慮以1開始的八位元字串，共有 $2^7 = 128$ 個

考慮以00結束的八位元字串，共有 $2^6 = 64$ 個

考慮以1開始且以00結束的八位元字串，共有 $2^5 = 32$ 個

因此所有的可能 $P = 128 + 64 - 32 = 160$ 種。

Introduce - 除法定理

若完成某項任務的所有方式總共有 n 個步驟，而每個方式都需要 d 個步驟，則完成此任務的方法有 n/d 種方式。

Example

將四個人安排坐在一張圓桌，若每個人左右坐的人相同時，視為同一種方式，則一共有多少種不同的安排方式？

任意選擇圓桌的一個位置，設定這張椅子的編號為1，接下來以順時鐘連續標記座位。

將人安排在1號座位有四種方法，2號座位有三種方法，3號座位有兩種方法，4號座位有一種方法。

所以共有 $4! = 24$ 種方式來安排四個人入座一個圓桌。

但無論把人先安排在1號、2號、3號、4號都沒差，因此 $24/4 = 6$ 種不同的方法。

Introduce - 樹狀圖

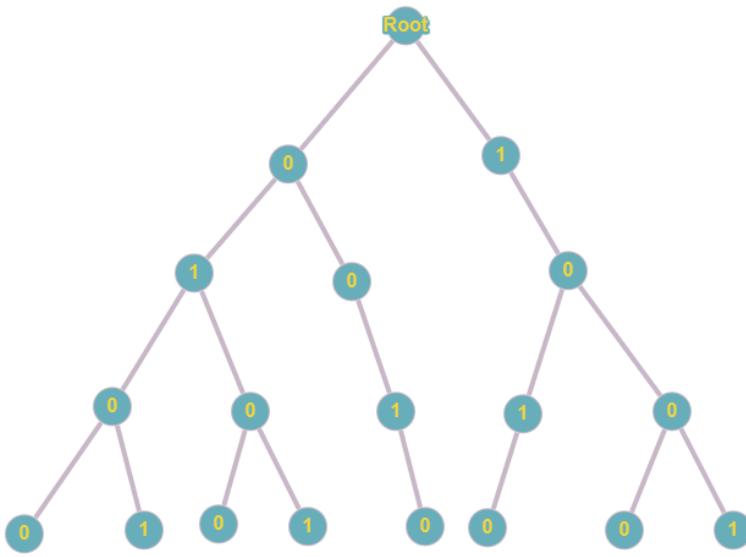
樹狀圖可以用來解決計數問題。

一棵樹包含根結點，以及分支出去的子節點，到無法分支的節點稱為葉節點。

使用樹狀圖求解時，所有的葉節點必須是獨一無二的。

Example

不含連續兩個1的四位元字串共有多少個？



答案是8。

6.2 鴿洞原理

Theorem - 鴿洞原理

若 k 為一正整數，如果有 $k + 1$ 個東西放進 k 個箱子內，則至少會有一個箱子包含2個或更多東西。

Corollary 1

一個函數 f 由含有 $k + 1$ ，或更多個元素的集合對應至只有 k 個元素的集合，一定不會是一對一的。

Proof

如果每個箱子最多都只有一個物品，則物品總數最多為 k 個，與 $k + 1$ 個物品矛盾。

Example 1

在367個人中，一定至少有2個人的生日是在一年的同一天，因為一年至多只有366天。

Example 2

在27個英文詞彙，一定至少有2個詞彙是以同一個字母開始，因為英文字母只有26個。

Example 3

如果考試分數是從0至100，則在班上至少需要101個人，才會存在兩個人分數相同。

Example 4

證明對每一個整數 n ，有一個 n 的倍數僅由0和1組成。

令 n 為一個整數，考慮1、11、111、1111、...、111...111等 $n + 1$ 個整數的序列，也就是序列最後面的元素有 $n + 1$ 個1

因為一個數除以 n 的餘數有 n 個可能，所以序列中有 $n + 1$ 個整數，根據鴿洞原理可以知道至少會存在兩個數字模 n 的餘數相同。

將這兩個數字相減就能得到一個可以被 n 整除的數字，而兩個都用1組出的數字相減位數只會有1或0的可能。

因此每一個整數 n ，必定有一個 n 的倍數僅由0和1組成。

Introduce - 廣義的鴿洞原理

如果 N 個物件放入 k 個箱子，則至少有一個盒子包含至少 $\lceil N/k \rceil$ 個物件。

Proof

假設沒有盒子包含超過 $\lceil N/k \rceil - 1$ 個物件，則物件總數最多是

$$k(\lceil N/k \rceil - 1) < k((N/k) + 1) = N$$

這裡與物品有 N 個東西矛盾。

Example 1

在100個人中，至少有 $\lceil 100/12 \rceil = 9$ 人在同一個月份裡出生。

Example 2

如果成績分為A、B、C、D、F五種標準，在一個離散數學班裡，最少要多少位學生才能保證至少有6位學生得到相同的分數？

已知 $\lceil N/5 \rceil = 6$ ，能找到符合最小的整數 $N = 26$ ，因此需要26個學生才能保證至少有6位學生得到相同的分數。

Introduce - 蘭姆西數字

假設一個團體有6個人，其中任意2個人不是敵人就是朋友。

證明這6個人中要不存在3人彼此都是朋友的小團體，要不存在一個3人都是敵人的小團體。

Example 1

令A是六個人中的其中一人，則其他五人中至少三人是A的朋友，或至少三人是A的敵人。

因為當五個物件被分成兩個集合時，根據廣義鴿洞原理，其中一個集合至少有 $\lceil 5/2 \rceil = 3$ 個元素

在第一種情況中，假設B、C、D是A的朋友，若B、C、D有兩個人也是朋友，則這兩個人和A構成彼此是朋友的三人組。

在第二種情況中，假設B、C、D是A的敵人，若B、C、D有兩個人也是敵人，則這兩個人和A構成彼此是敵人的三人組。

6.3 排列與組合

Theorem - 排列

若 n 與 r 為正整數，則從含有 n 個元素的集合中選取 r 個元素來做排列的方法數有

$$P(n, r) = n(n-1)(n-2)\dots(n-r+1) = \frac{n!}{(n-r)!}$$

Proof

利用乘法定理來進行證明，在排列的第一位有 n 個選擇，第二位有 $n-1$ 個選擇，第三位有 $n-2$ 個選擇...

