集合

School of Computer
Wuhan University

Georg Cantor (1845 — 1918)

不精确的自然语言

Example (不精确的自然语言)

- $P(x)$: x是大学生;
- $Q(x)$: x是好人.

Remark

不精确的自然语言

Example (不精确的自然语言)

- $P(x)$: x是大学生;
- $Q(x)$: x是好人.

Remark

不精确的自然语言

Example (不精确的自然语言)

- $P(x)$ ： x是大学生；
- $Q(x)$ ： x是好人.

Remark

- 用自然语言来表达断言不精确，容易引起二义；

不精确的自然语言

Example (不精确的自然语言)

- $P(x)$ ：x是大学生；
- $Q(x)$ ：x是好人.

Remark

- 用自然语言来表达断言不精确，容易引起二义；
- 原因：面对某个具体的对象x，只能主观地界定x是否具有某断言的所描述的性质.

不精确的自然语言

Example (不精确的自然语言)

- $P(x)$: x是大学生;
- $Q(x)$: x是好人.

Remark

- *用自然语言来表达断言不精确; 容易引起二义;*
- *原因: 面对某个具体的对象x, 只能主观地界定x是否具有某断言的所描述的性质.*

不精确的自然语言

Example (不精确的自然语言)

- $P(x)$: x是大学生;
- $Q(x)$: x是好人.

Remark

- *用自然语言来表达断言不精确; 容易引起二义;*
- *原因: 面对某个具体的对象x, 只能主观地界定x是否具有某断言的所描述的性质.*

精确的集合语言

Example (精确的集合语言)

- 大学生={ 张三, 李四, . . . };
- 好人={ 王五, 赵六, . . . }.

Remark

精确的集合语言

Example (精确的集合语言)

- 大学生={张三, 李四, ...};
- 好人={王五, 赵六, ...}.

Remark

精确的集合语言

Example (精确的集合语言)

- 大学生=$\{$张三, 李四, $\ldots\}$;
- 好人=$\{$王五, 赵六, $\ldots\}$.

Remark

- *用间接的方式表示断言：枚举具有该断言的所有对象；*

精确的集合语言

Example (精确的集合语言)

- 大学生=$\{$张三, 李四, $\ldots\}$;
- 好人=$\{$王五, 赵六, $\ldots\}$.

Remark

- 用间接的方式表示断言：枚举具有该断言的所有对象；
- $P(x)$是否成立，等价于，x是否在对应的集合中；
- 而是否在集合中是可以精确界定的.

精确的集合语言

Example (精确的集合语言)

- 大学生={张三, 李四, ...};
- 好人={王五, 赵六, ...}.

Remark

- *用间接的方式表示断言: 枚举具有该断言的所有对象;*
- *$P(x)$是否成立, 等价于, x是否在对应的集合中;*
- *而是否在集合中是可以精确界定的.*

精确的集合语言

Example (精确的集合语言)

- 大学生=$\{$ 张三, 李四, $\ldots\}$;
- 好人=$\{$ 王五, 赵六, $\ldots\}$.

Remark

- *用间接的方式表示断言: 枚举具有该断言的所有对象;*
- $P(x)$ *是否成立, 等价于,* x *是否在对应的集合中;*
- *而是否在集合中是可以精确界定的.*

精确的集合语言

Example (精确的集合语言)

- 大学生={张三, 李四, ...};
- 好人={王五, 赵六, ...}.

Remark

- *用间接的方式表示断言：枚举具有该断言的所有对象；*
- $P(x)$ *是否成立，等价于，* x *是否在对应的集合中；*
- *而是否在集合中是可以精确界定的.*

集合论

Remark

- *集合论是数学大厦的基础；集合论是表达数学的基础语言；*
- *集合论研究的对象是抽象的集合的关系，进入了二级抽象；A和B具体是什么集合不知道，如同中学的x和y具体是什么数不知道一样；*
- *集合是数学最原始的概念，因此没有定义，只有一组公理来刻画(Zermelo-Fraenkel axioms of set theory (ZF) or ZFC (+ axiom of choice))；*
- *由于集合论是用精确的语言来表达二义的自然语言的，逻辑中的相关规律，在集合论中自然成立；*
- *最重要的Poperty：对于一个具体的集合S和具体的对象x，可以精确地界定x是否属于S；*
- *计算机的许多概念也都是以集合为基础的，如：数据类型，数据库，程序设计语言等，集合是形式化的基础.*

集合论

Remark

- 集合论是数学大厦的基础；集合论是表达数学的基础语言；
- 集合论研究的对象是抽象的集合的关系，进入了二级抽象；A和B具体是什么集合不知道，如同中学的x和y具体是什么数不知道一样；
- 集合是数学最原始的概念，因此没有定义，只有一组公理来刻画(Zermelo-Fraenkel axioms of set theory (ZF) or ZFC (+ axiom of choice))；
- 由于集合论是用精确的语言来表达二义的自然语言的，逻辑中的相关规律，在集合论中自然成立；
- 最重要的Poperty：对于一个具体的集合S和具体的对象x，可以精确地界定x是否属于S；
- 计算机的许多概念也都是以集合为基础的，如：数据类型，数据库，程序设计语言等，集合是形式化的基础.

集合论

Remark

- 集合论是数学大厦的基础；集合论是表达数学的基础语言；
- 集合论研究的对象是抽象的集合的关系，进入了二级抽象；A和B具体是什么集合不知道，如同中学的x和y具体是什么数不知道一样；
- 集合是数学最原始的概念，因此没有定义，只有一组公理来刻画(Zermelo-Fraenkel axioms of set theory (ZF) or ZFC (+ axiom of choice))；
- 由于集合论是用精确的语言来表达二义的自然语言的，逻辑中的相关规律，在集合论中自然成立；
- 最重要的Poperty：对于一个具体的集合S和具体的对象x，可以精确地界定x是否属于S；
- 计算机的许多概念也都是以集合为基础的，如：数据类型，数据库，程序设计语言等，集合是形式化的基础.

集合论

Remark

- 集合论是数学大厦的基础；集合论是表达数学的基础语言；
- 集合论研究的对象是抽象的集合的关系，进入了二级抽象；A和B具体是什么集合不知道，如同中学的x和y具体是什么数不知道一样；
- 集合是数学最原始的概念，因此没有定义，只有一组公理来刻画(Zermelo-Fraenkel axioms of set theory (ZF) or ZFC (+ axiom of choice))；
- 由于集合论是用精确的语言来表达二义的自然语言的，逻辑中的相关规律，在集合论中自然成立；
- 最重要的Poperty：对于一个具体的集合S和具体的对象x，可以精确地界定x是否属于S；
- 计算机的许多概念也都是以集合为基础的，如：数据类型，数据库，程序设计语言等，集合是形式化的基础.

集合论

Remark

- 集合论是数学大厦的基础；集合论是表达数学的基础语言；
- 集合论研究的对象是抽象的集合的关系，进入了二级抽象；A和B具体是什么集合不知道，如同中学的x和y具体是什么数不知道一样；
- 集合是数学最原始的概念，因此没有定义，只有一组公理来刻画(Zermelo-Fraenkel axioms of set theory (ZF) or ZFC (+ axiom of choice))；
- 由于集合论是用精确的语言来表达二义的自然语言的，逻辑中的相关规律，在集合论中自然成立；
- 最重要的Poperty：对于一个具体的集合S和具体的对象x，可以精确地界定x是否属于S；
- 计算机的许多概念也都是以集合为基础的，如：数据类型，数据库，程序设计语言等，集合是形式化的基础.

集合论

Remark

- *集合论是数学大厦的基础；集合论是表达数学的基础语言；*
- *集合论研究的对象是抽象的集合的关系，进入了二级抽象；A和B具体是什么集合不知道，如同中学的x和y具体是什么数不知道一样；*
- *集合是数学最原始的概念，因此没有定义，只有一组公理来刻画(Zermelo-Fraenkel axioms of set theory (ZF) or ZFC (+ axiom of choice))；*
- *由于集合论是用精确的语言来表达二义的自然语言的，逻辑中的相关规律，在集合论中自然成立；*
- *最重要的Poperty：对于一个具体的集合S和具体的对象x，可以精确地界定x是否属于S；*
- *计算机的许多概念也都是以集合为基础的，如：数据类型，数据库，程序设计语言等，集合是形式化的基础.*

集合论

Remark

- 集合论是数学大厦的基础；集合论是表达数学的基础语言；
- 集合论研究的对象是抽象的集合的关系，进入了二级抽象；A和B具体是什么集合不知道，如同中学的x和y具体是什么数不知道一样；
- 集合是数学最原始的概念，因此没有定义，只有一组公理来刻画(Zermelo-Fraenkel axioms of set theory (ZF) or ZFC (+ axiom of choice))；
- 由于集合论是用精确的语言来表达二义的自然语言的，逻辑中的相关规律，在集合论中自然成立；
- 最重要的Poperty：对于一个具体的集合S和具体的对象x，可以精确地界定x是否属于S；
- 计算机的许多概念也都是以集合为基础的，如：数据类型，数据库，程序设计语言等，集合是形式化的基础.

集合

Description

直观上说：把满足某个具体谓词$P(x)$(取真值)的对象全体称为集合，用大写字母S, R, T, ...表示.

Example

集合

Description

直观上说：把满足某个具体谓词$P(x)$(取真值)的对象全体称为集合，用大写字母S, R, T, $\ldots$表示.

Example

- $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F$命题公式$\}$;
- $\{s \mid s$是由ASCII字母组成的字符串$\}$;
- $\{P \mid P$是C语言程序$\}$;
- $\{x \mid x$是MyClass对象的实例$\}$.

集合

Description

*直观上说：把满足某个具体谓词$P(x)$(取真值)的对象全体称为***集合**，用大写字母S, R, T, $\ldots$表示.

Example

- $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F$命题公式$\}$;
- $\{s \mid s$是由ASCII字母组成的字符串$\}$;
- $\{P \mid P$是C语言程序$\}$;
- $\{x \mid x$是MyClass对象的实例$\}$.

集合

Description

直观上说：把满足某个具体谓词$P(x)$(取真值)的对象全体称为集合，用大写字母$S, R, T, \ldots$表示.

Example

- $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F$命题公式$\}$;
- $\{s \mid s$是由ASCII字母组成的字符串$\}$;
- $\{P \mid P$是C语言程序$\}$;
- $\{x \mid x$是MyClass对象的实例$\}$.

集合

Description

直观上说：把满足某个具体谓词$P(x)$(取真值)的对象全体称为集合，用大写字母S, R, T, ...表示.

Example

- $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F$命题公式$\}$;
- $\{s \mid s$是由ASCII字母组成的字符串$\}$;
- $\{P \mid P$是C语言程序$\}$;
- $\{x \mid x$是MyClass对象的实例$\}$.

集合

Description

直观上说：把满足某个具体谓词$P(x)$(取真值)的对象全体称为集合，用大写字母S, R, T, $\ldots$表示.

Example

- $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F$命题公式$\}$;
- $\{s \mid s$是由ASCII字母组成的字符串$\}$;
- $\{P \mid P$是C语言程序$\}$;
- $\{x \mid x$是MyClass对象的实例$\}$.

集合

Description

直观上说：把满足某个具体谓词$P(x)$(取真值)的对象全体称为集合，用大写字母S, R, T, ...表示.

Example

- $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{命题公式}\}$;
- $\{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $\{P \mid P\text{是C语言程序}\}$;
- $\{x \mid x\text{是MyClass对象的实例}\}$.

元素

Description

集合的成员称为元素，一般用小写字母表示：

$$s\text{是集合}S\text{的元素} \triangleq s \in S$$

$$s\text{不是集合}S\text{的元素} \triangleq s \notin S$$

Property

设S是集合，对任意的一元素x，则：或者$x \in S$，或者$x \notin S$，并且，两则不能同时为真.

Example

元素

Description

集合的成员称为元素，一般用小写字母表示：

$$s\text{是集合}S\text{的元素} \triangleq s \in S$$

$$s\text{不是集合}S\text{的元素} \triangleq s \notin S$$

Property

设S是集合，对任意的一元素x，则：或者$x \in S$，或者$x \notin S$，并且，两则不能同时为真.

Example

元素

Description

集合的成员称为元素，一般用小写字母表示：

$$s\text{是集合}S\text{的元素} \triangleq s \in S$$

$$s\text{不是集合}S\text{的元素} \triangleq s \notin S$$

Property

设S是集合，对任意的一元素x，则：或者$x \in S$，或者$x \notin S$，并且，两则不能同时为真.

Example

- $\neg \in \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $P \rightarrow Q \in \{F \mid F\text{命题公式}\}$;
- $abc \in \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $abc \notin \{P \mid P\text{是C语言程序}\}$.

元素

Description

集合的成员称为元素，一般用小写字母表示：

$$s\text{是集合}S\text{的元素} \triangleq s \in S$$

$$s\text{不是集合}S\text{的元素} \triangleq s \notin S$$

Property

设S是集合，对任意的一元素x，则：或者$x \in S$，或者$x \notin S$，并且，两则不能同时为真.

Example

- $\neg \in \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $P \rightarrow Q \in \{F \mid F\text{命题公式}\}$;
- $abc \in \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $abc \notin \{P \mid P\text{是C语言程序}\}$.

元素

Description

集合的成员称为元素，一般用小写字母表示：

$$s\text{是集合}S\text{的元素} \triangleq s \in S$$
$$s\text{不是集合}S\text{的元素} \triangleq s \notin S$$

Property

设S是集合，对任意的一元素x，则：或者$x \in S$，或者$x \notin S$，并且，两则不能同时为真.

Example

- $\neg \in \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $P \rightarrow Q \in \{F \mid F\text{命题公式}\}$;
- $abc \in \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $abc \notin \{P \mid P\text{是C语言程序}\}$.

元素

Description

集合的成员称为元素，一般用小写字母表示：

$$s\text{是集合}S\text{的元素} \triangleq s \in S$$

$$s\text{不是集合}S\text{的元素} \triangleq s \notin S$$

Property

设S是集合，对任意的一元素x，则：或者$x \in S$，或者$x \notin S$，并且，两则不能同时为真.

Example

- $\neg \in \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $P \rightarrow Q \in \{F \mid F\text{命题公式}\}$;
- $abc \in \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $abc \notin \{P \mid P\text{是C语言程序}\}$.

元素

Description

集合的成员称为元素，一般用小写字母表示：

$$s\text{是集合}S\text{的元素} \triangleq s \in S$$

$$s\text{不是集合}S\text{的元素} \triangleq s \notin S$$

Property

设S是集合，对任意的一元素x，则：或者$x \in S$，或者$x \notin S$，并且，两则不能同时为真.

Example

- $\neg \in \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $P \rightarrow Q \in \{F \mid F\text{命题公式}\}$;
- $abc \in \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $abc \notin \{P \mid P\text{是C语言程序}\}$.

集合的表示方法

Description (表示法)

- *枚举法：有限元素或有规律的无限元素；*
  - $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- *描述法：*$\{\, x \mid P(x) \,\}$*，注意*P*一定不能模糊：*

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{\, x \mid P(x) \,\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法: 有限元素或有规律的无限元素;*
  - $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法: $\{x \mid P(x)\}$, 注意P一定不能模糊;*

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法*: *有限元素或有规律的无限元素;*
  - $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法*: $\{x \mid P(x)\}$, *注意P一定不能模糊:*

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法*: *有限元素或有规律的无限元素;*
  - $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法*: $\{x \mid P(x)\}$, *注意P一定不能模糊:*

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法*: *有限元素或有规律的无限元素*;
  - $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法*: $\{x \mid P(x)\}$, *注意*P*一定不能模糊*:
  - $\{F \mid F\text{命题公式}\}$;
  - $\{P \mid P\text{是}C\text{语言程序}\}$;
  - $\{x \mid x\text{是整数并且是}2\text{的倍数}\}$;
  - $\mathbf{1} = \{\langle x_n\rangle \mid \langle x_n\rangle\text{是有理数序列，并且和}1\text{有相同的极限}\}$.

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法*: *有限元素或有规律的无限元素*;
  - $\{\neg, \to, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法*: $\{x \mid P(x)\}$, *注意*P*一定不能模糊*:
  - $\{F \mid F$ 命题公式$\}$;
  - $\{P \mid P$ 是C语言程序$\}$;
  - $\{x \mid x$ 是整数并且是2的倍数$\}$;
  - $\mathbf{1} = \{\langle x_n\rangle \mid \langle x_n\rangle$ 是有理数序列，并且和1有相同的极限$\}$.

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法*: *有限元素或有规律的无限元素*;
  - $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法*: $\{x \mid P(x)\}$, *注意*P*一定不能模糊*:
  - $\{F \mid F\text{命题公式}\}$;
  - $\{P \mid P\text{是C语言程序}\}$;
  - $\{x \mid x\text{是整数并且是2的倍数}\}$;
  - $\mathbf{1} = \{\langle x_n \rangle \mid \langle x_n \rangle\text{是有理数序列，并且和1有相同的极限}\}$.

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- 枚举法：有限元素或有规律的无限元素；
  - $\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- 描述法：$\{x \mid P(x)\}$，注意P一定不能模糊：
  - $\{F \mid F$命题公式$\}$;
  - $\{P \mid P$是C语言程序$\}$;
  - $\{x \mid x$是整数并且是2的倍数$\}$;
  - $\mathbf{1} = \{\langle x_n \rangle \mid \langle x_n \rangle$是有理数序列，并且和1有相同的极限$\}$.

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法*: 有限元素或有规律的无限元素;
  - $\{\neg, \to, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法*: $\{x \mid P(x)\}$, 注意P一定不能模糊:
  - $\{F \mid F\text{命题公式}\}$;
  - $\{P \mid P\text{是C语言程序}\}$;
  - $\{x \mid x\text{是整数并且是2的倍数}\}$;
  - $\mathbf{1} = \{\langle x_n \rangle \mid \langle x_n \rangle\text{是有理数序列, 并且和1有相同的极限}\}$.