直至選取第 r 位時，集合中只剩下 $n-r+1$ 個選擇。

完全不取物件出來的排列方式只有一種，所以 $P(n, 0) = 1$

Example 1

從五個學生中，選出三人排出一列來拍照有幾種方法？將這五個學生排成一列來拍照有多少種方法？

$$P(5, 3) = \frac{5!}{3!} = 4 \times 5 = 20$$

$$P(5, 5) = \frac{5!}{0!} = 5! = 120$$

Example 2

在字母 ABCDEFGH 的排列中，包含字母 ABC 的排列有幾種？

因為 ABC 必須排在一起，所以其實這題只有六個物件(ABC)、D、E、F、G、H

將這 6 個物品做排列，也就是 $P(6, 6) = 6! = 720$

Introduce - r-排列

自一個集合中取出 r 個元素來排列稱做 r -排列。

Example

令 $S = \{a, b, c\}$ ，則所有 S 的 2-排列有： $\{a, b\}, \{a, c\}, \{b, a\}, \{b, c\}, \{c, a\}, \{c, b\}$ ，共有有 $P(3, 2) = 6$ 種可能

Theorem - 組合

當 n 為非負整數， r 為整數，其中 $0 \leq r \leq n$ ，則一個包含 n 個相異元素的集合

$$\text{其組合的個數為 } C(n, r) = \frac{n!}{r!(n-r)!} = \frac{(n)(n-1)\dots(n-r+1)}{r!}$$

且 $C(n, r) = C(n, n-r)$

Proof

由於組合無論順序，先找到其排列，除以 r 個元素所有可能的排列的總數即為組合數量（除法定理）。

也就是 $P(n, r)/P(r, r) = C(n, r)$

$$\text{已知 } C(n, r) = \frac{n!}{r!(n-r)!} \cdot C(n, n-r) = \frac{n!}{(n-r)!(n-(n-r))!} = \frac{n!}{r!(n-r)!}$$

因此 $C(n, r) = C(n, n-r)$

Example 1

從一副52張的撲克牌中取出五張牌有幾種可能的組合？若選取47張牌有幾種方法？

$$\text{取出五張牌不需要排列，因此使用組合 } C(52, 5) = \frac{52!}{47!5!} = \frac{52 \times 51 \times 50 \times 49 \times 48}{5!} = 2598960$$

$$\text{選取47張牌，可以知道 } C(52, 47) = \frac{52!}{47!5!} = \frac{52 \times 51 \times 50 \times 49 \times 48}{5!} = 2598960$$

Example 2

一個有10個隊員的網球隊，要挑出五個球員來參加比賽，可以有幾種不同的方式？

轉換問題成在10個元素的集合中，所有挑出五個元素的子集合個數，也就是 $C_5^{10} = 252$

Definition - 組合證明

一個等式的組合證明是一種利用計數論證的證明方式。

證明等式兩端都是在計算某種物件的數量，只是使用方法不同，或者說明有一種雙射的對應關係存在於等式兩端的物件。

這兩種證明的形態分別稱作二重計數證明或是雙射證明。

Proof

首先使用雙射證明法求證對所有整數 n 與 r ，當 $0 \leq r \leq n$ ， $C(n, r) = C(n, n-r)$ 。

假設 S 為包含 n 個元素的集合，每個由包含 r 個元素之 S 子集合 A 對應到包含 $n-r$ 個元素的集合 \bar{A} 的函數都是雙射函數。

這樣一來，我們便證得 $C(n, r) = C(n, n-r)$ ，因為有雙射關係的兩集合之元素個數相同。

使用二重計數證明求證對於所有整數 n 與 r ，當 $0 \leq r \leq n$ ， $C(n, r) = C(n, n-r)$ 。

根據定義，包含 r 個元素的子集合 A 有 $C(n, r)$ 個。每個這樣的子集合 A 都決定出哪些元素不在 A 中，也就是在 \bar{A} 中。

因為 S 中包含 r 個元素之子集合補集有 $n-r$ 個元素，所以應該有 $C(n, n-r)$ 個包含 r 個元素之子集合，所以 $C(n, r) = C(n, n-r)$

6.4 二項式係數及其等式

Introduce - 二項式係數

在 n 個元素中選出 r -組合的方法數為 $\binom{n}{r}$ ，這個數也稱作二項式係數。

因為這些數將出現在二項展開式的係數中，例如 $(a + b)^n$ 。

Theorem - 二項式定理

令 x 與 y 為變數，而 n 為非負整數，則

$$(x + y)^n = \sum_{j=0}^n \binom{n}{j} x^{n-j} y^j = \binom{n}{0} x^n + \binom{n}{1} x^{n-1} y + \dots + \binom{n}{n-1} x y^{n-1} + \binom{n}{n} y^n$$

Example 1

求出 $(x + y)^4$ 的展開式。

$$\begin{aligned} (x + y)^4 &= \sum_{j=0}^4 \binom{4}{j} x^{4-j} y^j = \binom{4}{0} x^4 + \binom{4}{1} x^3 y + \binom{4}{2} x^2 y^2 + \binom{4}{3} x y^3 + \binom{4}{4} y^4 \\ &= x^4 + 4x^3 y + 6x^2 y^2 + 4x y^3 + y^4 \end{aligned}$$

Example 2

求出 $(x + y)^{25}$ 展開式中 $x^{12} y^{13}$ 的係數。

$$\binom{25}{13} = \frac{25!}{12!13!} = 5200300$$

Example 2

求出 $(2x - 3y)^{25}$ 展開式中 $x^{12} y^{13}$ 的係數。

將式子轉化成 $(2x + (-3y))^{25}$

$$\text{因此在 } x^{12} y^{13} \text{ 的係數為 } \binom{25}{12} \times 2^{12} + (-3)^{13} = -\frac{25!}{12!13!} \times 2^{12} \times 3^{13}$$

Corollary 1

$$\text{令 } n \text{ 為非負整數，則 } \sum_{k=0}^n \binom{n}{k} = 2^n$$

Proof

利用二項式定理 · 令 $x = 1, y = 1$

$$\text{則 } 2^n = (1+1)^n = \sum_{k=0}^n \binom{n}{k} 1^k 1^{n-k} = \sum_{k=0}^n \binom{n}{k}$$

Corollary 2

$$\text{令 } n \text{ 為正整數} \cdot \text{ 則 } \sum_{k=0}^n (-1)^k \binom{n}{k} = 0$$

Proof

利用二項式定理 · 令 $x = 1, y = -1$

$$\text{則 } 0^n = (1+(-1))^n = \sum_{k=0}^n \binom{n}{k} (-1)^k 1^{n-k} = \sum_{k=0}^n \binom{n}{k} (-1)^k$$

Corollary 3

$$\text{令 } n \text{ 為非負整數} \cdot \text{ 則 } \sum_{k=0}^n 2^k \binom{n}{k} = 3^n$$

Proof

利用二項式定理 · 令 $x = 1, y = 2$

$$\text{則 } 3^n = (1+2)^n = \sum_{k=0}^n \binom{n}{k} 1^{n-k} 2^k = \sum_{k=0}^n \binom{n}{k} 2^k$$

Theorem - 帕斯卡等式

$$\text{令 } n \text{ 與 } k \text{ 為正整數} \cdot \text{ 其中 } n \geq k \cdot \text{ 則 } \binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$$

Proof

假設 T 為包含 $n+1$ 個元素的集合 · 且 $a \in T \cdot S = T - \{a\}$

其中 T 有 $\binom{n+1}{k}$ 個包含 k 個元素的不同子集合。

將這些子集合分成兩類

其中一個為包含 a 與 $k-1$ 個 S 中的元素 · 數量為 $\binom{n}{k-1}$

另一個為不包含 a · 包含 k 個 S 中的元素 · 數量為 $\binom{n}{k}$

因此得到 $\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$

Introduce - 帕斯卡三角形

上面的等式可以寫成一個三角形，如下。

$$\begin{array}{ccccccc}
 & \binom{0}{0} & & & & & 1 \\
 & \binom{1}{0} \binom{1}{1} & & & & 1 & 1 \\
 & \binom{2}{0} \binom{2}{1} \binom{2}{2} & & \text{By Pascal's identity:} & & 1 & 2 & 1 \\
 & \binom{3}{0} \binom{3}{1} \binom{3}{2} \binom{3}{3} & & \binom{6}{4} + \binom{6}{5} = \binom{7}{5} & & 1 & 3 & 3 & 1 \\
 & \binom{4}{0} \binom{4}{1} \binom{4}{2} \binom{4}{3} \binom{4}{4} & & & & 1 & 4 & 6 & 4 & 1 \\
 & \binom{5}{0} \binom{5}{1} \binom{5}{2} \binom{5}{3} \binom{5}{4} \binom{5}{5} & & & & 1 & 5 & 10 & 10 & 5 & 1 \\
 & \binom{6}{0} \binom{6}{1} \binom{6}{2} \binom{6}{3} \binom{6}{4} \binom{6}{5} \binom{6}{6} & & & & 1 & 6 & 15 & 20 & 15 & 6 & 1 \\
 & \binom{7}{0} \binom{7}{1} \binom{7}{2} \binom{7}{3} \binom{7}{4} \binom{7}{5} \binom{7}{6} \binom{7}{7} & & & & 1 & 7 & 21 & 35 & 35 & 21 & 7 & 1 \\
 & \binom{8}{0} \binom{8}{1} \binom{8}{2} \binom{8}{3} \binom{8}{4} \binom{8}{5} \binom{8}{6} \binom{8}{7} \binom{8}{8} & & & & 1 & 8 & 28 & 56 & 70 & 56 & 28 & 8 & 1
 \end{array}$$

Theorem - 凡德蒙德等式

令 m, n, r 為非負整數，而且 r 不能大於 m 與 n ，則 $\binom{m+n}{r} = \sum_{k=0}^r \binom{m}{r-k} \binom{n}{k}$

Proof

假設在一個集合有 m 個元素，第二個集合有 n 個元素，則在這兩個元素取 r 個元素的方法有 $\binom{n+m}{r}$ 個。

另外一種方法，若在第一個集合取出 k 個元素，第二個集合取出 $r-k$ 個元素，其中 $0 \leq k \leq r$

則在第一個集合取出 k 個元素有 $\binom{n}{k}$ 個方法，在第二個集合取出 $r-k$ 個元素有 $\binom{m}{r-k}$ 個方法。

利用乘法法則，則上面的選取方法有 $\binom{n}{k} \binom{m}{r-k}$ 種方法。

窮舉所有的 k 利用加法定理加起來，即為 $\binom{n+m}{r}$ 的方法數，因此 $\binom{n+m}{r} = \sum_{k=0}^r \binom{n}{k} \binom{m}{r-k}$

Corollary 4

若 n 為非負整數，則 $\binom{2n}{n} = \sum_{k=0}^n \binom{n}{k}^2$

Proof

令 $n = r = m$ ，套入凡德蒙德等式，得 $\binom{n+n}{n} = \sum_{k=0}^n \binom{n}{n-k} \binom{n}{k}$

因為 $\binom{n}{n-k} = \binom{n}{k}$ ，則 $\binom{2n}{n} = \sum_{k=0}^n \binom{n}{k}^2$

Theorem - 朱世傑恆等式

令 n 和 r 為非負整數，且 $r \leq n$ ，則 $\binom{n+1}{r+1} = \sum_{j=r}^n \binom{j}{r}$

Proof

利用數學歸納法來證明。

基礎步驟：令 $n = r$ ，則有 $\sum_{j=r}^r \binom{j}{r} = \binom{r}{r} = 1 = \binom{r+1}{r+1}$

歸納步驟：假設 $P(k)$ 代表 $\binom{k+1}{r+1} = \sum_{j=r}^k \binom{j}{r}$ 為真，其中 $k \geq r$

則證明若 $n = k + 1$ ， $\binom{k+2}{r+1} = \binom{k+1}{r} + \binom{k+1}{r+1} = \binom{k+1}{r} + (\sum_{j=r}^k \binom{j}{r}) = \sum_{j=r}^{k+1} \binom{j}{r}$

根據歸納步驟，得到 $\binom{k+2}{r+1} = \sum_{j=r}^{k+1}$ 。因此 $P(k) \rightarrow P(k+1)$ 為真，證畢。

6.5 一般化的排列與組合

Theorem - 重複排列

有 n 個元素的集合中，允許重複的 r -排列共有 n^r 種

Proof

在 r -排列中，每個位置有 n 個選擇，根據乘法法則，可以得到所有方法數為 n^r 。

Example

利用大寫英文字母能形成多少個長度為 r 的字串

根據乘法法則，大寫字母有 26 個，每個位置可以放的字母有 26 個可能，因此方法數總共有 26^r 個

Theorem - 重複組合

在包含 n 個元素的集合中，允許重複之 r -的組合的個數為 $C(n+r-1, r) = C(n+r-1, n-1)$

Proof

以元素的數量來考慮組合的數量。

我們先假設從四個元素中挑選總共六個數量的方法數，可以知道

這樣是一種選取方式：第一個元素中挑選 2 個，第二個元素中挑選 1 個，第三個元素中挑選 1 個，第四個元素中挑選 2 個

因此可以把問題規約成星號與直線做排列。

以上面的例子來做範例，我們可以考慮成這樣的形式：`**|*|*|**`

無論怎麼排列，都會有 $n - 1$ 個隔板，以及 r 個星星

我們可以把問題規約成在 $n - 1 + r$ 個位置中，選取 $n - 1$ 個位置來擺放隔板，或者選取 r 個位置來擺放星星。

因此方法數即為 $C(n - 1 + r, r) = C(n - 1 + r, n - 1)$

Example 1

假設某餅乾店中有4種不同口味的餅乾。若不計挑選時的順序，選取6塊餅乾有幾種方法？

問題可以規約成在4個元素中，重複選出6個元素的方法有幾個。

因此我們可以利用重複組合，得到 $C(6 + 4 - 1, 6) = C(9, 6) = 84$

Example 2

下面方程組有幾種解？

$$x_1 + x_2 + x_3 = 11$$

考慮到很多個1，總共需要11個，把這些1分成三堆來做組合。

因此問題變成了，有11個1與2個隔板，可以選擇放隔板或者放1來算出方法數

也就是 $C(11 + 3 - 1, 2) = C(11 + 3 - 1, 11) = C(13, 2) = 78$

Theorem - 不可區分物件的排列

若 n 個物件中，第一種相同的物件有 n_1 個，第二種相同的物件有 n_2 個，……，第 k 種相同的物件有 n_k 個

則此 n 個物件的排列方法數有 $\frac{n!}{n_1!n_2!n_3!\dots n_k!}$

Proof

將此 n 個物件排成一列。

首先挑選出 n_1 個位置來放第一種物件，方法數為 $C(n, n_1)$ ，此時剩下 $n - n_1$ 個位置可以放置新的物件。

接下來選出 n_2 個位置來放第二種物件，方法數為 $C(n - n_1, n_2)$ ，此時剩下 $n - n_1 - n_2$ 個位置可以放置新的物件。

以此類推後可以利用乘法法則，得到總方法數為

$$\begin{aligned} & C(n, n_1)C(n - n_1, n_2)\dots C(n - n_1 - n_2 - \dots - n_{k-1}, n_k) \\ &= \frac{n!}{n_1!(n - n_1)!} \frac{(n - n_1)!}{n_2!(n - n_1 - n_2)!} \dots \frac{n - n_1 - \dots - n_{k-1}}{n_k!0!} \\ &= \frac{n!}{n_1!n_2!n_3!\dots n_k!} \end{aligned}$$