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

集合的表示方法

Description (表示法)

- *枚举法*: 有限元素或有规律的无限元素;
  - $\{\neg, \to, \wedge, \vee, \leftrightarrow\}$;
  - $\{\ldots, -4, -2, 0, 2, 4, \ldots\}$;
- *描述法*: $\{x \mid P(x)\}$, 注意P一定不能模糊:
  - $\{F \mid F\text{命题公式}\}$;
  - $\{P \mid P\text{是C语言程序}\}$;
  - $\{x \mid x\text{是整数并且是2的倍数}\}$;
  - $\mathbf{1} = \{\langle x_n\rangle \mid \langle x_n\rangle\text{是有理数序列, 并且和1有相同的极限}\}$.

Remark (集合和谓词的关系)

$$P(x) \rightsquigarrow S_P = \{x \mid P(x)\}$$
$$S \rightsquigarrow P_S(x) = x \in S$$

相关概念

Definition

- 单点集(Singleton): $\{a\}$
- 有限集合(finite set): 元素的个数有限;
- 无限集合(infinite set): 元素的个数无限;
- 集合中所有元素的个数称为该集合的基数(Cardinal), 记为: $|S|$:
  - $|\{\neg, \to, \wedge, \vee, \leftrightarrow\}| = 5$
  - $|\mathbb{N}| = \infty$

相关概念

Definition

- 单点集(Singleton): $\{a\}$
- 有限集合(finite set): 元素的个数有限;
- 无限集合(infinite set): 元素的个数无限;
- 集合中所有元素的个数称为该集合的基数(Cardinal), 记为: $|S|$:

相关概念

Definition

- 单点集(Singleton): $\{a\}$
- 有限集合(finite set): 元素的个数有限;
- 无限集合(infinite set): 元素的个数无限;
- 集合中所有元素的个数称为该集合的基数(Cardinal), 记为: $|S|$:
  - $|\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}| = 5$
  - $|\mathbb{N}| = \infty$

相关概念

Definition

- 单点集(Singleton): $\{a\}$
- 有限集合(finite set): 元素的个数有限;
- 无限集合(infinite set): 元素的个数无限;
- 集合中所有元素的个数称为该集合的基数(Cardinal), 记为: $|S|$:
    - $|\{\neg, \to, \wedge, \vee, \leftrightarrow\}| = 5$
    - $|\mathbb{N}| = \infty$

相关概念

Definition

- 单点集(Singleton): $\{a\}$
- 有限集合(finite set): 元素的个数有限;
- 无限集合(infinite set): 元素的个数无限;
- 集合中所有元素的个数称为该集合的基数(Cardinal), 记为: $|S|$:
  - $|\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}| = 5$
  - $|\mathbb{N}| = \infty$

相关概念

Definition

- 单点集(Singleton): $\{a\}$
- 有限集合(finite set): 元素的个数有限;
- 无限集合(infinite set): 元素的个数无限;
- 集合中所有元素的个数称为该集合的基数(Cardinal), 记为: $|S|$:
  - $|\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}| = 5$
  - $|\mathbb{N}| = \infty$

相关概念

Definition

- 单点集(Singleton): $\{a\}$
- 有限集合(finite set): 元素的个数有限;
- 无限集合(infinite set): 元素的个数无限;
- 集合中所有元素的个数称为该集合的基数(Cardinal), 记为: $|S|$:
  - $|\{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}| = 5$
  - $|\mathbb{N}| = \infty$

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements.
记为：$A = B$.

Corollary

- *枚举法表示与元素出现的次序无关：$\{1,2,3\} = \{3,2,1\}$；*
  枚举法表示与元素是否重复出现无关：
  $\{1,2,3\} = \{3,3,1,2,1\}$；
  集合的表示方法不唯一.

Remark (证明两个集合相等)

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements.
记为: $A = B$.

Corollary

- 枚举法表示与元素出现的次序无关: $\{1, 2, 3\} = \{3, 2, 1\}$;
- 枚举法表示与元素是否重复出现无关: $\{1, 2, 3\} = \{3, 3, 1, 2, 1\}$;
- 集合的表示方法不唯一.

Remark (证明两个集合相等)

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements. 记为：$A = B$.

Corollary

- 枚举法表示与元素出现的次序无关：$\{1, 2, 3\} = \{3, 2, 1\}$;
- 枚举法表示与元素是否重复出现无关：$\{1, 2, 3\} = \{3, 3, 1, 2, 1\}$;
- 集合的表示方法不唯一.

Remark (证明两个集合相等)

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements.
记为：$A = B$.

Corollary

- 枚举法表示与元素出现的次序无关：$\{1, 2, 3\} = \{3, 2, 1\}$;
- 枚举法表示与元素是否重复出现无关：$\{1, 2, 3\} = \{3, 3, 1, 2, 1\}$;
- 集合的表示方法不唯一.

Remark (证明两个集合相等)

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements.
记为: $A = B$.

Corollary

- *枚举法表示与元素出现的次序无关*: $\{1, 2, 3\} = \{3, 2, 1\}$;
- *枚举法表示与元素是否重复出现无关*: $\{1, 2, 3\} = \{3, 3, 1, 2, 1\}$;
- *集合的表示方法不唯一*.

Remark (证明两个集合相等)

- $\forall x(x \in A \to x \in B)$

  $\forall x(x \in B \to x \in A)$

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements. 记为: $A = B$.

Corollary

- 枚举法表示与元素出现的次序无关: $\{1, 2, 3\} = \{3, 2, 1\}$;
- 枚举法表示与元素是否重复出现无关: $\{1, 2, 3\} = \{3, 3, 1, 2, 1\}$;
- 集合的表示方法不唯一.

Remark (证明两个集合相等)

- $\forall x (x \in A \to x \in B)$
- $\forall x (x \in B \to x \in A)$

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements.
记为： $A = B$.

Corollary

- *枚举法表示与元素出现的次序无关：* $\{1, 2, 3\} = \{3, 2, 1\}$;
- *枚举法表示与元素是否重复出现无关：* $\{1, 2, 3\} = \{3, 3, 1, 2, 1\}$;
- *集合的表示方法不唯一.*

Remark (证明两个集合相等)

- $\forall x(x \in A \to x \in B)$
- $\forall x(x \in B \to x \in A)$

外延公理

Axiom (外延公理(Axiom of extensionality))

Two sets are the same if and only if they have the same elements.
记为: $A = B$.

Corollary

- 枚举法表示与元素出现的次序无关: $\{1, 2, 3\} = \{3, 2, 1\}$;
- 枚举法表示与元素是否重复出现无关: $\{1, 2, 3\} = \{3, 3, 1, 2, 1\}$;
- 集合的表示方法不唯一.

Remark (证明两个集合相等)

- $\forall x(x \in A \to x \in B)$
- $\forall x(x \in B \to x \in A)$

集合的包含关系

Definition (包含关系(Inclusion))

设A和B是两集合，A是B的子集合(Subset)或称B是A的父集合(Superset), iff, $\forall x(x \in A \rightarrow x \in B)$，记为：$A \subseteq B$, or $B \supseteq A$.

Remark (对应的逻辑关系)

$$A \subseteq B \quad \rightsquigarrow \quad \forall x(P_A(x) \rightarrow P_B(x))\text{为真}$$

Example

集合的包含关系

Definition (包含关系(Inclusion))

设A和B是两集合，A是B的子集合(Subset)或称B是A的父集合(Superset), iff, $\forall x(x \in A \rightarrow x \in B)$，记为：$A \subseteq B$, or $B \supseteq A$.

Remark (对应的逻辑关系)

$$A \subseteq B \quad \leftrightsquigarrow \quad \forall x(P_A(x) \rightarrow P_B(x))\text{为真}$$

Example

集合的包含关系

Definition (包含关系(Inclusion))

设A和B是两集合，A是B的子集合(Subset)或称B是A的父集合(Superset), iff, $\forall x(x \in A \to x \in B)$，记为：$A \subseteq B$, or $B \supseteq A$.

Remark (对应的逻辑关系)

$$A \subseteq B \quad \leftrightsquigarrow \quad \forall x(P_A(x) \to P_B(x))\text{为真}$$

Example

- $\{\neg, \wedge, \vee, \} \subseteq \{\neg, \to, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subseteq \{F \mid F\text{是可满足公式}\}$;
- $\{P \mid P\text{是C语言程序}\} \subseteq \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $Subclass \subseteq Class$.

集合的包含关系

Definition (包含关系(Inclusion))

设A和B是两集合，A是B的子集合(Subset)或称B是A的父集合(Superset), iff, $\forall x(x \in A \rightarrow x \in B)$，记为：$A \subseteq B$, or $B \supseteq A$.

Remark (对应的逻辑关系)

$$A \subseteq B \quad \leftrightsquigarrow \quad \forall x(P_A(x) \rightarrow P_B(x))\text{为真}$$

Example

- $\{\neg, \wedge, \vee, \} \subseteq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subseteq \{F \mid F\text{是可满足公式}\}$;
- $\{P \mid P\text{是C语言程序}\} \subseteq \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $Subclass \subseteq Class$.

集合的包含关系

Definition (包含关系(Inclusion))

设A和B是两集合，A是B的子集合(Subset)或称B是A的父集合(Superset), iff, $\forall x(x \in A \to x \in B)$, 记为：$A \subseteq B$, or $B \supseteq A$.

Remark (对应的逻辑关系)

$$A \subseteq B \quad \leftrightsquigarrow \quad \forall x(P_A(x) \to P_B(x))\text{为真}$$

Example

- $\{\neg, \wedge, \vee, \} \subseteq \{\neg, \to, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subseteq \{F \mid F\text{是可满足公式}\}$;
- $\{P \mid P\text{是C语言程序}\} \subseteq \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $Subclass \subseteq Class$.

集合的包含关系

Definition (包含关系(Inclusion))

设A和B是两集合，A是B的子集合(Subset)或称B是A的父集合(Superset), iff, $\forall x(x \in A \rightarrow x \in B)$, 记为：$A \subseteq B$, or $B \supseteq A$.

Remark (对应的逻辑关系)

$$A \subseteq B \quad \leftrightsquigarrow \quad \forall x(P_A(x) \rightarrow P_B(x))\text{为真}$$

Example

- $\{\neg, \wedge, \vee,\} \subseteq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subseteq \{F \mid F\text{是可满足公式}\}$;
- $\{P \mid P\text{是C语言程序}\} \subseteq \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $Subclass \subseteq Class$.

集合的包含关系

Definition (包含关系(Inclusion))

设A和B是两集合，A是B的子集合(Subset)或称B是A的父集合(Superset), iff, $\forall x(x \in A \rightarrow x \in B)$，记为：$A \subseteq B$, or $B \supseteq A$.

Remark (对应的逻辑关系)

$$A \subseteq B \quad \leftrightsquigarrow \quad \forall x(P_A(x) \rightarrow P_B(x))\text{为真}$$

Example

- $\{\neg, \wedge, \vee, \} \subseteq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subseteq \{F \mid F\text{是可满足公式}\}$;
- $\{P \mid P\text{是C语言程序}\} \subseteq \{s \mid s\text{是由ASCII字母组成的字符串}\}$;
- $Subclass \subseteq Class$.

集合的包含关系

Definition (真包含关系(Proper Inclusion))

设A和B是两集合，A是B的真子集合(Subset)或称B是A的真父集合(Superset), iff, $A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$, 记为：$A \subsetneq B$ $(A \subset B)$, or $B \supsetneq A$ $(B \supset A)$.

Example

Remark ($\subseteq$和$\in$的不同)

集合的包含关系

Definition (真包含关系(Proper Inclusion))

设A和B是两集合，A是B的真子集合(Subset)或称B是A的真父集合(Superset)，iff, $A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$，记为：$A \subsetneq B$ ($A \subset B$), or $B \supsetneq A$ ($B \supset A$).

Example

- $\{\neg, \wedge, \vee\} \subsetneq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subsetneq \{F \mid F\text{是可满足公式}\}$.

Remark ($\subseteq$和$\in$的不同)

集合的包含关系

Definition (真包含关系(Proper Inclusion))

设A和B是两集合，A是B的真子集合(Subset)或称B是A的真父集合(Superset), iff, $A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$, 记为：$A \subsetneq B$ ($A \subset B$), or $B \supsetneq A$ ($B \supset A$).

Example

- $\{\neg, \wedge, \vee\} \subsetneq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subsetneq \{F \mid F\text{是可满足公式}\}$.

Remark ($\subseteq$和$\in$的不同)

集合的包含关系

Definition (真包含关系(Proper Inclusion))

设A和B是两集合，A是B的真子集合(Subset)或称B是A的真父集合(Superset), iff, $A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$, 记为：$A \subsetneq B$ ($A \subset B$), or $B \supsetneq A$ ($B \supset A$).

Example

- $\{\neg, \wedge, \vee\} \subsetneq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subsetneq \{F \mid F\text{是可满足公式}\}$.

Remark (⊆和∈的不同)

- 层次不同，∈是不同层次的关系，而⊆是同层次的关系；

集合的包含关系

Definition (真包含关系(Proper Inclusion))

设A和B是两集合，A是B的真子集合(Subset)或称B是A的真父集合(Superset), iff, $A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$，记为：$A \subsetneq B$ ($A \subset B$), or $B \supsetneq A$ ($B \supset A$).

Example

- $\{\neg, \wedge, \vee\} \subsetneq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subsetneq \{F \mid F\text{是可满足公式}\}$.

Remark ($\subseteq$和$\in$的不同)

- 层次不同，$\in$是不同层次的关系，而$\subseteq$是同层次的关系；
- 但是，可能会同时成立，if集合中有不同层次的对象时：
ex: $A = \{1, 2\}$, $B = \{1, 2, 3, \{1, 2\}\}$, 则，$A \in B \wedge A \subsetneq B$

集合的包含关系

Definition (真包含关系(Proper Inclusion))

设A和B是两集合，A是B的真子集合(Subset)或称B是A的真父集合(Superset), iff, $A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$, 记为：$A \subsetneq B$ $(A \subset B)$, or $B \supsetneq A$ $(B \supset A)$.

Example

- $\{\neg, \wedge, \vee\} \subsetneq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subsetneq \{F \mid F\text{是可满足公式}\}$.

Remark ($\subseteq$和$\in$的不同)

- *层次不同，$\in$是不同层次的关系，而$\subseteq$是同层次的关系；*
- *但是，可能会同时成立，if集合中有不同层次的对象时：ex: $A = \{1, 2\}$, $B = \{1, 2, 3, \{1, 2\}\}$, 则，$A \in B \wedge A \subsetneq B$*

集合的包含关系

Definition (真包含关系(Proper Inclusion))

设A和B是两集合，A是B的真子集合(Subset)或称B是A的真父集合(Superset)，iff，$A \subseteq B \wedge \exists x(x \in B \wedge x \notin A)$，记为：$A \subsetneq B$ $(A \subset B)$, or $B \supsetneq A$ $(B \supset A)$.

Example

- $\{\neg, \wedge, \vee\} \subsetneq \{\neg, \rightarrow, \wedge, \vee, \leftrightarrow\}$;
- $\{F \mid F\text{重言式}\} \subsetneq \{F \mid F\text{是可满足公式}\}$.

Remark (⊆和∈的不同)

- *层次不同，∈是不同层次的关系，而⊆是同层次的关系；*
- *但是，可能会同时成立，if集合中有不同层次的对象时：*
  ex: $A = \{1, 2\}$, $B = \{1, 2, 3, \{1, 2\}\}$，则，$A \in B \wedge A \subsetneq B$

相等

Theorem

$$A = B \iff A \subseteq B \wedge B \subseteq A$$

相等

Theorem

$$A = B \quad \Longleftrightarrow \quad A \subseteq B \wedge B \subseteq A$$

Proof.

$$
\begin{aligned}
& A = B \\
\Longleftrightarrow\ & \forall x((x \in A \to x \in B) \wedge (x \in B \to x \in A)) && \text{(by Axiom)} \\
\Longleftrightarrow\ & \forall x(x \in A \to x \in B) \wedge \forall x(x \in B \to x \in A) && \text{(by 量词分配律)} \\
\Longleftrightarrow\ & (A \subseteq B) \wedge (B \subseteq A) && \text{(by def)}
\end{aligned}
$$

□

Corollary

相等

Theorem

$$A = B \iff A \subseteq B \land B \subseteq A$$

Proof.

$$
\begin{aligned}
& A = B \\
\iff & \forall x((x \in A \to x \in B) \land (x \in B \to x \in A)) && \text{(by Axiom)} \\
\iff & \forall x(x \in A \to x \in B) \land \forall x(x \in B \to x \in A) && \text{(by 量词分配律)} \\
\iff & (A \subseteq B) \land (B \subseteq A) && \text{(by def)}
\end{aligned}
$$

□

Corollary

- $A \subseteq A$
- If $A \subseteq B \land B \subseteq C$, then $A \subseteq C$

相等

Theorem

$$A = B \iff A \subseteq B \land B \subseteq A$$

Proof.

$$\begin{aligned}
& A = B \\
\iff & \forall x((x \in A \to x \in B) \land (x \in B \to x \in A)) && \text{(by Axiom)} \\
\iff & \forall x(x \in A \to x \in B) \land \forall x(x \in B \to x \in A) && \text{(by 量词分配律)} \\
\iff & (A \subseteq B) \land (B \subseteq A) && \text{(by def)}
\end{aligned}$$

$\square$

Corollary

- $A \subseteq A$
- if $A \subseteq B \land B \subseteq C$, then $A \subseteq C$

相等

Theorem

$$A = B \iff A \subseteq B \land B \subseteq A$$

Proof.

$$
\begin{aligned}
& A = B \\
\iff & \forall x((x \in A \to x \in B) \land (x \in B \to x \in A)) && \text{(by Axiom)} \\
\iff & \forall x(x \in A \to x \in B) \land \forall x(x \in B \to x \in A) && \text{(by 量词分配律)} \\
\iff & (A \subseteq B) \land (B \subseteq A) && \text{(by def)}
\end{aligned}
$$

$\square$

Corollary

- $A \subseteq A$
- *if* $A \subseteq B \land B \subseteq C$, *then* $A \subseteq C$

相等

Theorem

$$A = B \iff A \subseteq B \wedge B \subseteq A$$

Proof.

$$\begin{aligned}
& A = B \\
\iff & \forall x((x \in A \to x \in B) \wedge (x \in B \to x \in A)) && \text{(by Axiom)} \\
\iff & \forall x(x \in A \to x \in B) \wedge \forall x(x \in B \to x \in A) && \text{(by 量词分配律)} \\
\iff & (A \subseteq B) \wedge (B \subseteq A) && \text{(by def)}
\end{aligned}$$

$\square$

Corollary

- $A \subseteq A$
- *if* $A \subseteq B \wedge B \subseteq C$, *then* $A \subseteq C$

Univers & Empty Set(1/2)

Description (Univers)

限定讨论对象的为某一论述域的集合，它包含了全部的讨论对象，但是不能包含自身(否则会引起悖论)，称这样的集合为全集合，记为$\mathcal{U}$.