Example

將單字SUCCESS的字母重新排列，將形成多少不同的字串？

$$\frac{7!}{3!2!1!1!} = 420$$

Theorem - 將不同物件分配至不同箱子中

將 n 個物件分配到 k 個不同的箱子，使得第 i 個箱子中有 n_i 個物件的方法數為 $\frac{n!}{n_1!n_2!\dots n_k!}$ ，其中 $i = 1, 2, \dots, k$

Proof

考慮到第一個箱子有 n_1 個物件，分配方法數量是 $C(n, n_1) = \frac{n!}{(n - n_1)!n_1!}$

考慮到第二個箱子有 n_2 個物件，分配方法數量是 $C(n - n_1, n_2) = \frac{(n - n_1)!}{(n - n_1 - n_2)!n_2!}$

利用乘法定理，可以得到

$$C(n, n_1) \times C(n - n_1, n_2) \times \dots \times C(n_k, n_k) = \frac{n!}{(n - n_1)!n_1!} \times \frac{(n - n_1)!}{(n - n_1 - n_2)!n_2!} \times \dots \times \frac{n_k!}{n_k!0!} = \frac{n!}{n_1!n_2!\dots n_k!}$$

Example

將一副標準的52張撲克牌分給四個人，一人五張，會有多少種不同的情形？

將問題轉化成，共有四個人，一人五張，剩下的收回牌組(32張)。

因此所有方法為： $\frac{52!}{5!5!5!5!32!} = 1478262843475644110602240$

Introduce - 將相同物件分配至不同箱子中

將 n 個相同物件分配至 k 個不同的箱子的方法數為 $C(n + k - 1, n)$ 。

Example

將10個相同的球放進8個不同的箱子中，共有幾種方法？

問題等於8個元素的集合中找出10—組合的方式，所以可能的情況為 $C(8 + 10 - 1, 10) = C(17, 10) = 19448$

Introduce - 不同物件與相同箱子

令 $S(n, j)$ 表示將 n 個不同的物件分配至 k 個相同的箱子中，且沒有箱子是空的，其中 $S(n, j)$ 稱為第二類斯特林數，其定義如下。

$$S(n, j) = \frac{1}{j!} \sum_{i=0}^{j-1} (-1)^i \binom{j}{i} (j-i)^n$$

將 n 個物件放至 k 個相同的箱子的方法數為 $S(n, 1) + S(n, 2) + \dots + S(n, k)$ ，也就是

$$\sum_{i=1}^k S(n, j) = \sum_{i=1}^k \frac{1}{j!} \sum_{i=0}^{j-1} (-1)^i \binom{j}{i} (j-i)^n$$

Proof

待補。

Introduce - 相同物件與相同箱子

將 n 個相同的物件分配至 k 個相同的箱子中，就是將 n 個分成 j 個小於 k 的正整數 $a_1, a_2, a_3 \dots a_j$

其中 $a_1 \geq a_2 \geq a_3 \dots \geq a_j$ 使得 $a_1 + a_2 + \dots + a_j = n$ 。

6.6 產生組合與排列

Introduce - 產生排列

任何一個包含 n 個元素的集合都能與集合 $\{1, 2, 3, \dots, n\}$ 做個一對一個對應關係。

任何一種 n 個元素的排列方式，都能對應於一個由整數 1 到 n 的排列，且能產生此集合的 $n!$ 種排列。

Introduce - 字典順序

若有兩排列 a, b 長度皆為 n 。

對某個 k ， $1 \leq k \leq n$ ， $a_1 = b_1, a_2 = b_2 \dots a_{k-1} = b_{k-1}$ ，但 $a_k < b_k$ ，則稱排列 $a_1 a_2 \dots a_n$ 大於排列 $b_1 b_2 \dots b_n$ 。

也就是在兩種排列中，在第一次出現不同數字的位置上，數字較大的排列小於數字較小的排列。

Example

在集合 $\{1, 2, 3, 4, 5\}$ 的排列中，23415 排在 23514 前面，因為第一次出現不相同數字的第三個位置 4 小於 5

同理，41532 排在 52143 之前。

Introduce - 利用字典順序產生下一個較大排列

若有一排列 $a_1 a_2 a_3 \dots a_n$

假設 $a_{n-1} < a_n$ ，則我們只需要調換這兩個整數，就可以找到下一個較大的排列。

舉個例子，231456，可知 5 < 6，因此調換這兩個數字得到 231465，就是下一個較大的排列。

假設 $a_{n-1} > a_n$ ，那就考慮倒數第三個數字，若 $a_{n-2} < a_{n-1}$ ，就重新安排最後三個數字來得到下一個較大的排列。

將 a_{n-1} 與 a_n 中較小的數字安排於第 $n-2$ 個位置，剩下兩個數字依照遞增的方式排列，例如 234165 下一個排列即為 231516。

Example 1

依字典順序方式 362541 的下一個較大排列是什麼？

可以知道，我們必須得要考慮第 3 個數字。

在 5、4、1 中選一個大於 2 的最小整數，也就是 4，安排在第 3 個位置。

因此得到364125為下一個較大排列。

Example 2

求出所有由1, 2, 3依據字典順序所得的排列。

首先拿到123。

因為 $2 < 3$ ，因此我們可以調換這兩個數字，得到132為次大排列。

接著因為 $3 > 2$

因此我們必須要考慮第一個數字，從2、3找到一個大於1的最小整數與其交換後將後兩個位置遞增排序，得到213為次大排列。

因為 $1 < 3$ ，因此我們可以調換這兩個數字，得到231為次大排列。

接著因為 $3 > 1$

因此我們必須要考慮第一個數字，從1、3找到一個大於1的最小整數與其交換後將後兩個位置遞增排序，得到312為次大排列。

因為 $1 < 2$ ，因此我們可以調換這兩個數字，得到321為次大排列。

Introduce - 產生組合

組合不過只是一個子集合，因此我們可以用 $\{a_1, a_2, \dots, a_n\}$ 之子集合與長度為 n 的位元字串的對應關係來產生組合。

假設有一個字元字串為 $s = s_1 s_2 s_3 \dots s_n$ ，若 $s_i = 0$ ，則 a_i 不在此子集合內，反之則 a_i 在此集合內。

產生所有組合的方式可以產生一組字元字串，從 s 皆為0慢慢累加1至 s 皆為1，共有 2^n 種。

Example

求出1000100111的下一個位元字串。

$$1000100111 + 1 = 1000101000$$

Introduce - 產生r-組合

每一個 r -組合都能用一個序列來表現，此序列包含依遞增方式排列的子集合，也就是此 r -組合能夠用這個序列的字典順序排列出來。

在字典順序中，最前面的 r -組合是 $\{1, 2, \dots, r-1\}$ ，最後面的組合為 $\{n+r-1, n+r-2, \dots, n-1, n\}$

產生 $a_1 a_2 \dots a_r$ 的下一個 r -組合，首先得要找到最後一個 a_i 的位置，其中 $a_i \neq n-r+i$

然後以 a_{i+1} 替換 a_i ，當 $j = i+1, i+2, \dots, r$ 時，以 $a_i + j - i + 1$ 來替換 a_j 。

Example

在集合 $\{1, 2, 3, 4, 5, 6\}$ 中，找出 $\{1, 2, 5, 6\}$ 的下一個4-組合。

已知最後一個出現 a_i 的地方為 $i = 2$ 。

因此，我們令 $a_2 = a_2 + 1 = 3$ ，接著 $a_3 = 2 + 3 - 2 + 1 = 4$ ， $a_4 = 2 + 4 - 2 + 1 = 5$

得到下一個較大的4—組合為 $\{1, 3, 4, 5\}$

9. 關係

9.1 關係與其性質

Definition - 二元關係

令 A 與 B 為集合，一個由 A 到 B 的二元關係是 $A \times B$ 的子集合。

由 A 到 B 的二元關係是個有序對集合 R ，每個有序對的第一個分量來自 A ，第二個分量來自集合 B 。

我們使用符號 $a R b$ 來表示 $(a, b) \in R$ ，而 $a \not R b$ 表示 $(a, b) \notin R$

當我們說 (a, b) 屬於 R 時，則我們說 a 與 b 有關係。

Example 1

令 A 為學校中的學生集合， B 為開授課程所形成的集合。

令關係 R 包含 (a, b) ，表示學生 a 選修課程 b 。

例如：若Xuan與Uriah都選修了計算機程式設計，則有序對(Xuan, 計算機程式設計)和(Uriah, 計算機程式設計)都屬於關係 R 。

若Xuan還選修了離散數學，則(Xuan, 離散數學)也在關係 R 中。

若Uriah沒有選修離散數學，則(Uriah, 離散數學)不在關係 R 中。

Example 2

若 A 為臺灣所有區所形成的集合，而 B 為臺灣所有縣市的集合。

若區 A 位於縣市 B 中，則定義元素 (a, b) 在關係 R 中。

例如：(新化區, 台南市) · (烏日區, 台中市) · (大安區, 台北市) · (新莊區, 新北市) 和 (左營區, 高雄市) 都在關係 R 中

但(板橋區, 台北市) · (新市區, 高雄市) 就不在關係 R 中，因為板橋區在新北市，新市區在臺南市。

Introduce - 將函數視為一種關係

由集合 A 到集合 B 的函數 f ，指派唯一一個 B 中的元素給每一個 A 中的元素，則 f 的元素即為有序對 (a, b) ，其中 $b = f(a)$ 。

因為 f 之圖形為 $A \times B$ 的子集合，所以是一個由 A 到 B 的關係。

反之，若 R 為 A 到 B 的關係，使得每個 A 中的元素指出現在有序對的第一個分量中一次，則可定義出一個函數，使得 R 為其圖形。

由指派唯一一個 B 中的元素 b 給每一個 A 中的元素 a ，使得 $(a, b) \in R$ ，便能觀察出此性質。

關係能用來呈現集合 A 和 B 元素間一對多的關聯性。

其中某個 A 的元素對應到多個 B 的元素，而函數表現的是每個 A 的元素只與一個 B 有關係。

Definition - 集合上的關係

一個在集合 A 上的關係 · 是指由 A 到 A 的關係 · 也就是 $A \times A$ 的子集合。

Example 1

令 A 為集合 $\{1, 2, 3, 4\}$ · 哪些序對屬於關係 $R = \{(a, b) | b \bmod a = 0\}$?

稍微列舉一下 · 可知 $R = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (2, 4), (3, 3), (4, 4)\}$

Example 2

考慮下面整數集合上的關係：

$$R_1 = \{(a, b) | a \leq b\}$$

$$R_2 = \{(a, b) | a > b\}$$

$$R_3 = \{(a, b) | a = b \vee a = -b\}$$

$$R_4 = \{(a, b) | a = b\}$$

$$R_5 = \{(a, b) | a = b + 1\}$$

$$R_6 = \{(a, b) | a + b \leq 3\}$$

哪一個關係包含下列每一個有序對 $(1, 1), (1, 2), (2, 1), (1, -1)$ 和 $(2, 2)$?

考慮 $(1, 1)$ · 可以知道 $(1, 1)$ 在關係 R_1, R_3, R_4, R_6 中

考慮 $(1, 2)$ · 可以知道 $(1, 2)$ 在關係 R_1, R_6 中

考慮 $(2, 1)$ · 可以知道 $(2, 1)$ 在關係 R_2, R_5, R_6 中

考慮 $(1, -1)$ · 可以知道 $(1, -1)$ 在關係 R_2, R_3, R_6 中

考慮 $(2, 2)$ · 可以知道 $(2, 2)$ 在關係 R_1, R_3, R_4 中

因此沒有一個關係包含上述所有有序對。

Example 3

在包含 n 個集合的關係上 · 共有多少個不同的關係？

在集合 A 上的關係是 $A \times A$ 的子集合 · 也就是當 A 有 n 個元素時 · $A \times A$ 有 n^2 個元素。

一個包含 m 個元素的集合共有 2^m 個子集合 · 所以 $A \times A$ 有 2^{n^2} 個子集合。

Definition - 反身性

對於所有的 $a \in A$ · 若 $(a, a) \in R$ · 則稱集合 A 上的關係 R 為反身性的 (reflexive) ·

Example 1

考慮下面集合 $\{1, 2, 3, 4\}$ 上的關係：

$$R_1 = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 4), (4, 1), (4, 4)\}$$

$$R_2 = \{(1, 1), (1, 2), (2, 1)\}$$

$$R_3 = \{(1, 1), (1, 2), (1, 4), (2, 1), (2, 2), (3, 3), (4, 1), (4, 4)\}$$

$$R_4 = \{(2, 1), (3, 1), (3, 2), (4, 1), (4, 2), (4, 3)\}$$

$$R_5 = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (2, 3), (2, 4), (3, 3), (3, 4), (4, 4)\}$$

$$R_7 = \{(3, 4)\}$$

哪一個關係是反身性的？

可以觀察到 R_3, R_5 是有反身性的，因為他們包含所有 (a, a) 的有序對。

#####

Example 2

正整數上的「整除」關係是否為反身性的？

因為對所有正整數 a 而言 $a|a$ ，所以「整除」關係是反身性的。

Example 3

在含有 n 個元素之集合上的關係中，有多少具不同的反身性？

在集合 A 上的關係是一個 $A \times A$ 的子集合，因此 R 中的元素由 $A \times A$ 個 n^2 元素中取出。

若 R 是反身性的，則 n 個 (a, a) 的有序對一定得包含在 R 中，其中 $a \in A$ 。

剩下的 $n(n - 1)$ 個 (a, b) ，其中 $a, b \in R, a \neq b$ 的元素不一定要在 R 中。

根據乘法定理，共有 $2^{n(n-1)}$ 個不同的反身性關係。

Definition - 對稱性與非對稱性

設有一集合 A 與一關係 R ，若對所有的 $a, b \in A$ ，當 $(a, b) \in R$ 時， $(b, a) \in R$ ，則集合 A 上的關係 R 有對稱性。