Property

$\forall A$ a set, then $A \subseteq \mathcal{U}$;

Axiom (of Empty Set)

There is a set with no elements. We will use $\emptyset$ to denote this empty set; So, $\forall x(x \notin \emptyset)$.

Property

$\forall A$ a set, then $\emptyset \subseteq A$ ($\because$前提是假的，$\therefore \forall x(x \in \emptyset \to x \in A)$ 成立).

Univers & Empty Set(1/2)

Description (Univers)

限定讨论对象的为某一论述域的集合，它包含了全部的讨论对象，但是不能包含自身(否则会引起悖论)，称这样的集合为全集合，记为$\mathcal{U}$.

Property

$\forall A$ *a set, then* $A \subseteq \mathcal{U}$;

Axiom (of Empty Set)

There is a set with no elements. We will use $\varnothing$ *to denote this empty set; So,* $\forall x(x \notin \varnothing)$.

Property

$\forall A$ *a set, then* $\varnothing \subseteq A$ $(\because$*前提是假的*$, \therefore \forall x(x \in \varnothing \rightarrow x \in A)$ *成立*$)$.

Univers & Empty Set(1/2)

Description (Univers)

限定讨论对象的为某一论述域的集合，它包含了全部的讨论对象，但是不能包含自身（否则会引起悖论），称这样的集合为全集合，记为$\mathcal{U}$.

Property

$\forall A$ *a set, then* $A \subseteq \mathcal{U}$;

Axiom (of Empty Set)

There is a set with no elements. We will use $\emptyset$ *to denote this empty set; So,* $\forall x(x \notin \emptyset)$.

Property

$\forall A$ *a set, then* $\emptyset \subseteq A$ ($\because$前提是假的，$\therefore \forall x(x \in \emptyset \to x \in A)$ 成立).

Univers & Empty Set(1/2)

Description (Univers)

限定讨论对象的为某一论述域的集合，它包含了全部的讨论对象，但是不能包含自身(否则会引起悖论)，称这样的集合为全集合，记为$\mathcal{U}$.

Property

$\forall A$ *a set, then* $A \subseteq \mathcal{U}$;

Axiom (of Empty Set)

There is a set with no elements. We will use $\varnothing$ *to denote this empty set; So,* $\forall x(x \notin \varnothing)$.

Property

$\forall A$ *a set, then* $\varnothing \subseteq A$ ($\because$前提是假的，$\therefore \forall x(x \in \varnothing \to x \in A)$ 成立).

Univers & Empty set(2/2)

Theorem

$\emptyset$是唯一的，因为，设另有一空集$\emptyset'$，则，$\emptyset \subseteq \emptyset' \wedge \emptyset' \subseteq \emptyset$，$\therefore \emptyset = \emptyset'$.

Example

- $\{a, b\}$的子集合有$\emptyset$, $\{a, b\}$, $\{a\}$, $\{b\}$, 其中$\emptyset$ 和$\{a, b\}$称为平凡子集合(trivial);
- *空集本身可以是其他集合的元素，如*
  $\emptyset \subseteq \{\emptyset\} \subseteq \{\emptyset, \{\emptyset\}\}$;
- $|\emptyset| = 0$.

Remark

$\mathcal{U}$ just likes $\mathbb{T}$, and $\emptyset$ likes $\mathbb{F}$ of logic.

Univers & Empty set(2/2)

Theorem

$\varnothing$是唯一的，因为，设另有一空集$\varnothing'$，则，$\varnothing \subseteq \varnothing' \wedge \varnothing' \subseteq \varnothing$，
$\therefore\ \varnothing = \varnothing'$.

Example

- $\{a, b\}$的子集合有$\varnothing$, $\{a, b\}$, $\{a\}$, $\{b\}$，其中$\varnothing$ 和$\{a, b\}$称为平凡子集合(trivial);
- 空集本身可以是其他集合的元素，如:
  $\varnothing \subsetneq \{\varnothing\} \subsetneq \{\varnothing, \{\varnothing\}\}$;
- $|\varnothing| = 0$.

Remark

$\mathcal{U}$ just likes $\mathbb{T}$, and $\varnothing$ likes $\mathbb{F}$ of logic.

Univers & Empty set(2/2)

Theorem

$\varnothing$是唯一的，因为，设另有一空集$\varnothing'$，则，$\varnothing \subseteq \varnothing' \land \varnothing' \subseteq \varnothing$，$\therefore \varnothing = \varnothing'$.

Example

- $\{a, b\}$的子集合有$\varnothing$，$\{a, b\}$，$\{a\}$，$\{b\}$，其中$\varnothing$ 和$\{a, b\}$称为平凡子集合(trivial);
- 空集本身可以是其他集合的元素，如：$\varnothing \subsetneq \{\varnothing\} \subsetneq \{\varnothing, \{\varnothing\}\}$;
- $|\varnothing| = 0$.

Remark

$\mathcal{U}$ just likes $\mathbb{T}$, and $\varnothing$ likes $\mathbb{F}$ of logic.

Univers & Empty set(2/2)

Theorem

$\varnothing$是唯一的，因为，设另有一空集$\varnothing\prime$，则，$\varnothing \subseteq \varnothing\prime \wedge \varnothing\prime \subseteq \varnothing$，
$\therefore \varnothing = \varnothing\prime$.

Example

- $\{a, b\}$的子集合有$\varnothing$，$\{a, b\}$，$\{a\}$，$\{b\}$，其中$\varnothing$ 和$\{a, b\}$称为平凡子集合(trivial);
- 空集本身可以是其他集合的元素，如：
  $\varnothing \subsetneq \{\varnothing\} \subsetneq \{\varnothing, \{\varnothing\}\}$;
- $|\varnothing| = 0$.

Remark

$\mathcal{U}$ just likes $\mathbb{T}$, and $\varnothing$ likes $\mathbb{F}$ of logic.

Univers & Empty set(2/2)

Theorem

$\emptyset$是唯一的，因为，设另有一空集$\emptyset\prime$，则，$\emptyset \subseteq \emptyset\prime \wedge \emptyset\prime \subseteq \emptyset$，
$\therefore\ \emptyset = \emptyset\prime$.

Example

- $\{a, b\}$的子集合有$\emptyset$，$\{a, b\}$，$\{a\}$，$\{b\}$，其中$\emptyset$ 和$\{a, b\}$称为平凡子集合(trivial);
- 空集本身可以是其他集合的元素，如：
  $\emptyset \subsetneq \{\emptyset\} \subsetneq \{\emptyset, \{\emptyset\}\}$;
- $|\emptyset| = 0$.

Remark

$\mathcal{U}$ *just likes* $\mathbb{T}$, *and* $\emptyset$ *likes* $\mathbb{F}$ *of logic.*

Univers & Empty set(2/2)

Theorem

$\emptyset$是唯一的，因为，设另有一空集$\emptyset\prime$，则，$\emptyset \subseteq \emptyset\prime \wedge \emptyset\prime \subseteq \emptyset$，$\therefore\ \emptyset = \emptyset\prime$.

Example

- $\{a, b\}$的子集合有$\emptyset$，$\{a, b\}$，$\{a\}$，$\{b\}$，其中$\emptyset$ 和$\{a, b\}$称为平凡子集合(trivial);
- 空集本身可以是其他集合的元素，如:
  $\emptyset \subsetneq \{\emptyset\} \subsetneq \{\emptyset, \{\emptyset\}\}$;
- $|\emptyset| = 0$.

Remark

$\mathcal{U}$ just likes $\mathbb{T}$, and $\emptyset$ likes $\mathbb{F}$ of logic.

交并差运算

Definition (Union, Intersection & diffrence)

- $A \cup B \triangleq \{x \mid x \in A \vee x \in B\}$;
- $A \cap B \triangleq \{x \mid x \in A \wedge x \in B\}$;
- $A - B \triangleq \{x \mid x \in A \wedge x \notin B\}$.

Description (文氏图John Venn Diagram)

(a) $A \cup B$ (b) $A \cap B$ (c) $A - B$

交并差运算

Definition (Union, Intersection & diffrence)

- $A \cup B \triangleq \{ x \mid x \in A \vee x \in B \}$;
- $A \cap B \triangleq \{ x \mid x \in A \wedge x \in B \}$;
- $A - B \triangleq \{ x \mid x \in A \wedge x \notin B \}$.

Description (文氏图John Venn Diagram)

(a) $A \cup B$ (b) $A \cap B$ (c) $A - B$

交并差运算

Definition (Union, Intersection & diffrence)

- $A \cup B \triangleq \{\, x \mid x \in A \vee x \in B \,\}$;
- $A \cap B \triangleq \{\, x \mid x \in A \wedge x \in B \,\}$;
- $A - B \triangleq \{\, x \mid x \in A \wedge x \notin B \,\}$.

Description (文氏图John Venn Diagram)

(a) $A \cup B$ (b) $A \cap B$ (c) $A - B$

交并差运算

Definition (Union, Intersection & diffrence)

- $A \cup B \triangleq \{x \mid x \in A \vee x \in B\}$;
- $A \cap B \triangleq \{x \mid x \in A \wedge x \in B\}$;
- $A - B \triangleq \{x \mid x \in A \wedge x \notin B\}$.

Description (文氏图John Venn Diagram)

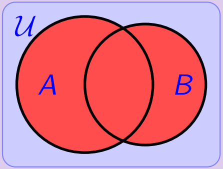
(a) $A \cup B$ (b) $A \cap B$ (c) $A - B$

交并差运算

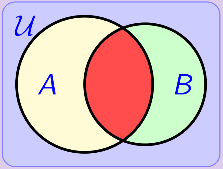
Definition (Union, Intersection & diffrence)

- $A \cup B \triangleq \{ x \,|\, x \in A \vee x \in B \}$;
- $A \cap B \triangleq \{ x \,|\, x \in A \wedge x \in B \}$;
- $A - B \triangleq \{ x \,|\, x \in A \wedge x \notin B \}$.

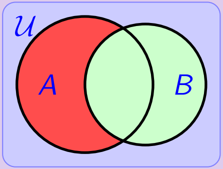
Description (文氏图John Venn Diagram)



(a) $A \cup B$



(b) $A \cap B$



(c) $A - B$

Example

Example

- $A = \{$ 所有含关键字“离散数学”的网页 $\}$;
- $B = \{$ 所有含关键字“wiki”的网页 $\}$;
- $A \cup B = \{$ 所有含关键字“离散数学”或“wiki”的网页 $\}$;
- $A \cap B = \{$ 所有含关键字“离散数学”和“wiki”的网页 $\}$;
- $A - B = \{$ 所有含关键字“离散数学”且不含“wiki”的网页 $\}$.

Example

Example

- $A = \{$ 所有含关键字"离散数学"的网页 $\}$;
- $B = \{$ 所有含关键字"wiki"的网页 $\}$;
- $A \cup B = \{$ 所有含关键字"离散数学"或"wiki"的网页 $\}$;
- $A \cap B = \{$ 所有含关键字"离散数学"和"wiki"的网页 $\}$;
- $A - B = \{$ 所有含关键字"离散数学"且不含"wiki"的网页 $\}$.

Example

Example

- $A = \{$所有含关键字“离散数学”的网页$\}$;
- $B = \{$所有含关键字“wiki”的网页$\}$;
- $A \cup B = \{$所有含关键字“离散数学”或“wiki”的网页$\}$;
- $A \cap B = \{$所有含关键字“离散数学”和“wiki”的网页$\}$;
- $A - B = \{$所有含关键字“离散数学”且不含“wiki”的网页$\}$.

Example

Example

- $A = \{$ 所有含关键字“离散数学”的网页 $\}$;
- $B = \{$ 所有含关键字“wiki”的网页 $\}$;
- $A \cup B = \{$ 所有含关键字“离散数学”或“wiki”的网页 $\}$;
- $A \cap B = \{$ 所有含关键字“离散数学”和“wiki”的网页 $\}$;
- $A - B = \{$ 所有含关键字“离散数学”且不含“wiki”的网页 $\}$.

Example

Example

- $A = \{$ 所有含关键字"离散数学"的网页 $\}$;
- $B = \{$ 所有含关键字"wiki"的网页 $\}$;
- $A \cup B = \{$ 所有含关键字"离散数学"或"wiki"的网页 $\}$;
- $A \cap B = \{$ 所有含关键字"离散数学"和"wiki"的网页 $\}$;
- $A - B = \{$ 所有含关键字"离散数学"且不含"wiki"的网页 $\}$.

Example

Example

- $A = \{$ 所有含关键字"离散数学"的网页 $\}$;
- $B = \{$ 所有含关键字"wiki"的网页 $\}$;
- $A \cup B = \{$ 所有含关键字"离散数学"或"wiki"的网页 $\}$;
- $A \cap B = \{$ 所有含关键字"离散数学"和"wiki"的网页 $\}$;
- $A - B = \{$ 所有含关键字"离散数学"且不含"wiki"的网页 $\}$.

不相交

Definition (不相交)

称集合A和B不相交，
iff. $A \cap B = \emptyset$

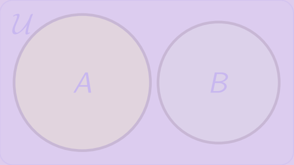
Example

$T \cap C = \emptyset, S \cap C = \emptyset, T \cap S \neq \emptyset$

不相交

Definition (不相交)

称集合A和B不相交,
iff, $A \cap B = \varnothing$



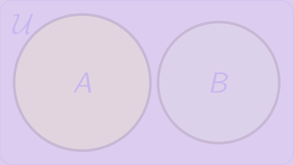
Example

$T = \{$所有重言式的集合$\}$,
$S = \{$所有可满足公式的集合$\}$,
$C = \{$所有矛盾式的集合$\}$;
$T \cap C = \varnothing$, $S \cap C = \varnothing$, $T \cap S \neq \varnothing$.

不相交

Definition (不相交)

称集合A和B不相交，
iff, $A \cap B = \varnothing$



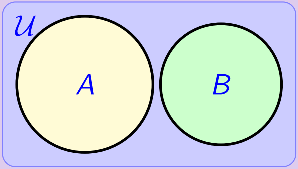
Example

不相交

Definition (不相交)

称集合A和B不相交，

iff, $A \cap B = \emptyset$

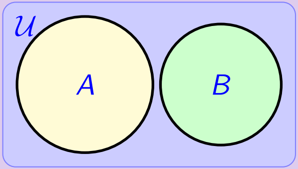


Example

不相交

Definition (不相交)

称集合A和B不相交，
iff, $A \cap B = \varnothing$



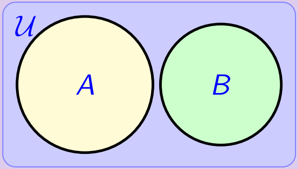
Example

- $T = \{$所有重言式的集合$\}$;
- $S = \{$所有可满足公式的集合$\}$;
- $C = \{$所有矛盾式的集合$\}$;
- $T \cap C = \varnothing$, $S \cap C = \varnothing$, $T \cap S \neq \varnothing$.

不相交

Definition (不相交)

称集合A和B不相交,
iff, $A \cap B = \varnothing$



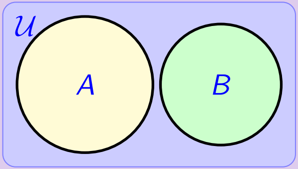
Example

- $T = \{$所有重言式的集合$\}$;
- $S = \{$所有可满足公式的集合$\}$;
- $C = \{$所有矛盾式的集合$\}$;
- $T \cap C = \varnothing$, $S \cap C = \varnothing$, $T \cap S \neq \varnothing$.

不相交

Definition (不相交)

称集合A和B不相交，
iff, $A \cap B = \varnothing$



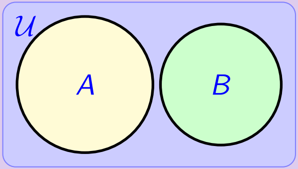
Example

- $T = \{$所有重言式的集合$\}$;
- $S = \{$所有可满足公式的集合$\}$;
- $C = \{$所有矛盾式的集合$\}$;
- $T \cap C = \varnothing$, $S \cap C = \varnothing$, $T \cap S \neq \varnothing$.

不相交

Definition (不相交)

称集合A和B不相交，
iff, $A \cap B = \emptyset$



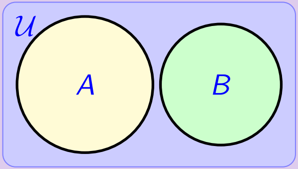
Example

- $T = \{$所有重言式的集合$\}$;
- $S = \{$所有可满足公式的集合$\}$;
- $C = \{$所有矛盾式的集合$\}$;
- $T \cap C = \emptyset$, $S \cap C = \emptyset$, $T \cap S \neq \emptyset$.

不相交

Definition (不相交)

称集合A和B不相交，
iff, $A \cap B = \varnothing$



Example

- $T = \{$所有重言式的集合$\}$;
- $S = \{$所有可满足公式的集合$\}$;
- $C = \{$所有矛盾式的集合$\}$;
- $T \cap C = \varnothing$, $S \cap C = \varnothing$, $T \cap S \neq \varnothing$.