若對所有的 $a, b \in A$ ，當 $(a, b) \in R$ 與 $(b, a) \in R$ 時 $a = b$ ，則集合 A 上的關係 R 有反對稱性。

對稱性可以寫成 $\forall a \forall b ((a, b) \in R \rightarrow (b, a) \in R)$ ，而反對稱性可以寫成 $\forall a \forall b ((a, b) \in R \wedge (b, a) \in R \rightarrow (a = b))$

Example 1

參考以下的關係，找出對稱性與反對稱性。

$$R_1 = \{(a, b) | a \leq b\}$$

$$R_2 = \{(a, b) | a > b\}$$

$$R_3 = \{(a, b) | a = b \vee a = -b\}$$

$$R_4 = \{(a, b) | a = b\}$$

$$R_5 = \{(a, b) | a = b + 1\}$$

$$R_6 = \{(a, b) | a + b \leq 3\}$$

R_3, R_4, R_6 會是對稱性。

R_3 是對稱性的，因為當 $a = b$ 時 $b = a$ ，或者當 $a = -b$ 時 $b = -a$ 。

R_4 是對稱性的，因為當 $a = b$ 時 $b = a$

R_6 是對稱性的，因為當 $a + b \leq 3$ 時 $b + a \leq 3$ 。

R_1, R_2, R_4, R_5 會是反對稱性的。

R_1 是反對稱性的，因為當 $a \leq b$ 時，若 $b \leq a$ ，則 $a = b$ 。

R_2 是反對稱性的，因為不可能出現 $a > b$ 且 $b > a$ 。

R_4 是反對稱性的，因為若出現 $a = b$ 且 $b = a$ 時， $a = b$ 。

R_5 是反對稱性的，因為若出現 $a = b + 1$ 且 $b = a + 1$ ，則 $a - b = b - a = 1$ ，由於不存在 (a, b) 來滿足等式，故 R_5 是反對稱性的。

Example 2

正整數集合上的「整除」關係是否為對稱性的？是否為反對稱性的？

可以知道，若 $a \bmod b = 0$ ，則 $a \geq b$ ，正整數上的整除不符合對稱性。

但若 $a \bmod b = 0$ 且 $b \bmod a = 0$ ，則 $a \geq b$ 且 $b \geq a$ ，因此 $a = b$ ，故正整數上的整除符合反對稱性。

Definition - 遷移性

設有一集合 A 與一關係 R ，對所有的 $a, b, c \in A$ ，且 $(a, b) \in R \wedge (b, c) \in R$ 時 $(a, c) \in R$ ，則稱集合 A 上的關係 R 稱為遷移性的。

可以寫成 $\forall a \forall b \forall c (((a, b) \in R \wedge (b, c) \in R) \rightarrow (a, c) \in R)$

Example 1

參考以下的關係，找出遷移性的關係。

$$R_1 = \{(a, b) | a \leq b\}$$

$$R_2 = \{(a, b) | a > b\}$$

$$R_3 = \{(a, b) | a = b \vee a = -b\}$$

$$R_4 = \{(a, b) | a = b\}$$

$$R_5 = \{(a, b) | a = b + 1\}$$

$$R_6 = \{(a, b) | a + b \leq 3\}$$

R_1, R_2, R_3, R_4 是遷移性的。

R_1 是遷移性的，因為當 $a \leq b$ 且 $b \leq c$ 時符合 $a \leq c$ 。

R_2 是遷移性的，因為當 $a > b$ 且 $b > c$ 時符合 $a > c$

R_3 是遷移性的，因為當 $a = \pm b$ 且 $b = \pm c$ 時符合 $a = c$ 。

R_4 是遞移性的，因為當 $a = b$ 且 $b = c$ 時符合 $a = c$ 。

其餘都不是遞移性的。

R_5 不是遞移性的，因為當 $a = b + 1$ 且 $b = c + 1$ 時，則 $a = c + 1$ 可能不成立。例如 $(a, b) = (4, 3)$ 且 $(b, c) = (3, 1)$ 。

R_6 不是遞移性的，因為當 $a + b \leq 3$ 且 $b + c \leq 3$ 時， $a + c \leq 3$ 可能不成立。例如 $(a, b) = (4, -5)$ 且 $(b, c) = (-5, 6)$

Example 2

正整數集合上的「整除」關係是否為遞移性的？

若 $(a, b) \in R$ ，也就是 a 能整除 b ，則 $b \bmod a = 0$ ，則我們可以說 $b = ad$

若 $(b, c) \in R$ ，也就是 b 能整除 c ，則 $c \bmod b = 0$ ，則我們可以說 $c = bk$

因此 $c = bk = (ad)b$ ，利用定義知道 a 能整除 c ，可以知道正整數集合上的「整除」關係為遞移性的。

Introduce - 關係的組合

由 A 到 B 的關係是 $A \times B$ 的子集合，因此兩個關係的組合可以使用所有集合間的組合來表示。

Example

令 $A = \{1, 2, 3\}$ 且 $B = \{1, 2, 3, 4\}$ 。

關係 $R_1 = \{(1, 1), (2, 2), (3, 3)\}$ ， $R_2 = \{(1, 1), (1, 2), (1, 3), (1, 4)\}$ ，能用以下組合。

$$R_1 \cup R_2 = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (3, 3)\}$$

$$R_1 \cap R_2 = \{(1, 1)\}$$

$$R_1 - R_2 = \{(2, 2), (3, 3)\}$$

$$R_2 - R_1 = \{(1, 2), (1, 3), (1, 4)\}$$

Definition - 合成關係

令 R 為由集合 A 到集合 B 的關係，而 S 為由集合 B 到集合 C 的關係。

R 與 S 的合成也是個關係，包含有序對 (a, c) ，其中 $a \in A, c \in C$ ，而且存在一個 $b \in B$ ，使得 $(a, b) \in R$ 和 $(b, c) \in S$ 。

我們將 R 和 S 的合成記為 $S \circ R$ 。

Example

令 R 為由 $\{1, 2, 3\}$ 到 $\{1, 2, 3, 4\}$ 的關係， $R = \{(1, 1), (1, 4), (2, 3), (3, 1), (3, 4)\}$ 。