集合运算的恒等式和不等式

1	$A \cup A = A$ $A \cap A = A$	幂等律
2	$A \cup B = B \cup A$ $A \cap B = B \cap A$	交换律
3	$(A \cup B) \cup C = A \cup (B \cup C)$ $(A \cap B) \cap C = A \cap (B \cap C)$	结合律
4	$(A \cup B) \cap C = (A \cap C) \cup (B \cap C)$ $(A \cap B) \cup C = (A \cup C) \cap (B \cup C)$	分配律
5	$(A \cap B) \cup A = A$ $(A \cup B) \cap A = A$	吸收律
6	$A \cup \varnothing = A,\ A \cap \varnothing = \varnothing$ $A \cup \mathcal{U} = \mathcal{U},\ A \cap \mathcal{U} = A$	简化式
7	$A \subseteq A \cup B,\ \ A \cap B \subseteq A$	
8	$A - B \subseteq A$	
9	$if\ A \subseteq B \wedge C \subseteq D,\ then$ $A \cup C \subseteq B \cup D$ $A \cap C \subseteq B \cap D$	

集合恒等式和不等式的证明

注意：集合上的运算没有定义优先级别，所以$\cancel{A \cup B \cap C}$，但是因为有结合律，所以：$A \cup B \cup C$ ✓.

Remark

集合上的运算本质上和逻辑运算相同，因此逻辑上的恒等式和不等式可以完全平移到集合上来：

$$\text{“}=\text{”} \quad \leftrightsquigarrow \quad \text{“}\Leftrightarrow\text{”}$$
$$\text{“}\subseteq\text{”} \quad \leftrightsquigarrow \quad \text{“}\Rightarrow\text{”}$$

Remark (证明方法)

集合恒等式和不等式的证明

注意：集合上的运算没有定义优先级别，所以$\cancel{A \cup B \cap C}$，但是因为有结合律，所以：$A \cup B \cup C$ ✓.

Remark

集合上的运算本质上和逻辑运算相同，因此逻辑上的恒等式和不等式可以完全平移到集合上来：

$$\text{“}=\text{”} \quad \leftrightsquigarrow \quad \text{“}\Leftrightarrow\text{”}$$

$$\text{“}\subseteq\text{”} \quad \leftrightsquigarrow \quad \text{“}\Rightarrow\text{”}$$

Remark (证明方法)

- 利用定义证明；

集合恒等式和不等式的证明

注意：集合上的运算没有定义优先级别，所以 $\cancel{A \cup B \cap C}$，但是因为有结合律，所以：$A \cup B \cup C$ ✓.

Remark

集合上的运算本质上和逻辑运算相同，因此逻辑上的恒等式和不等式可以完全平移到集合上来：

$$\text{“} = \text{”} \quad \leftrightsquigarrow \quad \text{“} \Leftrightarrow \text{”}$$
$$\text{“} \subseteq \text{”} \quad \leftrightsquigarrow \quad \text{“} \Rightarrow \text{”}$$

Remark (证明方法)

- 利用定义证明；
- 利用代入和替换规则+恒等和不等变换；
- 证明的表述方式：纯一阶逻辑语言，或者自然语言.

集合恒等式和不等式的证明

注意：集合上的运算没有定义优先级别，所以~~$A \cup B \cap C$~~，但是因为有结合律，所以：$A \cup B \cup C$ ✓.

Remark

集合上的运算本质上和逻辑运算相同，因此逻辑上的恒等式和不等式可以完全平移到集合上来：

$$\text{“} = \text{”} \quad \leftrightsquigarrow \quad \text{“} \Leftrightarrow \text{”}$$
$$\text{“} \subseteq \text{”} \quad \leftrightsquigarrow \quad \text{“} \Rightarrow \text{”}$$

Remark (证明方法)

- 利用定义证明；
- 利用代入和替换规则+恒等和不等变换；
- 证明的表述方式：纯一阶逻辑语言，或者自然语言.

集合恒等式和不等式的证明

注意：集合上的运算没有定义优先级别，所以 ~~$A \cup B \cap C$~~，但是因为有结合律，所以：$A \cup B \cup C$ ✓.

Remark

集合上的运算本质上和逻辑运算相同，因此逻辑上的恒等式和不等式可以完全平移到集合上来：

$$\text{“} = \text{”} \quad \leftrightsquigarrow \quad \text{“} \Leftrightarrow \text{”}$$
$$\text{“} \subseteq \text{”} \quad \leftrightsquigarrow \quad \text{“} \Rightarrow \text{”}$$

Remark (证明方法)

- 利用定义证明；
- 利用代入和替换规则＋恒等和不等变换；
- 证明的表述方式：纯一阶逻辑语言，或者自然语言.

集合恒等式和不等式的证明

注意：集合上的运算没有定义优先级别，所以$\cancel{A \cup B \cap C}$，但是因为有结合律，所以：$A \cup B \cup C$ ✓.

Remark

集合上的运算本质上和逻辑运算相同，因此逻辑上的恒等式和不等式可以完全平移到集合上来：

$$\text{“} = \text{”} \quad \leftrightsquigarrow \quad \text{“} \Leftrightarrow \text{”}$$
$$\text{“} \subseteq \text{”} \quad \leftrightsquigarrow \quad \text{“} \Rightarrow \text{”}$$

Remark (证明方法)

- 利用定义证明；
- 利用代入和替换规则＋恒等和不等变换；
- 证明的表述方式：纯一阶逻辑语言，或者自然语言.

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof. (自然语言)

Example

Example

$$A \subseteq B \land C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(逻辑语言)

Proof.(自然语言)

① $\forall x \in A \cap C$;

则$x \in A$, 并且$x \in C$;

因为$A \subseteq B \land C \subseteq D$;

所以$x \in B$, 并且$x \in D$;

即$x \in B \cap D$;

故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(逻辑语言)

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(逻辑语言)

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(逻辑语言)

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \Longrightarrow A \cap C \subseteq B \cap D$$

Proof.(逻辑语言)

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \Longrightarrow A \cap C \subseteq B \cap D$$

Proof.(逻辑语言)

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \Longrightarrow A \cap C \subseteq B \cap D$$

Proof.(逻辑语言)

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x(x \in A \cap C)$
2. $\iff \forall x(x \in A \wedge x \in C)$ (by def)
3. $\iff \forall x(x \in A) \wedge \forall x(x \in C)$ (by 量词分配)
4. $\implies \forall x(x \in B) \wedge \forall x(x \in D)$ (by 条件)
5. $\iff \forall x(x \in B \wedge x \in D)$ (by 量词分配)
6. $\iff \forall x(x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x(x \in A \cap C)$
2. $\iff \forall x(x \in A \wedge x \in C)$ (by def)
3. $\iff \forall x(x \in A) \wedge \forall x(x \in C)$ (by 量词分配)
4. $\implies \forall x(x \in B) \wedge \forall x(x \in D)$ (by 条件)
5. $\iff \forall x(x \in B \wedge x \in D)$ (by 量词分配)
6. $\iff \forall x(x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x(x \in A \cap C)$
2. $\iff \forall x(x \in A \wedge x \in C)$ (by def)
3. $\iff \forall x(x \in A) \wedge \forall x(x \in C)$ (by 量词分配)
4. $\implies \forall x(x \in B) \wedge \forall x(x \in D)$ (by 条件)
5. $\iff \forall x(x \in B \wedge x \in D)$ (by 量词分配)
6. $\iff \forall x(x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example

Example

$$A \subseteq B \wedge C \subseteq D \Longrightarrow A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x(x \in A \cap C)$
2. $\Longleftrightarrow \forall x(x \in A \wedge x \in C)$ (by def)
3. $\Longleftrightarrow \forall x(x \in A) \wedge \forall x(x \in C)$ (by 量词分配)
4. $\Longrightarrow \forall x(x \in B) \wedge \forall x(x \in D)$ (by 条件)
5. $\Longleftrightarrow \forall x(x \in B \wedge x \in D)$ (by 量词分配)
6. $\Longleftrightarrow \forall x(x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x(x \in A \cap C)$
2. $\iff \forall x(x \in A \wedge x \in C)$ (by def)
3. $\iff \forall x(x \in A) \wedge \forall x(x \in C)$ (by 量词分配)
4. $\implies \forall x(x \in B) \wedge \forall x(x \in D)$ (by 条件)
5. $\iff \forall x(x \in B \wedge x \in D)$ (by 量词分配)
6. $\iff \forall x(x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x(x \in A \cap C)$
2. $\iff \forall x(x \in A \wedge x \in C)$ (by def)
3. $\iff \forall x(x \in A) \wedge \forall x(x \in C)$ (by 量词分配)
4. $\implies \forall x(x \in B) \wedge \forall x(x \in D)$ (by 条件)
5. $\iff \forall x(x \in B \wedge x \in D)$ (by 量词分配)
6. $\iff \forall x(x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example

Example

$$A \subseteq B \wedge C \subseteq D \Longrightarrow A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x (x \in A \cap C)$
2. $\Longleftrightarrow \forall x (x \in A \wedge x \in C)$ (by def)
3. $\Longleftrightarrow \forall x (x \in A) \wedge \forall x (x \in C)$ (by 量词分配)
4. $\Longrightarrow \forall x (x \in B) \wedge \forall x (x \in D)$ (by 条件)
5. $\Longleftrightarrow \forall x (x \in B \wedge x \in D)$ (by 量词分配)
6. $\Longleftrightarrow \forall x (x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example

Example

$$A \subseteq B \wedge C \subseteq D \implies A \cap C \subseteq B \cap D$$

Proof.(自然语言)

1. $\forall x \in A \cap C$;
2. 则$x \in A$, 并且$x \in C$;
3. 因为$A \subseteq B \wedge C \subseteq D$;
4. 所以$x \in B$, 并且$x \in D$;
5. 即$x \in B \cap D$;
6. 故$A \cap C \subseteq B \cap D$.

Proof.(逻辑语言)

1. $\forall x(x \in A \cap C)$
2. $\iff \forall x(x \in A \wedge x \in C)$ (by def)
3. $\iff \forall x(x \in A) \wedge \forall x(x \in C)$ (by 量词分配)
4. $\implies \forall x(x \in B) \wedge \forall x(x \in D)$ (by 条件)
5. $\iff \forall x(x \in B \wedge x \in D)$ (by 量词分配)
6. $\iff \forall x(x \in B \cap D)$ (by def)
7. $\therefore A \cap C \subseteq B \cap D$

Example(2/2)

Theorem

$$\underbrace{A \subseteq B}_{①} \Longleftrightarrow \underbrace{A \cup B = B}_{②} \Longleftrightarrow \underbrace{A \cap B = A}_{③}$$

Proof.

① $\Longrightarrow$ ② :

- $\because A \subseteq B \wedge B \subseteq B$;

  $\therefore A \cup B \subseteq B \cup B = B$ ([illegible])

② $\Longrightarrow$ ③ :

$\because A \cup B = B$,

$\therefore A \cap B = A \cap (A \cup B) = A$ ([illegible])

③ $\Longrightarrow$ ① :

$A = A \cap B \subseteq B$;

Example(2/2)

Theorem

$$\underbrace{A \subseteq B}_{①} \Longleftrightarrow \underbrace{A \cup B = B}_{②} \Longleftrightarrow \underbrace{A \cap B = A}_{③}$$

Proof.

① $\Longrightarrow$ ② :

- $\because A \subseteq B \wedge B \subseteq B$;
- $\therefore A \cup B \subseteq B \cup B = B$ (幂等律)

② $\Longrightarrow$ ③ :

- $\because A \cup B = B$;
- $\therefore A \cap B = A \cap (A \cup B) = A$ (吸收律)

③ $\Longrightarrow$ ① :

- $A = A \cap B \subseteq B$;

$\square$

Example(2/2)

Theorem

$$\underbrace{A \subseteq B}_{①} \Longleftrightarrow \underbrace{A \cup B = B}_{②} \Longleftrightarrow \underbrace{A \cap B = A}_{③}$$

Proof.

① $\Longrightarrow$ ② :

- $\because A \subseteq B \wedge B \subseteq B$;
- $\therefore A \cup B \subseteq B \cup B = B$ (幂等律)

② $\Longrightarrow$ ③ :

- $\because A \cup B = B$;
- $\therefore A \cap B = A \cap (A \cup B) = A$ (吸收律)

③ $\Longrightarrow$ ① :

- $A = A \cap B \subseteq B$;

□

Example(2/2)

Theorem

$$\underbrace{A \subseteq B}_{①} \iff \underbrace{A \cup B = B}_{②} \iff \underbrace{A \cap B = A}_{③}$$

Proof.

① $\Longrightarrow$ ② :

- $\because A \subseteq B \wedge B \subseteq B$;
- $\therefore A \cup B \subseteq B \cup B = B$ (幂等律)

② $\Longrightarrow$ ③ :

- $\because A \cup B = B$;
- $\therefore A \cap B = A \cap (A \cup B) = A$ (吸收律)

③ $\Longrightarrow$ ① :

- $A = A \cap B \subseteq B$;

$\square$

Example(2/2)

Theorem

$$\underbrace{A \subseteq B}_{①} \iff \underbrace{A \cup B = B}_{②} \iff \underbrace{A \cap B = A}_{③}$$

Proof.

① $\Longrightarrow$ ② :

- $\because A \subseteq B \wedge B \subseteq B$;
- $\therefore A \cup B \subseteq B \cup B = B$ (幂等律)

② $\Longrightarrow$ ③ :

- $\because A \cup B = B$;
- $\therefore A \cap B = A \cap (A \cup B) = A$ (吸收律)

③ $\Longrightarrow$ ① :

- $A = A \cap B \subseteq B$;

$\square$

Example(2/2)

Theorem

$$\underbrace{A \subseteq B}_{①} \iff \underbrace{A \cup B = B}_{②} \iff \underbrace{A \cap B = A}_{③}$$

Proof.

① $\Longrightarrow$ ② :

- $\because A \subseteq B \wedge B \subseteq B$;
- $\therefore A \cup B \subseteq B \cup B = B$ (幂等律)

② $\Longrightarrow$ ③ :

- $\because A \cup B = B$;
- $\therefore A \cap B = A \cap (A \cup B) = A$ (吸收律)

③ $\Longrightarrow$ ① :

- $A = A \cap B \subseteq B$;

$\square$

Example(2/2)

Theorem

$$\underbrace{A \subseteq B}_{①} \iff \underbrace{A \cup B = B}_{②} \iff \underbrace{A \cap B = A}_{③}$$

Proof.

① $\Longrightarrow$ ② :

- $\because A \subseteq B \wedge B \subseteq B$;
- $\therefore A \cup B \subseteq B \cup B = B$ (幂等律)

② $\Longrightarrow$ ③ :

- $\because A \cup B = B$;
- $\therefore A \cap B = A \cap (A \cup B) = A$ (吸收律)

③ $\Longrightarrow$ ① :

- $A = A \cap B \subseteq B$;

$\square$

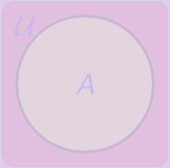
补运算

Definition (补集, Complementary Set)

补运算

Definition (补集, Complementary Set)

$$\overline{A} \triangleq \mathcal{U} - A$$
$$= \{x \mid x \in \mathcal{U} \wedge x \notin A\}$$



Example

$S = \{$所有可满足公式的集合$\}$;

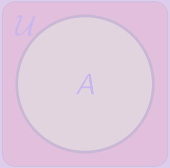
$C = \{$所有矛盾式的集合$\}$;

$C = \overline{S}$ if $\mathcal{U} = \{$所有命题公式的集合$\}$

补运算

Definition (补集, Complementary Set)

$$\overline{A} \triangleq \mathcal{U} - A = \{x \mid x \in \mathcal{U} \wedge x \notin A\}$$

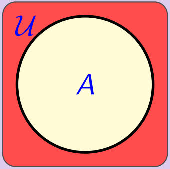


Example

补运算

Definition (补集, Complementary Set)

$$\overline{A} \triangleq \mathcal{U} - A$$
$$= \{x \mid x \in \mathcal{U} \land x \notin A\}$$

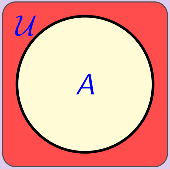


Example

补运算

Definition (补集, Complementary Set)

$$\overline{A} \triangleq \mathcal{U} - A$$
$$= \{\, x \mid x \in \mathcal{U} \wedge x \notin A \,\}$$



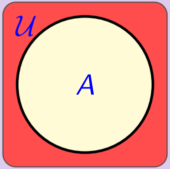
Example

- $S = \{$ 所有可满足公式的集合 $\}$;
- $C = \{$ 所有矛盾式的集合 $\}$;
- $\overline{C} = S$ if $\mathcal{U} = \{$ 所有命题公式的集合 $\}$.

补运算

Definition (补集, Complementary Set)

$$\begin{aligned}\overline{A} &\triangleq \mathcal{U} - A \\ &= \{\, x \mid x \in \mathcal{U} \wedge x \notin A \,\}\end{aligned}$$



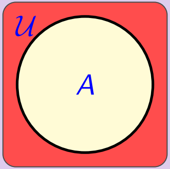
Example

- $S = \{$ 所有可满足公式的集合 $\}$;
- $C = \{$ 所有矛盾式的集合 $\}$;
- $\overline{C} = S$ if $\mathcal{U} = \{$ 所有命题公式的集合 $\}$.

补运算

Definition (补集, Complementary Set)

$$\overline{A} \triangleq \mathcal{U} - A$$
$$= \{ x \mid x \in \mathcal{U} \land x \notin A \}$$



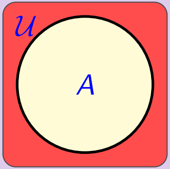
Example

- $S = \{$ 所有可满足公式的集合 $\}$;
- $C = \{$ 所有矛盾式的集合 $\}$;
- $\overline{C} = S$ if $\mathcal{U} = \{$ 所有命题公式的集合 $\}$.

补运算

Definition (补集, Complementary Set)

$$
\begin{aligned}
\overline{A} &\triangleq \mathcal{U} - A \\
&= \{\, x \mid x \in \mathcal{U} \wedge x \notin A \,\}
\end{aligned}
$$



Example

- $S = \{$所有可满足公式的集合$\}$;
- $C = \{$所有矛盾式的集合$\}$;
- $\overline{C} = S$ if $\mathcal{U} = \{$所有命题公式的集合$\}$.

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$
\begin{array}{lll}
1 & B & = \mathcal{U} \cap B \\
2 & & = (A \cup \overline{A}) \cap B \\
3 & & = (A \cap B) \cup (\overline{A} \cap B) \\
4 & & = (A \cap \overline{A}) \cup (\overline{A} \cap B) \\
5 & & = \overline{A} \cap (A \cup B) \\
6 & & = \overline{A}
\end{array}
$$

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$
\begin{array}{lll}
1 & B & = \mathcal{U} \cap B \\
2 & & = (A \cup \overline{A}) \cap B \\
3 & & = (A \cap B) \cup (\overline{A} \cap B) \\
4 & & = (\overline{A} \cap A) \cup (\overline{A} \cap B) \\
5 & & = \overline{A} \cap (A \cup B) \\
6 & & = \overline{A}
\end{array}
$$

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$\begin{aligned}
1 \quad B &= \mathcal{U} \cap B \\
2 \quad &= (A \cup \overline{A}) \cap B \\
3 \quad &= (A \cap B) \cup (\overline{A} \cap B) \\
4 \quad &= (\overline{A} \cap A) \cup (\overline{A} \cap B) \\
5 \quad &= \overline{A} \cap (A \cup B) \\
6 \quad &= \overline{A}
\end{aligned}$$

□

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$
\begin{array}{lll}
1 & B & = \mathcal{U} \cap B \\
2 & & = (A \cup \overline{A}) \cap B \\
3 & & = (A \cap B) \cup (\overline{A} \cap B) \\
4 & & = (\overline{A} \cap A) \cup (\overline{A} \cap B) \\
5 & & = \overline{A} \cap (A \cup B) \\
6 & & = \overline{A}
\end{array}
$$

$\square$

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$\begin{array}{lll}
1 & B & = \mathcal{U} \cap B \\
2 & & = (A \cup \overline{A}) \cap B \\
3 & & = (A \cap B) \cup (\overline{A} \cap B) \\
4 & & = (\overline{A} \cap A) \cup (\overline{A} \cap B) \\
5 & & = \overline{A} \cap (A \cup B) \\
6 & & = \overline{A}
\end{array}$$

$\square$

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$\begin{array}{lll}
1 & B & = \mathcal{U} \cap B \\
2 & & = (A \cup \overline{A}) \cap B \\
3 & & = (A \cap B) \cup (\overline{A} \cap B) \\
4 & & = (\overline{A} \cap A) \cup (\overline{A} \cap B) \\
5 & & = \overline{A} \cap (A \cup B) \\
6 & & = \overline{A}
\end{array}$$

$\square$

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$
\begin{array}{lll}
1 & B & = \mathcal{U} \cap B \\
2 & & = (A \cup \overline{A}) \cap B \\
3 & & = (A \cap B) \cup (\overline{A} \cap B) \\
4 & & = (\overline{A} \cap A) \cup (\overline{A} \cap B) \\
5 & & = \overline{A} \cap (A \cup B) \\
6 & & = \overline{A}
\end{array}
$$

□

补的性质

Theorem

1. $A \cup \overline{A} = \mathcal{U}, \quad A \cap \overline{A} = \varnothing$
2. (唯一性定理) A, B 2 sets, B是A的补的充要条件是:
$$A \cup B = \mathcal{U} \wedge A \cap B = \varnothing$$

②的充分性的证明.

$$\begin{array}{lll}
1 & B & = \mathcal{U} \cap B \\
2 & & = (A \cup \overline{A}) \cap B \\
3 & & = (A \cap B) \cup (\overline{A} \cap B) \\
4 & & = (\overline{A} \cap A) \cup (\overline{A} \cap B) \\
5 & & = \overline{A} \cap (A \cup B) \\
6 & & = \overline{A}
\end{array}$$

□

补的恒等式和不等式

1	$\overline{\varnothing} = \mathcal{U}, \overline{\mathcal{U}} = \varnothing$	幂等律
2	$\overline{\overline{A}} = A$	排中律
3	$\overline{A \cup B} = \overline{A} \cap \overline{B}$ $\overline{A \cap B} = \overline{A} \cup \overline{B}$	De Morgan律
4	$A \subseteq B \Longleftrightarrow \overline{B} \subseteq \overline{A}$	
5	$A - B = A \cap \overline{B}$	
6	$A \subseteq B \Longleftrightarrow A - B = \varnothing$	

求补运算相当于逻辑否定运算.

补的恒等式和不等式

1	$\overline{\varnothing} = \mathcal{U}, \overline{\mathcal{U}} = \varnothing$	幂等律
2	$\overline{\overline{A}} = A$	排中律
3	$\overline{A \cup B} = \overline{A} \cap \overline{B}$ $\overline{A \cap B} = \overline{A} \cup \overline{B}$	De Morgan律
4	$A \subseteq B \Longleftrightarrow \overline{B} \subseteq \overline{A}$	
5	$A - B = A \cap \overline{B}$	
6	$A \subseteq B \Longleftrightarrow A - B = \varnothing$	

求补运算相当于逻辑否定运算.

example

Example

$$A \subseteq B \Longleftrightarrow A - B = \varnothing$$

Proof.

example

Example

$$A \subseteq B \iff A - B = \emptyset$$

Proof.

$\Longleftarrow$:

$$\begin{array}{lll}
 & & A - B = \emptyset \\
1 & \iff & A \cap \overline{B} = \emptyset \\
2 & \Longrightarrow & A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\
3 & \Longrightarrow & A \subseteq B
\end{array}$$

$\Longrightarrow$:

$$\begin{array}{lll}
 & & A \subseteq B, \overline{B} \subseteq \overline{B} \\
4 & \Longrightarrow & \emptyset \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \emptyset \\
5 & \Longrightarrow & A \cap \overline{B} = \emptyset
\end{array}$$

□

example

Example

$$A \subseteq B \Longleftrightarrow A - B = \varnothing$$

Proof.

$\Longleftarrow$:

$$A - B = \varnothing$$

1 $\Longleftrightarrow A \cap \overline{B} = \varnothing$

2 $\Longrightarrow A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B$

3 $\Longrightarrow A \subseteq B$

$\Longrightarrow$:

$$A \subseteq B, \overline{B} \subseteq \overline{B}$$

4 $\Longrightarrow \varnothing \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \varnothing$

5 $\Longrightarrow A \cap \overline{B} = \varnothing$

$\square$

example

Example

$$A \subseteq B \iff A - B = \emptyset$$

Proof.

$\Longleftarrow$:

$$
\begin{aligned}
& A - B = \emptyset \\
1 \quad & \iff A \cap \overline{B} = \emptyset \\
2 \quad & \Longrightarrow A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\
3 \quad & \Longrightarrow A \subseteq B
\end{aligned}
$$

$\Longrightarrow$:

$$
\begin{aligned}
& A \subseteq B, \overline{B} \subseteq \overline{B} \\
4 \quad & \Longrightarrow \emptyset \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \emptyset \\
5 \quad & \Longrightarrow A \cap \overline{B} = \emptyset
\end{aligned}
$$

□

example

Example

$$A \subseteq B \Longleftrightarrow A - B = \emptyset$$

Proof.

$\Longleftarrow$:

$$
\begin{aligned}
& & A - B &= \emptyset \\
1 & \Longleftrightarrow & A \cap \overline{B} &= \emptyset \\
2 & \Longrightarrow & A &= A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\
3 & \Longrightarrow & A &\subseteq B
\end{aligned}
$$

$\Longrightarrow$:

$$
\begin{aligned}
& & & A \subseteq B, \overline{B} \subseteq \overline{B} \\
4 & \Longrightarrow & & \emptyset \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \emptyset \\
5 & \Longrightarrow & & A \cap \overline{B} = \emptyset
\end{aligned}
$$

$\square$

example

Example

$$A \subseteq B \iff A - B = \varnothing$$

Proof.

$\Longleftarrow$:

$$\begin{array}{lll} & & A - B = \varnothing \\ 1 & \iff & A \cap \overline{B} = \varnothing \\ 2 & \Longrightarrow & A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\ 3 & \Longrightarrow & A \subseteq B \end{array}$$

$\Longrightarrow$:

$$\begin{array}{lll} & & A \subseteq B, \overline{B} \subseteq \overline{B} \\ 4 & \Longrightarrow & \varnothing \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \varnothing \\ 5 & \Longrightarrow & A \cap \overline{B} = \varnothing \end{array}$$

$\square$

example

Example

$$A \subseteq B \Longleftrightarrow A - B = \emptyset$$

Proof.

$\Longleftarrow$:

$$\begin{array}{lll} & & A - B = \emptyset \\ 1 & \Longleftrightarrow & A \cap \overline{B} = \emptyset \\ 2 & \Longrightarrow & A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\ 3 & \Longrightarrow & A \subseteq B \end{array}$$

$\Longrightarrow$:

$$\begin{array}{lll} & & A \subseteq B, \overline{B} \subseteq \overline{B} \\ 4 & \Longrightarrow & \emptyset \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \emptyset \\ 5 & \Longrightarrow & A \cap \overline{B} = \emptyset \end{array}$$

□

example

Example

$$A \subseteq B \Longleftrightarrow A - B = \emptyset$$

Proof.

$\Longleftarrow$:

$$\begin{array}{lll} & & A - B = \emptyset \\ 1 & \Longleftrightarrow & A \cap \overline{B} = \emptyset \\ 2 & \Longrightarrow & A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\ 3 & \Longrightarrow & A \subseteq B \end{array}$$

$\Longrightarrow$:

$$\begin{array}{lll} & & A \subseteq B, \overline{B} \subseteq \overline{B} \\ 4 & \Longrightarrow & \emptyset \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \emptyset \\ 5 & \Longrightarrow & A \cap \overline{B} = \emptyset \end{array}$$

□

example

Example

$$A \subseteq B \Longleftrightarrow A - B = \varnothing$$

Proof.

$\Longleftarrow$:

$$\begin{aligned}
& & A - B = \varnothing \\
1 & \Longleftrightarrow & A \cap \overline{B} = \varnothing \\
2 & \Longrightarrow & A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\
3 & \Longrightarrow & A \subseteq B
\end{aligned}$$

$\Longrightarrow$:

$$\begin{aligned}
& & A \subseteq B, \overline{B} \subseteq \overline{B} \\
4 & \Longrightarrow & \varnothing \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \varnothing \\
5 & \Longrightarrow & A \cap \overline{B} = \varnothing
\end{aligned}$$

□

example

Example

$$A \subseteq B \iff A - B = \varnothing$$

Proof.

$\Longleftarrow$:

$$
\begin{array}{lll}
 & & A - B = \varnothing \\
1 & \iff & A \cap \overline{B} = \varnothing \\
2 & \Longrightarrow & A = A \cap (B \cup \overline{B}) = (A \cap B) \cup (A \cap \overline{B}) = A \cap B \\
3 & \Longrightarrow & A \subseteq B
\end{array}
$$

$\Longrightarrow$:

$$
\begin{array}{lll}
 & & A \subseteq B,\ \overline{B} \subseteq \overline{B} \\
4 & \Longrightarrow & \varnothing \subseteq A \cap \overline{B} \subseteq B \cap \overline{B} = \varnothing \\
5 & \Longrightarrow & A \cap \overline{B} = \varnothing
\end{array}
$$

$\square$

文氏图和范式(见维基)

极小项

$$\left.\begin{array}{cc} 0 & 1 \\ \neg P & Q \end{array}\right\} \neg P \wedge Q (\overline{A} \cap B)$$

极大项

$$\left.\begin{array}{cc} 0 & 1 \\ P & \neg Q \end{array}\right\} P \vee \neg Q (A \cup \overline{B})$$

区域与极大项和极小项的关系

序号	A	B	极小项	对应区域	极大项	对应区域
0	0	0	$\overline{A} \cap \overline{B}$	0	$A \cup B$	123
1	0	1	$\overline{A} \cap B$	1	$A \cup \overline{B}$	023
2	1	0	$A \cap \overline{B}$	2	$\overline{A} \cup B$	013
3	1	1	$A \cap B$	3	$\overline{A} \cup \overline{B}$	012

Example

$$\text{区域}3 = A \cap B = (A \cup B) \cap (A \cup \overline{B}) \cap (\overline{A} \cup B)$$

文氏图和范式(见维基)

极小项

$$\left.\begin{array}{cc} 0 & 1 \\ \neg P & Q \end{array}\right\} \neg P \wedge Q(\overline{A} \cap B)$$

极大项

$$\left.\begin{array}{cc} 0 & 1 \\ P & \neg Q \end{array}\right\} P \vee \neg Q(A \cup \overline{B})$$

区域与极大项和极小项的关系

序号	A	B	极小项	对应区域	极大项	对应区域
0	0	0	$\overline{A} \cap \overline{B}$	0	$A \cup B$	123
1	0	1	$\overline{A} \cap B$	1	$A \cup \overline{B}$	023
2	1	0	$A \cap \overline{B}$	2	$\overline{A} \cup B$	013
3	1	1	$A \cap B$	3	$\overline{A} \cup \overline{B}$	012

Example

$$区域3 = A \cap B = (A \cup B) \cap (A \cup \overline{B}) \cap (\overline{A} \cup B)$$

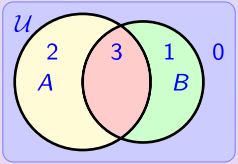
文氏图和范式(见维基)

极小项

$$\left.\begin{array}{cc} 0 & 1 \\ \neg P & Q \end{array}\right\} \neg P \wedge Q (\overline{A} \cap B)$$

极大项

$$\left.\begin{array}{cc} 0 & 1 \\ P & \neg Q \end{array}\right\} P \vee \neg Q (A \cup \overline{B})$$



区域与极大项和极小项的关系

序号	A	B	极小项	对应区域	极大项	对应区域
0	0	0	$\overline{A} \cap \overline{B}$	0	$A \cup B$	123
1	0	1	$\overline{A} \cap B$	1	$A \cup \overline{B}$	023
2	1	0	$A \cap \overline{B}$	2	$\overline{A} \cup B$	013
3	1	1	$A \cap B$	3	$\overline{A} \cup \overline{B}$	012

Example

$$\text{区域}3 = A \cap B = (A \cup B) \cap (A \cup \overline{B}) \cap (\overline{A} \cup B)$$

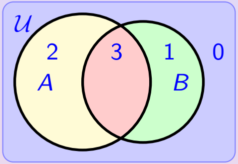
文氏图和范式(见维基)

极小项

$$\left.\begin{array}{cc} 0 & 1 \\ \neg P & Q \end{array}\right\} \neg P \wedge Q(\overline{A} \cap B)$$

极大项

$$\left.\begin{array}{cc} 0 & 1 \\ P & \neg Q \end{array}\right\} P \vee \neg Q(A \cup \overline{B})$$



区域与极大项和极小项的关系

序号	A	B	极小项	对应区域	极大项	对应区域
0	0	0	$\overline{A} \cap \overline{B}$	0	$A \cup B$	123
1	0	1	$\overline{A} \cap B$	1	$A \cup \overline{B}$	023
2	1	0	$A \cap \overline{B}$	2	$\overline{A} \cup B$	013
3	1	1	$A \cap B$	3	$\overline{A} \cup \overline{B}$	012

Example

$$\text{区域}3 = A \cap B = (A \cup B) \cap (A \cup \overline{B}) \cap (\overline{A} \cup B)$$

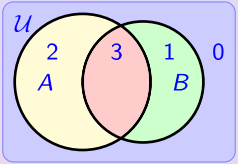
文氏图和范式(见维基)

极小项

$$\left.\begin{array}{cc} 0 & 1 \\ \neg P & Q \end{array}\right\} \neg P \wedge Q(\overline{A} \cap B)$$

极大项

$$\left.\begin{array}{cc} 0 & 1 \\ P & \neg Q \end{array}\right\} P \vee \neg Q(A \cup \overline{B})$$



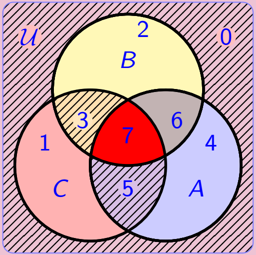
区域与极大项和极小项的关系

序号	A	B	极小项	对应区域	极大项	对应区域
0	0	0	$\overline{A} \cap \overline{B}$	0	$A \cup B$	123
1	0	1	$\overline{A} \cap B$	1	$A \cup \overline{B}$	023
2	1	0	$A \cap \overline{B}$	2	$\overline{A} \cup B$	013
3	1	1	$A \cap B$	3	$\overline{A} \cup \overline{B}$	012

Example

$$\text{区域}3 = A \cap B = (A \cup B) \cap (A \cup \overline{B}) \cap (\overline{A} \cup B)$$

Example —求阴影部分的集合解析式



成真指派与极小项

	A	B	C	Minterm
0	0	0	0	$\overline{A}\cap\overline{B}\cap\overline{C}$
3	0	1	1	$\overline{A}\cap B\cap C$
5	1	0	1	$A\cap\overline{B}\cap C$

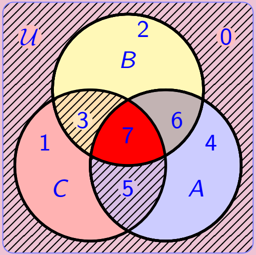
成假指派与极大项

	A	B	C	Maxterm
1	0	0	1	$A\cup B\cup\overline{C}$
2	0	1	0	$A\cup\overline{B}\cup C$
4	1	0	0	$\overline{A}\cup B\cup C$
6	1	1	0	$\overline{A}\cup\overline{B}\cup C$
7	1	1	1	$\overline{A}\cup\overline{B}\cup\overline{C}$

阴影对应的集合

$$\text{区域}035=(\overline{A}\cap\overline{B}\cap\overline{C})\cup(\overline{A}\cap B\cap C)\cup A\cap\overline{B}\cap C$$
$$=(A\cup B\cup\overline{C})\cap(A\cup\overline{B}\cup C)\cap(\overline{A}\cup B\cup C)\cap(\overline{A}\cup\overline{B}\cup C)\cap(\overline{A}\cup\overline{B}\cup\overline{C})$$