而 S 為由 $\{1, 2, 3, 4\}$ 到 $\{0, 1, 2\}$ 的關係， $S = \{(1, 0), (2, 0), (3, 1), (3, 2), (4, 1)\}$ 。

R 與 S 的合成為何？

檢驗所有的有序對，得到 $S \circ R = \{(1, 0), (2, 1), (2, 2), (3, 0), (3, 1)\}$

Definition - R的冪次定義

令 R 為集合 A 上的關係。

當 $n = 1, 2, 3, \dots$ 冪次 R^n 的遞迴定義如下： $R^1 = R$ 且 $R^{n+1} = R^n \circ R$

因此 $R^2 = R \circ R \cdot R^3 = R^2 \circ R = (R \circ R) \circ R$

Example

令 $R = \{(1, 1), (2, 1), (3, 2), (4, 3)\}$ 。求出冪次 R^n 。其中 $n = 1, 2, 3, \dots$ 。

因為 $R^2 = R \circ R \cdot$ 我們得到 $R^2 = \{(1, 1), (2, 1), (3, 1), (4, 2)\}$ 。

接下來，因為 $R^3 = R^2 \circ R \cdot R^3 = \{(1, 1), (2, 1), (3, 1), (4, 1)\}$

繼續計算可以發現 $R^4 = R^3 \cdot R^5 = R^3 \dots$ 可以得到 $R^n = R^3, n = 4, 5, 6, 7\dots$

Theorem 1

集合 A 上的關係 R 是遞移性的，若且唯若 $R^n \subseteq R \cdot n = 1, 2, 3, \dots$

Proof

假設 $R^n \subseteq R \cdot n = 1, 2, 3, \dots$ 因此 $R^2 \subseteq R$ 。

我們能由此證明 R 是遞移性的，若 $(a, b) \in R$ 和 $(b, c) \in R$ ，根據合成的定義， $(a, c) \in R^2$ 。

由於 $R^2 \subseteq R$ ，得到 $(a, c) \in R$ ，因此 R 是遞移性的。

利用數學歸納法來證明，可以知道 $n = 1$ 是明顯會成立的。

因此我們假設 $R^n \subseteq R$ ，其中 n 為正整數，往證 R^{n+1} 亦為 R 的子集合。

首先假設 $(a, b) \in R^{n+1} = R^n \circ R$ ，因而存在 $x \in A$ 使得 $(a, x) \in R$ 且 $(x, b) \in R^n$ 。

根據歸納假說， $R^n \subseteq R$ ，我們有 $(x, b) \in R$ 。

由於 R 是遞移性的， $(a, x) \in R$ 且 $(x, b) \in R$ ，可推導出 $(a, b) \in R$ 。

因而 $R^{n+1} \subseteq R$ ，得證。

9.2 n元關係及其應用

Definition - n元關係

令 A_1, A_2, \dots, A_n 為集合，一個在這些集合上的 n 元關係是 $A_1 \times A_2 \times \dots \times A_n$ 的子集合。

集合 A_1, A_2, \dots, A_n 稱為這個關係的域(domain)，而 n 稱為這個關係的階(degree)

Example 1

令 R 為 $N \times N \times N$ 的關係，由有序三項 (a, b, c) 組成。其中 a, b, c 為整數，且 $a < b < c$ 。

$(1, 2, 3) \in R$ 但 $(2, 4, 3) \notin R$ 。此關係的階數為3，而其域皆為自然數集合。

Example 2

令 R 為 $\mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}$ 的關係，由有序三項 (a, b, c) 組成。其中 a, b, c 形成一個算術數列。

$(a, b, c) \in R$ ，若為且若存在整數 k ，使得 $b = a + k$ 且 $c = a + 2k$ ，或是 $b - a = k$ 且 $c - b = k$ 。

可以發現 $(1, 3, 5) \in R$ ，因為 $3 = 1 + 2$ 且 $5 = 1 + 2 \times 2$ 。但是 $(2, 5, 9) \notin R$ ，因為 $5 - 2 = 3$ 而 $9 - 5 = 4$ 。

此關係的階數為3，而其域皆為整數集合。

Example 3

令 R 為 $\mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}^+$ 上的關係，由有序三項 (a, b, m) 組成。其中 a, b 和 m 為整數， $m \geq 1$ ，且 $a \equiv b \pmod{m}$ 。

$(8, 2, 3), (-1, 9, 5)$ 和 $(14, 0, 7)$ 都屬於 R 。但是 $(7, 2, 3), (-2, -8, 5)$ 和 $(11, 0, 6)$ 不屬於 R 。

因為 $8 \equiv 2 \pmod{3}, -1 \equiv 9 \pmod{5}$ 和 $14 \equiv 0 \pmod{7}$ 。

然而 $7 \not\equiv 2 \pmod{3}, -2 \not\equiv -8 \pmod{5}$ 與 $11 \not\equiv 0 \pmod{6}$ 。

此關係的階數為3，而其域前兩個為整數集合，第三個為正整數集合。

Example 4

令 R 為有序5元 (A, N, S, D, T) 所組成的關係，其中 A 為航空公司， N 為航班數， S 為出發地， D 為目的地， T 為起飛時間。

例如，中華航空的123班機在19:00由松山機場飛往澎湖機場，則 $(\text{中華}, 123, \text{松山}, \text{澎湖}, 19:00)$ 屬於 R 。

此關係的階數為5，而其域為所有航空公司集合、航班集合、城市集合與時間集合。

Introduce - 資料庫與關係

在資料庫中進行資訊操作所需的時間與資訊儲存方式有關，以下將討論一種根據關係的概念而來的關聯式資料模型。

資料庫由紀錄組成，紀錄是由欄位形成的有序 n 項。

例如，一個學生紀錄資料庫可能包含姓名、學號、主修科系與平均分數等欄位。

關聯式資料模型將資料庫的資料以有序 n 項來表示，所以學生紀錄可以表現呈有序4元。

以下是一個簡單的資料庫，包含三筆資料：

$(\text{Jason}, 590001, \text{資訊工程}, 3.88)$

$(\text{Uriah}, 590002, \text{資訊工程}, 3.66)$

$(\text{Alex}, 390003, \text{環境工程}, 3.69)$

因為用來表現資料庫的關係經常以表格呈現，所以這些關係也稱為表格。

表格中每一行表示資料庫的一個屬性，見以下表示方式。

| 姓名 | 學號 | 主修科系 | 平均分數 |
|-------|--------|------|------|
| Jason | 590001 | 資訊工程 | 3.88 |
| Uriah | 590002 | 資訊工程 | 3.66 |
| Alex | 390003 | 環境工程 | 3.69 |

n 元關係中稱為主鍵的域，是指有序 n 項的值可由這個域來決定，也就是說關係中沒有兩個不同的有序 n 項在主鍵的值相同。

資料庫中的紀錄經常需要增加或刪除，基於這一點，一個域是否成為主鍵的性質會隨著時間改變。

所以選用的主鍵應該是無論資料庫如何改變都能繼續使用的欄位。

在一個關係中，現有的有序 n 項聚集稱為此關係的擴展。

資料庫較固定的部分，包括資料庫的名子和屬性為其內涵。

當選擇主鍵時，目標是選一個對所有資料庫的擴展都能適用的鍵來作主鍵。

因此我們必須檢驗資料庫的擴展，來了解出現在擴展中有序 n 項集合。

在 n 元關係中，域的組合也可用來唯一地判斷出有序 n 項。

當一組域的值能判斷關係中的有序 n 項時，這些域的笛卡爾積稱為複合鍵。

Example 1

假設以下的表格未來不會再加入有序 n 項，則以下的表格中，哪些域是主鍵？

| 姓名 | 學號 | 主修科系 | 平均分數 |
|------------|--------|------|------|
| Jason Lin | 590001 | 資訊工程 | 3.88 |
| Uriah Xuan | 590002 | 資訊工程 | 3.69 |
| Alex Huang | 390003 | 環境工程 | 3.69 |