Example —求阴影部分的集合解析式



成真指派与极小项

	A	B	C	Minterm
0	0	0	0	$\overline{A} \cap \overline{B} \cap \overline{C}$
3	0	1	1	$\overline{A} \cap B \cap C$
5	1	0	1	$A \cap \overline{B} \cap C$

成假指派与极大项

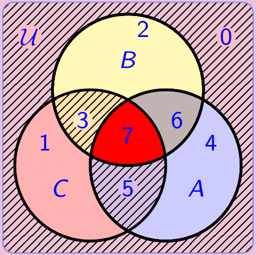
	A	B	C	Maxterm
1	0	0	1	$A \cup B \cup \overline{C}$
2	0	1	0	$A \cup \overline{B} \cup C$
4	1	0	0	$\overline{A} \cup B \cup C$
6	1	1	0	$\overline{A} \cup \overline{B} \cup C$
7	1	1	1	$\overline{A} \cup \overline{B} \cup \overline{C}$

阴影对应的集合

区域035 $= (\overline{A} \cap \overline{B} \cap \overline{C}) \cup (\overline{A} \cap B \cap C) \cup A \cap \overline{B} \cap C$

$= (A \cup B \cup \overline{C}) \cap (A \cup \overline{B} \cup C) \cap (\overline{A} \cup B \cup C) \cap (\overline{A} \cup \overline{B} \cup C) \cap (\overline{A} \cup \overline{B} \cup \overline{C})$

Example —求阴影部分的集合解析式



成真指派与极小项

	A	B	C	Minterm
0	0	0	0	$\overline{A}\cap\overline{B}\cap\overline{C}$
3	0	1	1	$\overline{A}\cap B\cap C$
5	1	0	1	$A\cap\overline{B}\cap C$

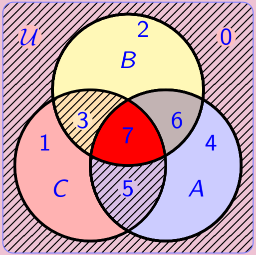
成假指派与极大项

	A	B	C	Maxterm
1	0	0	1	$A\cup B\cup\overline{C}$
2	0	1	0	$A\cup\overline{B}\cup C$
4	1	0	0	$\overline{A}\cup B\cup C$
6	1	1	0	$\overline{A}\cup\overline{B}\cup C$
7	1	1	1	$\overline{A}\cup\overline{B}\cup\overline{C}$

阴影对应的集合

$$\text{区域}035 = (\overline{A}\cap\overline{B}\cap\overline{C})\cup(\overline{A}\cap B\cap C)\cup A\cap\overline{B}\cap C$$
$$= (A\cup B\cup\overline{C})\cap(A\cup\overline{B}\cup C)\cap(\overline{A}\cup B\cup C)\cap(\overline{A}\cup\overline{B}\cup C)\cap(\overline{A}\cup\overline{B}\cup\overline{C})$$

Example —求阴影部分的集合解析式



成真指派与极小项

	A	B	C	*Minterm*
0	0	0	0	$\overline{A}\cap\overline{B}\cap\overline{C}$
3	0	1	1	$\overline{A}\cap B\cap C$
5	1	0	1	$A\cap\overline{B}\cap C$

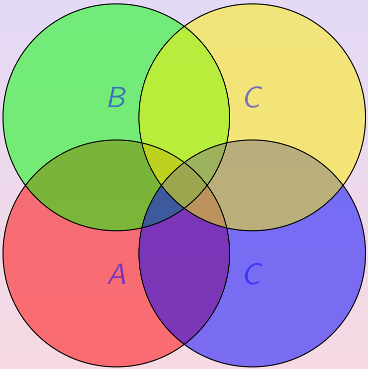
成假指派与极大项

	A	B	C	*Maxterm*
1	0	0	1	$A\cup B\cup\overline{C}$
2	0	1	0	$A\cup\overline{B}\cup C$
4	1	0	0	$\overline{A}\cup B\cup C$
6	1	1	0	$\overline{A}\cup\overline{B}\cup C$
7	1	1	1	$\overline{A}\cup\overline{B}\cup\overline{C}$

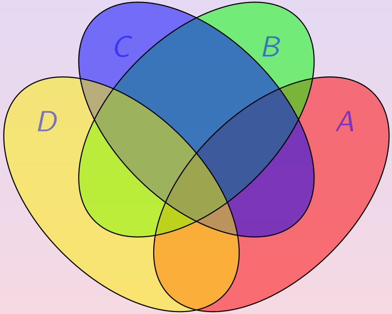
阴影对应的集合

$$\text{区域}035 = (\overline{A}\cap\overline{B}\cap\overline{C})\cup(\overline{A}\cap B\cap C)\cup A\cap\overline{B}\cap C$$
$$= (A\cup B\cup\overline{C})\cap(A\cup\overline{B}\cup C)\cap(\overline{A}\cup B\cup C)\cap(\overline{A}\cup\overline{B}\cup C)\cap(\overline{A}\cup\overline{B}\cup\overline{C})$$

更多的文氏图



(a) 非文氏图(仅有14个区域)



(b) 4个集合文氏图(16个区域)

环和

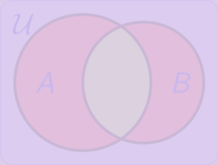
Definition (环和, Symmetric Difference, XOR in logic)

$$
\begin{aligned}
A \oplus B &\triangleq (A \cap \overline{B}) \cup (\overline{A} \cap B) \\
&= (A \cup B) \cap (\overline{A} \cup \overline{B}) \\
&= (A \cup B) - (A \cap B)
\end{aligned}
$$

环和

Definition (环和, Symmetric Difference, XOR in logic)

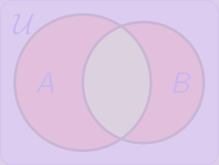
$$
\begin{aligned}
A \oplus B &\triangleq (A \cap \overline{B}) \cup (\overline{A} \cap B) \\
&= (A \cup B) \cap (\overline{A} \cup \overline{B}) \\
&= (A \cup B) - (A \cap B)
\end{aligned}
$$



环和

Definition (环和, Symmetric Difference, XOR in logic)

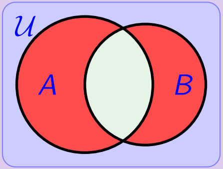
$$
\begin{aligned}
A \oplus B &\triangleq (A \cap \overline{B}) \cup (\overline{A} \cap B) \\
&= (A \cup B) \cap (\overline{A} \cup \overline{B}) \\
&= (A \cup B) - (A \cap B)
\end{aligned}
$$



环和

Definition (环和, Symmetric Difference, XOR in logic)

$$
\begin{aligned}
A \oplus B &\triangleq (A \cap \overline{B}) \cup (\overline{A} \cap B) \\
&= (A \cup B) \cap (\overline{A} \cup \overline{B}) \\
&= (A \cup B) - (A \cap B)
\end{aligned}
$$



环积

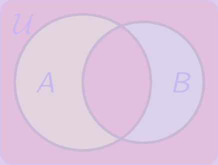
Definition (环积, XNOR in logic)

$$
\begin{aligned}
A \otimes B &\triangleq \overline{A \oplus B} \\
&= (\overline{A} \cap \overline{B}) \cup (A \cap B) \\
&= (\overline{A} \cup B) \cap (A \cup \overline{B})
\end{aligned}
$$

环积

Definition (环积, XNOR in logic)

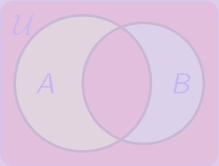
$$\begin{aligned} A \otimes B &\triangleq \overline{A \oplus B} \\ &= (\overline{A} \cap \overline{B}) \cup (A \cap B) \\ &= (\overline{A} \cup B) \cap (A \cup \overline{B}) \end{aligned}$$



环积

Definition (环积, XNOR in logic)

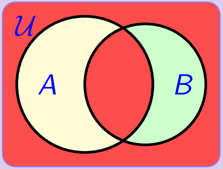
$$\begin{aligned} A \otimes B &\triangleq \overline{A \oplus B} \\ &= (\overline{A} \cap \overline{B}) \cup (A \cap B) \\ &= (\overline{A} \cup B) \cap (A \cup \overline{B}) \end{aligned}$$



环积

Definition (环积, XNOR in logic)

$$
\begin{aligned}
A \otimes B \triangleq \overline{A \oplus B} &= (\overline{A} \cap \overline{B}) \cup (A \cap B) \\
&= (\overline{A} \cup B) \cap (A \cup \overline{B})
\end{aligned}
$$



相关性质

Theorem

① 的图示证明见右图.

相关性质

Theorem

1. $\overline{A}\oplus\overline{B}=A\oplus B$;
2. $A\oplus A=\emptyset$;
3. $(A\oplus B)\oplus C=A\oplus(B\oplus C)$;
4. $\overline{A}\otimes\overline{B}=A\otimes B$;
5. $A\otimes A=U$;
6. $(A\otimes B)\otimes C=A\otimes(B\otimes C)$;
7. $C\cap(A\oplus B)=(C\cap A)\oplus(C\cap B)$.

⑦ 的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

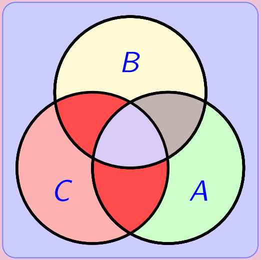
⑦的图示证明见右图.

相关性质

Theorem

1. $\overline{A} \oplus \overline{B} = A \oplus B$;
2. $A \oplus A = \varnothing$;
3. $(A \oplus B) \oplus C = A \oplus (B \oplus C)$;
4. $\overline{A} \otimes \overline{B} = A \otimes B$;
5. $A \otimes A = \mathcal{U}$;
6. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$;
7. $C \cap (A \oplus B) = (C \cap A) \oplus (C \cap B)$;

⑦的图示证明见右图.



无限交和无限并(1/2)

Definition

I是一个集合，并且对$\forall i \in I$都有一个集合S_i与之对应，称这样的集合为指标集(Index).

Example

- $I \triangleq \mathbb{N}, \forall n \in \mathbb{N}, S_n \triangleq \{x \mid x \in \mathbb{N} \wedge x \geqslant n\}$
- $I \triangleq \mathbb{R}^+ - \{0\}, \forall x \in \mathbb{R}^+ - \{0\}, S_x = [1/x, +\infty[$

无限交和无限并(1/2)

Definition

I是一个集合，并且对$\forall i \in I$都有一个集合S_i与之对应，称这样的集合为指标集(Index).

Example

- $I \triangleq \mathbb{N}, \forall n \in \mathbb{N}, \ S_n \triangleq \{x \mid x \in \mathbb{N} \wedge x \geq n\}$
- $I \triangleq \mathbb{R}^+ - \{0\}, \forall x \in \mathbb{R}^+ - \{0\}, \ S_x = [1/x, +\infty[$

无限交和无限并(1/2)

Definition

I是一个集合，并且对$\forall i \in I$都有一个集合S_i与之对应，称这样的集合为指标集(Index).

Example

- $I \triangleq \mathbb{N},\ \forall n \in \mathbb{N},\ S_n \triangleq \{\, x \mid x \in \mathbb{N} \wedge x \geqslant n \,\}$
- $I \triangleq \mathbb{R}^+ - \{\, 0 \,\},\ \forall x \in \mathbb{R}^+ - \{\, 0 \,\},\ S_x = [1/x, +\infty[$

无限交和无限并(1/2)

Definition

I是一个集合，并且对$\forall i \in I$都有一个集合S_i与之对应，称这样的集合为指标集(Index).

Example

- $I \triangleq \mathbb{N}, \forall n \in \mathbb{N},\ S_n \triangleq \{\, x \,|\, x \in \mathbb{N} \wedge x \geqslant n \,\}$
- $I \triangleq \mathbb{R}^+ - \{\, 0 \,\}, \forall x \in \mathbb{R}^+ - \{\, 0 \,\},\ S_x = [1/x, +\infty[$

无限交和无限并(2/2)

Definition (无限交和无限并)

$$\bigcap_{i \in I} S_i \triangleq \{\, x \mid \forall i (i \in I \to x \in S_i) \,\}, \quad \bigcup_{i \in I} S_i \triangleq \{\, x \mid \exists i (i \in I \land x \in S_i) \,\};$$

Example

Remark

无限交和无限并(2/2)

Definition (无限交和无限并)

$$\bigcap_{i\in I} S_i \triangleq \{\, x \mid \forall i(i\in I \to x\in S_i)\,\}, \quad \bigcup_{i\in I} S_i \triangleq \{\, x \mid \exists i(i\in I \land x\in S_i)\,\};$$

Example

- $\bigcap_{i\in\mathbb{N}} \{\, n \mid n\in\mathbb{N} \land n \geq i\,\} = \varnothing$;
- $\bigcup_{x\in\mathbb{R}^+ - \{0\}} [\frac{1}{x}, +\infty[=]0, +\infty[$;
- 设$\mathcal{I}$是所有的谓词公式解释集合，则：
$$\text{永真公式} = \bigcap_{I\in\mathcal{I}} \{\, F \mid F\in WFF \land F|_I = 1\,\}$$

Remark

无限交和无限并(2/2)

Definition (无限交和无限并)

$$\bigcap_{i\in I} S_i \triangleq \{\, x \mid \forall i(i \in I \to x \in S_i)\,\}, \quad \bigcup_{i\in I} S_i \triangleq \{\, x \mid \exists i(i \in I \land x \in S_i)\,\};$$

Example

- $\bigcap_{i\in\mathbb{N}} \{\, n \mid n \in \mathbb{N} \land n \geqslant i \,\} = \varnothing$;
- $\bigcup_{x\in\mathbb{R}^+ - \{0\}} [\frac{1}{x}, +\infty[=]0, +\infty[$;
- 设$\mathcal{I}$是所有的谓词公式解释集合，则：
  $$\text{永真公式} = \bigcap_{I\in\mathcal{I}} \{\, F \mid F \in WFF \land F|_I = 1 \,\}$$

Remark

无限交和无限并(2/2)

Definition (无限交和无限并)

$$\bigcap_{i\in I} S_i \triangleq \{\, x \mid \forall i(i\in I \rightarrow x\in S_i)\,\},\quad \bigcup_{i\in I} S_i \triangleq \{\, x \mid \exists i(i\in I \wedge x\in S_i)\,\};$$

Example

- $\bigcap_{i\in\mathbb{N}}\{\, n \mid n\in\mathbb{N} \wedge n\geqslant i\,\} = \varnothing$;
- $\bigcup_{x\in\mathbb{R}^+-\{\,0\,\}} [\frac{1}{x}, +\infty[=]0, +\infty[$;
- 设$\mathcal{I}$是所有的谓词公式解释集合，则：
  $$\text{永真公式} = \bigcap_{I\in\mathcal{I}}\{\, F \mid F\in WFF \wedge F|_I = 1\,\}$$

Remark

无限交和无限并(2/2)

Definition (无限交和无限并)

$$\bigcap_{i\in I} S_i \triangleq \{\, x \mid \forall i (i \in I \rightarrow x \in S_i) \,\}, \qquad \bigcup_{i\in I} S_i \triangleq \{\, x \mid \exists i (i \in I \wedge x \in S_i) \,\};$$

Example

- $\bigcap_{i\in\mathbb{N}} \{\, n \mid n \in \mathbb{N} \wedge n \geqslant i \,\} = \varnothing;$
- $\bigcup_{x\in\mathbb{R}^+ - \{0\}} [\frac{1}{x}, +\infty[=]0, +\infty[;$
- 设$\mathcal{I}$是所有的谓词公式解释集合，则：
  $$\text{永真公式} = \bigcap_{I\in\mathcal{I}} \{\, F \mid F \in WFF \wedge F|_I = 1 \,\}$$

Remark

相关性质

Theorem

1. $A \cup \left(\bigcap_{i \in I} S_i \right) = \bigcap_{i \in I} (A \cup S_i);$
2. $A \cap \left(\bigcup_{i \in I} S_i \right) = \bigcup_{i \in I} (A \cap S_i);$
3. $if\ \forall i \in I,\ A \subseteq S_i,\ then,\ A \subseteq \bigcap_{i \in I} S_i;$
4. $if\ \forall i \in I,\ S_i \subseteq A,\ then,\ \bigcup_{i \in I} S_i \subseteq A.$

相关性质

Theorem

1. $A \cup \left(\bigcap_{i \in I} S_i \right) = \bigcap_{i \in I} (A \cup S_i)$;
2. $A \cap \left(\bigcup_{i \in I} S_i \right) = \bigcup_{i \in I} (A \cap S_i)$;
3. $if\ \forall i \in I,\ A \subseteq S_i,\ then,\ A \subseteq \bigcap_{i \in I} S_i$;
4. $if\ \forall i \in I,\ S_i \subseteq A,\ then,\ \bigcup_{i \in I} S_i \subseteq A$.

相关性质

Theorem

1. $A \cup \left(\bigcap_{i \in I} S_i\right) = \bigcap_{i \in I}(A \cup S_i)$;
2. $A \cap \left(\bigcup_{i \in I} S_i\right) = \bigcup_{i \in I}(A \cap S_i)$;
3. $if\ \forall i \in I,\ A \subseteq S_i,\ then,\ A \subseteq \bigcap_{i \in I} S_i$;
4. $if\ \forall i \in I,\ S_i \subseteq A,\ then,\ \bigcup_{i \in I} S_i \subseteq A$.

相关性质

Theorem

1. $A \cup \left(\bigcap_{i \in I} S_i \right) = \bigcap_{i \in I} (A \cup S_i)$;
2. $A \cap \left(\bigcup_{i \in I} S_i \right) = \bigcup_{i \in I} (A \cap S_i)$;
3. *if* $\forall i \in I,\ A \subseteq S_i$, *then*, $A \subseteq \bigcap_{i \in I} S_i$;
4. *if* $\forall i \in I,\ S_i \subseteq A$, *then*, $\bigcup_{i \in I} S_i \subseteq A$.

相关性质

Theorem

1. $A \cup \left(\bigcap_{i \in I} S_i \right) = \bigcap_{i \in I} (A \cup S_i)$;
2. $A \cap \left(\bigcup_{i \in I} S_i \right) = \bigcup_{i \in I} (A \cap S_i)$;
3. $if\ \forall i \in I,\ A \subseteq S_i,\ then,\ A \subseteq \bigcap_{i \in I} S_i$;
4. $if\ \forall i \in I,\ S_i \subseteq A,\ then,\ \bigcup_{i \in I} S_i \subseteq A$.

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i \right) \\
\iff\ & x \in A \ \vee \ x \in \left(\bigcap_{i \in I} S_i \right) \\
\iff\ & x \in A \ \vee \ \forall i (i \in I \to x \in S_i) \\
\iff\ & \forall i (x \in A \ \vee \ (i \in I \to x \in S_i)) \\
\iff\ & \forall i (i \in I \to x \in A \vee x \in S_i) \\
\iff\ & \forall i (i \in I \to x \in A \cup S_i) \\
\iff\ & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

$\square$

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i \right) \\
\iff\ & x \in A \lor x \in \left(\bigcap_{i \in I} S_i \right) \\
\iff\ & x \in A \lor \forall i (i \in I \to x \in S_i) \\
\iff\ & \forall i (x \in A \lor (i \in I \to x \in S_i)) \\
\iff\ & \forall i (i \in I \to x \in A \lor x \in S_i) \\
\iff\ & \forall i (i \in I \to x \in A \cup S_i) \\
\iff\ & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

□

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i \right) \\
\Longleftrightarrow\ & x \in A \ \vee \ x \in \left(\bigcap_{i \in I} S_i \right) \\
\Longleftrightarrow\ & x \in A \ \vee \ \forall i (i \in I \to x \in S_i) \\
\Longleftrightarrow\ & \forall i (x \in A \ \vee \ (i \in I \to x \in S_i)) \\
\Longleftrightarrow\ & \forall i (i \in I \to x \in A \vee x \in S_i) \\
\Longleftrightarrow\ & \forall i (i \in I \to x \in A \cup S_i) \\
\Longleftrightarrow\ & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

□

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i \right) \\
\iff & x \in A \lor x \in \left(\bigcap_{i \in I} S_i \right) \\
\iff & x \in A \lor \forall i (i \in I \to x \in S_i) \\
\iff & \forall i (x \in A \lor (i \in I \to x \in S_i)) \\
\iff & \forall i (i \in I \to x \in A \lor x \in S_i) \\
\iff & \forall i (i \in I \to x \in A \cup S_i) \\
\iff & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

□

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i \right) \\
\iff \quad & x \in A \ \vee \ x \in \left(\bigcap_{i \in I} S_i \right) \\
\iff \quad & x \in A \ \vee \ \forall i (i \in I \to x \in S_i) \\
\iff \quad & \forall i (x \in A \ \vee \ (i \in I \to x \in S_i)) \\
\iff \quad & \forall i (i \in I \to x \in A \vee x \in S_i) \\
\iff \quad & \forall i (i \in I \to x \in A \cup S_i) \\
\iff \quad & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

□

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i\right) \\
\iff\ & x \in A \lor x \in \left(\bigcap_{i \in I} S_i\right) \\
\iff\ & x \in A \lor \forall i (i \in I \to x \in S_i) \\
\iff\ & \forall i (x \in A \lor (i \in I \to x \in S_i)) \\
\iff\ & \forall i (i \in I \to x \in A \lor x \in S_i) \\
\iff\ & \forall i (i \in I \to x \in A \cup S_i) \\
\iff\ & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

□

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i \right) \\
\iff \quad & x \in A \vee x \in \left(\bigcap_{i \in I} S_i \right) \\
\iff \quad & x \in A \vee \forall i (i \in I \rightarrow x \in S_i) \\
\iff \quad & \forall i (x \in A \vee (i \in I \rightarrow x \in S_i)) \\
\iff \quad & \forall i (i \in I \rightarrow x \in A \vee x \in S_i) \\
\iff \quad & \forall i (i \in I \rightarrow x \in A \cup S_i) \\
\iff \quad & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

□

并对无限交的分配律(①)的证明

Proof.

$$
\begin{aligned}
& x \in A \cup \left(\bigcap_{i \in I} S_i \right) \\
\iff\ & x \in A \vee x \in \left(\bigcap_{i \in I} S_i \right) \\
\iff\ & x \in A \vee \forall i (i \in I \to x \in S_i) \\
\iff\ & \forall i (x \in A \vee (i \in I \to x \in S_i)) \\
\iff\ & \forall i (i \in I \to x \in A \vee x \in S_i) \\
\iff\ & \forall i (i \in I \to x \in A \cup S_i) \\
\iff\ & x \in \bigcap_{i \in I} (A \cup S_i)
\end{aligned}
$$

□

集合的程序实现

Description (设计要点)

- *抽象类型：所有的能够对两个元素进行比较的数据类型都可以是集合的元素；*
- 支持集合的创建，元素与集合及集合与集合上的运算；
- *支持对集合上的所有元素进行枚举的操作；*
- *C++ STL ;*
- *Java Interface Set.*

集合的程序实现

Description (设计要点)

- *抽象类型：所有的能够对两个元素进行比较的数据类型都可以是集合的元素；*
- *支持集合的创建，元素与集合及集合与集合上的运算；*
- *支持对集合上的所有元素进行枚举的操作；*
- *C++ STL ;*
- *Java Interface Set.*

集合的程序实现

Description (设计要点)

- 抽象类型：所有的能够对两个元素进行比较的数据类型都可以是集合的元素；
- 支持集合的创建，元素与集合及集合与集合上的运算；
- 支持对集合上的所有元素进行枚举的操作；
- C++ STL；
- Java Interface Set.

集合的程序实现

Description (设计要点)

- *抽象类型：所有的能够对两个元素进行比较的数据类型都可以是集合的元素；*
- *支持集合的创建，元素与集合及集合与集合上的运算；*
- *支持对集合上的所有元素进行枚举的操作；*
- *C++ STL ;*
- *Java Interface Set.*

集合的程序实现

Description (设计要点)

- *抽象类型：所有的能够对两个元素进行比较的数据类型都可以是集合的元素；*
- *支持集合的创建，元素与集合及集合与集合上的运算；*
- *支持对集合上的所有元素进行枚举的操作；*
- *C++ STL ;*
- *Java Interface Set.*

集合的程序实现

Description (设计要点)

- *抽象类型：所有的能够对两个元素进行比较的数据类型都可以是集合的元素；*
- *支持集合的创建，元素与集合及集合与集合上的运算；*
- *支持对集合上的所有元素进行枚举的操作；*
- *C++ STL ;*
- *Java Interface Set.*

集合的C语言程序实现

```c
/* set.h from "C: A Reference Manual" by S. Harbison */
#include <limits.h>    /* defines CHAR_BIT */
typedef unsigned int SET;
#define SET_BITS                 (sizeof(SET)*CHAR_BIT)
#define check(i)                 (((unsigned)(i)) < SET_BITS)
#define emptyset                 ((SET) 0)
#define add(set,i)               ((set) | singleset (i))
#define singleset(i)             (((SET) 1) << (i))
#define intersect(set1,set2)     ((set1) & (set2))
#define unions(set1,set2)        ((set1) | (set2))
#define setdiff(set1,set2)       ((set1) ^ (set2))
#define element(i,set)           (singleset((i)) & (set))
#define forallelements(j,s) \
   for((j)=0; (j)<SET_BITS; ++(j)) if (element((j),(s)))
#define first_set_of_n_elements(n)  (SET)((1<<(n))-1)
```

求与集合x有相同的元素的个数并且其二进制数正好比x大的集合

```
SET next_set_of_n_elements(SET x)
{
/*
      if x                 == 001011001111000, then
      smallest             == 000000000001000
      ripple               == 001011010000000
      new_smallest         == 000000010000000
      ones                 == 000000000000111
      the returned value   == 001011010000111
 */
    SET smallest, ripple, new_smallest, ones;
    if (x == emptyset) return x;
    smallest     = (x & -x);
    ripple       = x + smallest;
    new_smallest = (ripple & -ripple);
    ones         = ((new_smallest / smallest) >> 1) - 1;
    return (ripple | ones);
}
```

1. 基本概念
   - 精确的集合语言
   - 什么是集合
   - 集合的包含关系
2. 集合上的运算
   - 运算的定义
   - 集合运算的恒等式和不等式
   - 补运算
   - 文氏图和范式的关系
   - 集合运算的扩充
   - 无限交和无限并
   - 集合的程序实现
3. **集合的构造**
   - 幂集合
   - 自然数 — 集合的递归构造
   - 乘积集合
   - 字符串集合
   - 容斥原理

幂集合

Axiom (Power Set)

对每个集合S，都存在一个集合，该集合的元素正好是集合S的子集合，称之为S的幂集合，记为：$\mathscr{P}(S)$(or 2^S)；
$\mathscr{P}(S) = \{\, T \mid T \subseteq S \,\}$.

Property

$$\emptyset \in \mathscr{P}(S) \quad \wedge \quad S \in \mathscr{P}(S)$$

Example

幂集合

Axiom (Power Set)

对每个集合S，都存在一个集合，该集合的元素正好是集合S的子集合，称之为S的幂集合，记为：$\mathscr{P}(S)$ (or 2^S);
$\mathscr{P}(S) = \{\, T \mid T \subseteq S \,\}$.

Property

$$\emptyset \in \mathscr{P}(S) \quad \wedge \quad S \in \mathscr{P}(S)$$

Example

- $\mathscr{P}(\emptyset) = \{\emptyset\}$，注意：$\emptyset \neq \{\emptyset\}$，$\because \emptyset \in \{\emptyset\}$，$\emptyset \notin \emptyset$；
  $\mathscr{P}(\{\emptyset\}) = \{\emptyset, \{\emptyset\}\}$；
  $\mathscr{P}(\{\emptyset, \{\emptyset\}\}) = \{\emptyset, \{\emptyset\}, \{\{\emptyset\}\}, \{\emptyset, \{\emptyset\}\}\}$

幂集合

Axiom (Power Set)

对每个集合S，都存在一个集合，该集合的元素正好是集合S的子集合，称之为S的幂集合，记为：$\mathscr{P}(S)$(or 2^S);
$\mathscr{P}(S) = \{\, T \mid T \subseteq S \,\}$.

Property

$$\emptyset \in \mathscr{P}(S) \quad \wedge \quad S \in \mathscr{P}(S)$$

Example

- $\mathscr{P}(\emptyset) = \{\emptyset\}$，注意：$\emptyset \neq \{\emptyset\}, \because \emptyset \in \{\emptyset\}, \emptyset \notin \emptyset$;
- $\mathscr{P}(\{\emptyset\}) = \{\emptyset, \{\emptyset\}\}$;
- $\mathscr{P}(\{\emptyset, \{\emptyset\}\}) = \{\emptyset, \{\emptyset\}, \{\{\emptyset\}\}, \{\emptyset, \{\emptyset\}\}\}$.

幂集合

Axiom (Power Set)

对每个集合S，都存在一个集合，该集合的元素正好是集合S的子集合，称之为S的幂集合，记为：$\mathscr{P}(S)$(or 2^S);
$\mathscr{P}(S) = \{ T | T \subseteq S \}$.

Property

$$\varnothing \in \mathscr{P}(S) \quad \wedge \quad S \in \mathscr{P}(S)$$

Example

- $\mathscr{P}(\varnothing) = \{ \varnothing \}$，注意：$\varnothing \neq \{ \varnothing \}, \because \varnothing \in \{ \varnothing \}, \varnothing \notin \varnothing$;
- $\mathscr{P}(\{ \varnothing \}) = \{ \varnothing, \{ \varnothing \} \}$;
- $\mathscr{P}(\{ \varnothing, \{ \varnothing \} \}) = \{ \varnothing, \{ \varnothing \}, \{ \{ \varnothing \} \}, \{ \varnothing, \{ \varnothing \} \} \}$.

幂集合

Axiom (Power Set)

对每个集合S，都存在一个集合，该集合的元素正好是集合S的子集合，称之为S的幂集合，记为：$\mathscr{P}(S)$(or 2^S);
$\mathscr{P}(S) = \{ T | T \subseteq S \}$.

Property

$$\varnothing \in \mathscr{P}(S) \quad \wedge \quad S \in \mathscr{P}(S)$$

Example

- $\mathscr{P}(\varnothing) = \{ \varnothing \}$，注意：$\varnothing \neq \{ \varnothing \}, \because \varnothing \in \{ \varnothing \}, \varnothing \notin \varnothing$;
- $\mathscr{P}(\{ \varnothing \}) = \{ \varnothing, \{ \varnothing \} \}$;
- $\mathscr{P}(\{ \varnothing, \{ \varnothing \} \}) = \{ \varnothing, \{ \varnothing \}, \{ \{ \varnothing \} \}, \{ \varnothing, \{ \varnothing \} \} \}$.

幂集合

Axiom (Power Set)

对每个集合S，都存在一个集合，该集合的元素正好是集合S的子集合，称之为S的幂集合，记为：$\mathscr{P}(S)$(or 2^S);
$\mathscr{P}(S) = \{ T | T \subseteq S \}$.

Property

$$\varnothing \in \mathscr{P}(S) \quad \wedge \quad S \in \mathscr{P}(S)$$

Example

- $\mathscr{P}(\varnothing) = \{ \varnothing \}$，注意：$\varnothing \neq \{ \varnothing \}, \because \varnothing \in \{ \varnothing \}, \varnothing \notin \varnothing$;
- $\mathscr{P}(\{ \varnothing \}) = \{ \varnothing, \{ \varnothing \} \}$;
- $\mathscr{P}(\{ \varnothing, \{ \varnothing \} \}) = \{ \varnothing, \{ \varnothing \}, \{ \{ \varnothing \} \}, \{ \varnothing, \{ \varnothing \} \} \}$.

幂集合基数

Theorem

if S finite set, $|S| = n$, then $|\mathscr{P}(S)| = 2^n$.

幂集合基数

Theorem

if S finite set, $|S| = n$, then $|\mathscr{P}(S)| = 2^n$.

Example

- $|\mathscr{P}(\varnothing)| = |\{\varnothing\}| = 2^0 = 1$;
- $|\mathscr{P}(\{\varnothing\})| = |\{\varnothing, \{\varnothing\}\}| = 2^1 = 2$;
- $|\mathscr{P}(\{\varnothing, \{\varnothing\}\})| = 2^2 = 4$;
- 设$\mathcal{M}_n$是关于n个原子$P_1, P_2, \ldots, P_n$的极小项的集合，则$\mathscr{P}(\mathcal{M}_n)$对应与所有的成真指派，而$|\mathcal{M}_n| = 2^n$，$\therefore |\mathscr{P}(\mathcal{M}_n)| = 2^{2^n}$.

幂集合基数

Theorem

if S finite set, $|S| = n$, then $|\mathscr{P}(S)| = 2^n$.

Example

- $|\mathscr{P}(\varnothing)| = |\{\varnothing\}| = 2^0 = 1$;
- $|\mathscr{P}(\{\varnothing\})| = |\{\varnothing, \{\varnothing\}\}| = 2^1 = 2$;
- $|\mathscr{P}(\{\varnothing, \{\varnothing\}\})| = 2^2 = 4$;
- 设$\mathcal{M}_n$是关于n个原子$P_1, P_2, \ldots, P_n$的极小项的集合，则$\mathscr{P}(\mathcal{M}_n)$对应与所有的成真指派，而$|\mathcal{M}_n| = 2^n$，$\therefore |\mathscr{P}(\mathcal{M}_n)| = 2^{2^n}$.

幂集合基数

Theorem

if S finite set, $|S| = n$, then $|\mathscr{P}(S)| = 2^n$.

Example

- $|\mathscr{P}(\varnothing)| = |\{\varnothing\}| = 2^0 = 1$;
- $|\mathscr{P}(\{\varnothing\})| = |\{\varnothing, \{\varnothing\}\}| = 2^1 = 2$;
- $|\mathscr{P}(\{\varnothing, \{\varnothing\}\})| = 2^2 = 4$;
- 设$\mathcal{M}_n$是关于n个原子$P_1, P_2, \ldots, P_n$的极小项的集合，则$\mathscr{P}(\mathcal{M}_n)$对应与所有的成真指派，而$|\mathcal{M}_n| = 2^n$，$\therefore |\mathscr{P}(\mathcal{M}_n)| = 2^{2^n}$.

幂集合基数

Theorem

if S finite set, $|S| = n$, then $|\mathscr{P}(S)| = 2^n$.

Example

- $|\mathscr{P}(\varnothing)| = |\{\varnothing\}| = 2^0 = 1$;
- $|\mathscr{P}(\{\varnothing\})| = |\{\varnothing, \{\varnothing\}\}| = 2^1 = 2$;
- $|\mathscr{P}(\{\varnothing, \{\varnothing\}\})| = 2^2 = 4$;
- 设$\mathcal{M}_n$是关于n个原子$P_1, P_2, \ldots, P_n$的极小项的集合，则$\mathscr{P}(\mathcal{M}_n)$对应与所有的成真指派，而$|\mathcal{M}_n| = 2^n$，$\therefore |\mathscr{P}(\mathcal{M}_n)| = 2^{2^n}$.

幂集合基数

Theorem

if S finite set, $|S| = n$, then $|\mathscr{P}(S)| = 2^n$.

Example

- $|\mathscr{P}(\varnothing)| = |\{\varnothing\}| = 2^0 = 1$;
- $|\mathscr{P}(\{\varnothing\})| = |\{\varnothing, \{\varnothing\}\}| = 2^1 = 2$;
- $|\mathscr{P}(\{\varnothing, \{\varnothing\}\})| = 2^2 = 4$;
- 设$\mathcal{M}_n$是关于n个原子$P_1, P_2, \ldots, P_n$的极小项的集合，则$\mathscr{P}(\mathcal{M}_n)$对应与所有的成真指派，而$|\mathcal{M}_n| = 2^n$，$\therefore |\mathscr{P}(\mathcal{M}_n)| = 2^{2^n}$.