表中每個學生的姓名皆是唯一的有序四項，所以姓名是主鍵

學號在表中也是唯一，所以學號也是主鍵。

但主修科系、平均分數不是唯一的，所以兩者皆不是主鍵。

Example 2

假設以下的表格未來不會再加入有序 n 項，則以下的表格中，主修科系和平均分數的笛卡爾積是否為複合鍵？

| 姓名 | 學號 | 主修科系 | 平均分數 |
|------------|--------|------|------|
| Jason Lin | 590001 | 資訊工程 | 3.88 |
| Uriah Xuan | 590002 | 資訊工程 | 3.69 |
| Alex Huang | 390003 | 環境工程 | 3.69 |

因為在表格中沒有兩個有序四項同時有相同的主修科系和平均分數，所以主修科系跟平均分數的笛卡爾積是複合鍵。

Introduce - n元關係上的運算

有許多 n 元關係的運算可用來產生新的 n 元關係。

使用這些運算能回答許多資料庫上滿足某些性質的問題。

n 元關係上最基本的運算是找出滿足某些性質的有序 n 項，見以下的定義。

Definition - 選擇算子

令 R 為一個 n 元關係，而 C 是 R 中元素必須滿足的條件，則選擇算子 s_c 將 n 元關係 R 對應至 n 元關係中滿足條件 C 的所有有序 n 項。

選擇算子可以用在從學生紀錄中，找到主修科系是資訊工程且平均分數大於3.7的紀錄，則可以使用選擇算子來尋找。

Example

| 姓名 | 學號 | 主修科系 | 平均分數 |
|------------|--------|------|------|
| Jason Lin | 590001 | 資訊工程 | 3.88 |
| Uriah Xuan | 590002 | 資訊工程 | 3.59 |
| Alex Huang | 390003 | 環境工程 | 3.79 |

從以上的表格，我們使用算子，其中 C_1 為條件「主修科系」為「資訊工程」，得到兩個有序4項。

要找到平均分數大於3.5的學生，我們用選擇算子，定義 C_2 為條件「平均分數」>「3.7」，同樣得到兩個有序4項。

最後，為了找到主修科系是資訊工程且平均分數大於3.5的紀錄，我們使用算子，其中 C_3 為條件(「主修科系」為資訊工程 \wedge 「平均分數」> 3.7)，結果只剩下一個有序四項。

Definition - 投影

投影是透過刪除關係中每筆紀錄的同一欄位所產生的新 n 元關係，定義如下。

投影 P_{i_1, i_2, \dots, i_m} ，其中 $i_1 < i_2 < \dots < i_m$ 。將有序 n 項 (a_1, a_2, \dots, a_n) 對應到有序 m 項 $a_{i_1}, a_{i_2}, \dots, a_{i_m}$ ，其中 $m \leq n$ 。

也就是投影 P_{i_1, i_2, \dots, i_m} 刪去有序 n 項的 $n - m$ 個分量，留下第 i_1 個分量、第 i_2 個分量、.....和第 i_m 個分量。

Example 1

執行 $P_{1,3}$ 後，下列有序4項 $(2, 3, 0, 4) \cdot (a, b, c, d) \cdot (a_1, a_2, a_3, a_4)$ 會變成什麼？

執行投影 $P_{1,3}$ 後，這些有序4項將變成 $(2, 0) \cdot (a, c) \cdot (a_1, a_3)$ 。

Example 2

當執行投影 $P_{1,2}$ 後，表3的關係將會變成什麼？

| 姓名 | 主修科系 | 課程 |
|-----------|-------|---------|
| Uriah | 資工系 | 離散數學 |
| Uriah | 資工系 | 計算機程式設計 |
| Uriah | 資工系 | 計算機組織 |
| Alex | 環境工程系 | 環境工程導論 |
| Alex | 環境工程系 | 微積分 |
| Christina | 電工系 | 電路學 |
| Christina | 電工系 | 電子學 |
| Christina | 電工系 | 微分方程 |

表3將會變成這樣。

| 姓名 | 主修科系 |
|-----------|-------|
| Uriah | 資工系 |
| Alex | 環境工程系 |
| Christina | 電工系 |

Definition - 連結

當表格有相同的欄位時，連結運算可以用來合併兩個表格，定義如下。

令 R 為一個 m 階關係， S 是一個 n 階關係。

當 $p \leq m$ 且 $p \leq n$ ，連結 $J_p(R, S)$ 是一個 $m + n - p$ 階關係。

包含所有的有序 $(m + n - p)$ 項 $(a_1, a_2, \dots, a_{m-p}, c_1, c_2, \dots, c_p, b_1, b_2, \dots, b_{n-p})$

其中有序 m 項 $(a_1, a_2, \dots, a_{m-p}, c_1, c_2, \dots, c_p)$ 屬於 R ，而有序 n 項 $(c_1, c_2, \dots, c_p, b_1, b_2, \dots, b_{n-p})$ 屬於 S 。

Example

利用連結算子將以下兩個表格連結在一起。

| 教授 | 系別 | 課程代碼 |
|-------|-------|------|
| Uriah | 資訊工程系 | 031 |
| Uriah | 資訊工程系 | 049 |
| Alex | 環境工程系 | 029 |
| Alex | 環境工程系 | 011 |
| Tom | 電子工程系 | 055 |

| 系別 | 課程代碼 | 時間 |
|-------|------|-------|
| 資訊工程系 | 031 | 17:00 |
| 資訊工程系 | 049 | 19:00 |
| 資訊工程系 | 022 | 08:00 |
| 環境工程系 | 029 | 08:00 |
| 環境工程系 | 011 | 16:00 |
| 環境工程系 | 032 | 18:00 |
| 電子工程系 | 055 | 11:00 |

連結的結果如下

| 教授 | 系別 | 課程代碼 | 時間 |
|-------|-------|------|-------|
| Uriah | 資訊工程系 | 031 | 17:00 |
| Uriah | 資訊工程系 | 049 | 19:00 |
| Alex | 環境工程系 | 029 | 08:00 |
| Alex | 環境工程系 | 011 | 16:00 |
| Tom | 電子工程系 | 055 | 11:00 |