自然数的构造

Axiom (Infinity)

存在集合$\mathbb{N}$满足下条件:

Description

集合	数字
$\emptyset$	0
$\{\emptyset\}$	1
$\{\emptyset, \{\emptyset\}\}$	2
$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$	3
$\cdots$	
$\cdots$	n
$n \cup \{n\}$	$n+1$
$\cdots$	

自然数的构造

Axiom (Infinity)

存在集合$\mathbb{N}$满足下条件:

1. $\emptyset \in \mathbb{N}$;
2. *if* $n \in \mathbb{N}$, *then* $n \cup \{n\} \in \mathbb{N}$.

Description

集合	编号
$\emptyset$	0
$\{\emptyset\}$	1
$\{\emptyset, \{\emptyset\}\}$	2
$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$	3
$\cdots$	$\cdots$
$\cdots$	n
$n \cup \{n\}$	$n+1$
$\cdots$	$\cdots$

自然数的构造

Axiom (Infinity)

存在集合$\mathbb{N}$满足下条件:

1. $\varnothing \in \mathbb{N}$;
2. *if* $n \in \mathbb{N}$, *then* $n \cup \{n\} \in \mathbb{N}$;

自然数的构造

Axiom (Infinity)

存在集合$\mathbb{N}$满足下条件:

1. $\varnothing \in \mathbb{N}$;
2. *if* $n \in \mathbb{N}$, *then* $n \cup \{n\} \in \mathbb{N}$;

Description

集合	数字
$\varnothing$	0
$\{\varnothing\}$	1
$\{\varnothing, \{\varnothing\}\}$	2
$\{\varnothing, \{\varnothing\}, \{\varnothing, \{\varnothing\}\}\}$	3
$\vdots$	$\vdots$
	n
$n \cup \{n\}$	$n+1$

自然数的构造

Axiom (Infinity)

存在集合$\mathbb{N}$满足下条件:

1. $\emptyset \in \mathbb{N}$;
2. *if* $n \in \mathbb{N}$, *then* $n \cup \{n\} \in \mathbb{N}$;

Description

集合	编号
$\emptyset$	0
$\{\emptyset\}$	1
$\{\emptyset, \{\emptyset\}\}$	2
$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$	3
$\cdots\cdots$	$\cdots$
$\cdots\cdots$	n
$n \cup \{n\}$	$n+1$
$\cdots\cdots$	$\cdots$

自然数的构造

Axiom (Infinity)

存在集合$\mathbb{N}$*满足下条件:*

1. $\varnothing \in \mathbb{N}$;
2. *if* $n \in \mathbb{N}$, *then* $n \cup \{n\} \in \mathbb{N}$;

Description

集合	编号
$\varnothing$	0
$\{\varnothing\}$	1
$\{\varnothing, \{\varnothing\}\}$	2
$\{\varnothing, \{\varnothing\}, \{\varnothing, \{\varnothing\}\}\}$	3
$\cdots\cdots$	$\ldots$
$\cdots\cdots$	n
$n \cup \{n\}$	$n+1$
$\cdots\cdots$	$\ldots$

自然数的构造

Axiom (Infinity)

存在集合$\mathbb{N}$*满足下条件:*

1. $\emptyset \in \mathbb{N}$;
2. *if* $n \in \mathbb{N}$, *then* $n \cup \{n\} \in \mathbb{N}$;

Description

集合	编号
$\emptyset$	0
$\{\emptyset\}$	1
$\{\emptyset, \{\emptyset\}\}$	2
$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$	3
....	...
....	n
$n \cup \{n\}$	$n+1$
....	...

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s \rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

Example

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s\rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

Example

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s \rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

Example

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s\rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

Example

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s\rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

- 如果A是任意的集合，A^*中的元素是有限序列，即是数据结构中的链表;
- 如果A是有限字母表Σ，Σ^*是字符串集合，其元素称为词(words).

Example

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s\rangle \in A^*$(记为$as$);
3. [极小性条款]：A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

- *如果A是任意的集合，A^*中的元素是有限序列，即是数据结构中的链表；*
- *如果A是有限字母表Σ，Σ^*是字符串集合，其元素称为词(words).*

Example

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s \rangle \in A^*$(记为$as$);
3. [极小性条款]：A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

- *如果A是任意的集合，A^*中的元素是有限序列，即是数据结构中的链表；*
- *如果A是有限字母表Σ，Σ^*是字符串集合，其元素称为词(words).*

Example

- $\Sigma = \{a\}$，则，$\Sigma^* = \{\varepsilon, a, aa, aaa, \ldots\}$;
- $A = \mathbb{N}$，则，$\mathbb{N}^* = \{$自然数组成的链表$\}$.

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s \rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

- 如果A是任意的集合，A^*中的元素是有限序列，即是数据结构中的链表;
- 如果A是有限字母表Σ，Σ^*是字符串集合，其元素称为词(words).

Example

- $\Sigma = \{a\}$，则，$\Sigma^* = \{\varepsilon, a, aa, aaa, \ldots\}$;
- $A = \mathbb{N}$，则，$\mathbb{N}^* = \{$ 自然数组成的链表 $\}$.

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s \rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

- 如果A是任意的集合，A^*中的元素是有限序列，即是数据结构中的链表；
- 如果A是有限字母表Σ，Σ^*是字符串集合，其元素称为词(words).

Example

- $\Sigma = \{a\}$，则，$\Sigma^* = \{\varepsilon, a, aa, aaa, \ldots\}$;
- $A = \mathbb{N}$，则，$\mathbb{N}^* = \{$ 自然数组成的链表 $\}$.

字符串集合的归纳定义

Definition

设A是一个集合，则，A上的有限序列集合A^*归纳(递归)定义 (inductive (recursive) definition)：如下：

1. [inductive base]: $\varepsilon \in A^*$ (空串);
2. [inductive rule]: if $s \in A^* \wedge a \in A$, then $\langle a, s\rangle \in A^*$(记为$as$);
3. [极小性条款]: A^*的所有元素都由并仅由以上步骤在有限步生成.

Remark

- *如果A是任意的集合，A^*中的元素是有限序列，即是数据结构中的链表；*
- *如果A是有限字母表Σ，Σ^*是字符串集合，其元素称为词(words).*

Example

- $\Sigma = \{a\}$，则，$\Sigma^* = \{\varepsilon, a, aa, aaa, \ldots\}$;
- $A = \mathbb{N}$，则，$\mathbb{N}^* = \{$ 自然数组成的链表 $\}$.

注释

Remark

- *归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)*
- *有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款；*
- *根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:*

Example

注释

Remark

- *归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)*
- *有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款；*
- *根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:*

Example

注释

Remark

- *归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)*
- *有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款；*
- *根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:*

注释

Remark

- *归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)*
- *有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款；*
- *根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:*
  - *如程序语言是字符串集合的子集合，该子集合必须有是归纳结构，否则计算机将不能识别该语言；*
  - *所有的数据类型也必须是归纳结构，否则将不能创建和操作该结构的数据；*
  - *同样能够计算的函数也一定是可归纳定义的函数.*

Example

注释

Remark

- *归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)*
- *有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款；*
- *根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:*
  - *如程序语言是字符串集合的子集合，该子集合必须有是归纳结构，否则计算机将不能识别该语言；*
  - *所有的数据类型也必须是归纳结构，否则将不能创建和操作该结构的数据；*
  - *同样能够计算的函数也一定是可归纳定义的函数.*

Example

注释

Remark

- 归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)
- 有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款；
- 根据可计算性理论，计算机处理的对象一定是有归纳结构的集合：
  - 如程序语言是字符串集合的子集合，该子集合必须有是归纳结构，否则计算机将不能识别该语言；
  - 所有的数据类型也必须是归纳结构，否则将不能创建和操作该结构的数据；
  - 同样能够计算的函数也一定是可归纳定义的函数.

Example

注释

Remark

- *归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)*
- *有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款;*
- *根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:*
  - *如程序语言是字符串集合的子集合，该子集合必须有是归纳结构，否则计算机将不能识别该语言;*
  - *所有的数据类型也必须是归纳结构，否则将不能创建和操作该结构的数据;*
  - *同样能够计算的函数也一定是可归纳定义的函数.*

Example

- 命题公式和谓词公式的归纳定义;

注释

Remark

- 归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)
- 有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款;
- 根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:
  - 如程序语言是字符串集合的子集合，该子集合必须有是归纳结构，否则计算机将不能识别该语言;
  - 所有的数据类型也必须是归纳结构，否则将不能创建和操作该结构的数据;
  - 同样能够计算的函数也一定是可归纳定义的函数.

Example

- 命题公式和谓词公式的归纳定义;
- 数据结构的链表，栈，树等.

注释

Remark

- *归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)*
- *有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款；*
- *根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:*
  - *如程序语言是字符串集合的子集合，该子集合必须有是归纳结构，否则计算机将不能识别该语言；*
  - *所有的数据类型也必须是归纳结构，否则将不能创建和操作该结构的数据；*
  - *同样能够计算的函数也一定是可归纳定义的函数.*

Example

- 命题公式和谓词公式的归纳定义；
- 数据结构的链表，栈，树等.

注释

Remark

- 归纳定义一定是从最小的元素开始按照归纳规则逐步增长被定义的集合，直到用有限方法不能再增长(最小不动点)
- 有限集合也可看成归纳结构，其归纳基础是该有限集合，而没有归纳条款;
- 根据可计算性理论，计算机处理的对象一定是有归纳结构的集合:
  - 如程序语言是字符串集合的子集合，该子集合必须有是归纳结构，否则计算机将不能识别该语言;
  - 所有的数据类型也必须是归纳结构，否则将不能创建和操作该结构的数据;
  - 同样能够计算的函数也一定是可归纳定义的函数.

Example

- 命题公式和谓词公式的归纳定义;
- 数据结构的链表，栈，树等.

Example

Example

设$\Sigma = \{ a, b \}$, $S \triangleq \{ a^n b^n \mid n \in \mathbb{N} \} \subseteq \Sigma^*$, 其中: $a^0 \triangleq \varepsilon$, $a^{n+1} = aa^n$, 试写出S的归纳定义.

Description

- *Base:* $\varepsilon \in S$;
- *Inductive rules:* if $s \in S$, then $asb \in S$.

Example

Example

设$\Sigma = \{ a, b \}$, $S \triangleq \{ a^n b^n \mid n \in \mathbb{N} \} \subseteq \Sigma^*$, 其中: $a^0 \triangleq \varepsilon$, $a^{n+1} = aa^n$, 试写出S的归纳定义.

Description

- *Base:* $\varepsilon \in S$;
- *Inductive rule: if* $s \in S$, *then* $asb \in S$;

Example

Example

设$\Sigma = \{ a, b \}$, $S \triangleq \{ a^n b^n \mid n \in \mathbb{N} \} \subseteq \Sigma^*$, 其中: $a^0 \triangleq \varepsilon$, $a^{n+1} = aa^n$, 试写出S的归纳定义.

Description

- *Base:* $\varepsilon \in S$;
- *Inductive rule: if* $s \in S$, *then* $asb \in S$;

Example

Example

设$\Sigma = \{ a, b \}$, $S \triangleq \{ a^n b^n \mid n \in \mathbb{N} \} \subseteq \Sigma^*$, 其中: $a^0 \triangleq \varepsilon$, $a^{n+1} = aa^n$, 试写出S的归纳定义.

Description

- *Base:* $\varepsilon \in S$;
- *Inductive rule: if* $s \in S$, *then* $asb \in S$;

归纳证明

Description (结构归纳法Structural Induction)

设S是一个具有归纳结构的集合，$\forall x(x \in S \to P(x))$，当且仅当，下述两条件同时成立：

- Base: 对S递归定义中的基础集合B有：$\forall x(x \in B \to P(x))$
- Inductive rule: 对每个归纳条款$r(x,y)$(表示由x和y按照规则r生成的新元素)，有：$P(x) \wedge P(y) \to P(r(x,y))$.

Remark

归纳证明

Description (结构归纳法Structural Induction)

设S是一个具有归纳结构的集合，$\forall x(x \in S \rightarrow P(x))$，当且仅当，下述两条件同时成立：

- *Base:* 对S递归定义中的基础集合B有：$\forall x(x \in B \rightarrow P(x))$
- *Inductive rule:* 对每个归纳条款$r(x, y)$(表示由x和y按照规则r生成的新元素)，有：$P(x) \wedge P(y) \rightarrow P(r(x, y))$.

Remark

归纳证明

Description (结构归纳法Structural Induction)

设S是一个具有归纳结构的集合，$\forall x(x \in S \to P(x))$，当且仅当，下述两条件同时成立：

- *Base:* 对S递归定义中的基础集合B有：$\forall x(x \in B \to P(x))$
- *Inductive rule:* 对每个归纳条款$r(x, y)$(表示由x和y按照规则r生成的新元素)，有：$P(x) \wedge P(y) \to P(r(x, y))$.

Remark

- *if* $S = \mathbb{N}$, *then* $r(n) = n + 1$，此时称为数学归纳法(*Mathematical Induction*)；但是在计算机中使用更多的是针对一般归纳结构的结构归纳法；
- 两个条件必须同时成立，反例：$P(n) = n \geq 3$，则$\forall n(n \in \mathbb{N} \to P(n))$为假，而$\forall n(P(n) \to P(n+1))$为真

归纳证明

Description (结构归纳法Structural Induction)

设S是一个具有归纳结构的集合，$\forall x(x \in S \to P(x))$，当且仅当，下述两条件同时成立：

- Base：对S递归定义中的基础集合B有：$\forall x(x \in B \to P(x))$
- Inductive rule：对每个归纳条款$r(x, y)$(表示由x和y按照规则r生成的新元素)，有：$P(x) \land P(y) \to P(r(x, y))$.

Remark

- if $S = \mathbb{N}$, then $r(n) = n + 1$; 此时称为数学归纳法(Mathematical Induction); 但是在计算机中使用更多的是针对一般归纳结构的结构归纳法;
- 两个条件必须同时成立，反例：$P(n) = n > 3$ 则$\forall n(n \in \mathbb{N} \to P(n))$为假，而$\forall n(P(n) \to P(n+1))$为真.

归纳证明

Description (结构归纳法Structural Induction)

设S是一个具有归纳结构的集合，$\forall x(x \in S \to P(x))$，当且仅当，下述两条件同时成立：

- *Base:* 对S递归定义中的基础集合B有：$\forall x(x \in B \to P(x))$
- *Inductive rule:* 对每个归纳条款$r(x, y)$(表示由x和y按照规则r生成的新元素)，有：$P(x) \wedge P(y) \to P(r(x, y))$.

Remark

- *if* $S = \mathbb{N}$, *then* $r(n) = n + 1$; 此时称为数学归纳法(Mathematical Induction); 但是在计算机中使用更多的是针对一般归纳结构的结构归纳法;
- 两个条件必须同时成立，反例：$P(n) = n > 3$ 则$\forall n(n \in \mathbb{N} \to P(n))$为假，而$\forall n(P(n) \to P(n+1))$为真.

归纳证明

Description (结构归纳法Structural Induction)

设S是一个具有归纳结构的集合，$\forall x(x \in S \to P(x))$，当且仅当，下述两条件同时成立：

- *Base:* 对S递归定义中的基础集合B有：$\forall x(x \in B \to P(x))$
- *Inductive rule:* 对每个归纳条款$r(x, y)$(表示由x和y按照规则r生成的新元素)，有：$P(x) \wedge P(y) \to P(r(x, y))$.

Remark

- *if* $S = \mathbb{N}$, *then* $r(n) = n + 1$; 此时称为数学归纳法(*Mathematical Induction*); 但是在计算机中使用更多的是针对一般归纳结构的结构归纳法;
- 两个条件必须同时成立，反例：$P(n) = n > 3$
  则$\forall n(n \in \mathbb{N} \to P(n))$为假，而$\forall n(P(n) \to P(n+1))$为真.

Example(1/2)

Definition (Σ^*上的联结运算(Concatenation))

1. Base: $\forall t \in \Sigma^*, \varepsilon \cdot t = t$;
2. Induction rule: if $s \cdot t$ is defined, then $\forall a \in \Sigma^*, (as) \cdot t = a(s \cdot t)$

Definition (Σ^*上的字符串长度的定义)

Example

试证明: $\forall s, t \in \Sigma^*, \|s \cdot t\| = \|s\| + \|t\|$;

Example(1/2)

Definition (Σ^*上的联结运算(Concatenation))

1. Base: $\forall t \in \Sigma^*, \varepsilon \cdot t = t$;
2. Induction rule: if $s \cdot t$ is defined, then $\forall a \in \Sigma^*, (as) \cdot t = a(s \cdot t)$

Definition (Σ^*上的字符串长度的定义)

1. Base: $\|\varepsilon\| = 0$;
2. Induction rule: if $\|s\| = n$, then $\forall a \in \Sigma^*, \|as\| = n+1$

Example

试证明: $\forall s, t \in \Sigma^*, \|s \cdot t\| = \|s\| + \|t\|$;

Example(1/2)

Definition (Σ^*上的联结运算(Concatenation))

1. Base: $\forall t \in \Sigma^*, \varepsilon \cdot t = t$;
2. Induction rule: if $s \cdot t$ is defined, then
   $\forall a \in \Sigma\ , (as) \cdot t = a(s \cdot t)$

Definition (Σ^*上的字符串长度的定义)

1. Base: $\|\varepsilon\| = 0$;
2. Induction rule: if $\|s\| = n$, then $\forall a \in \Sigma$, $\|as\| = n+1$

Example

试证明: $\forall s, t \in \Sigma^*, \|s \cdot t\| = \|s\| + \|t\|$;

Example(1/2)

Definition (Σ^*上的联结运算(Concatenation))

1. Base: $\forall t \in \Sigma^*$, $\varepsilon \cdot t = t$;
2. Induction rule: if $s \cdot t$ is defined, then $\forall a \in \Sigma$, $(as) \cdot t = a(s \cdot t)$

Definition (Σ^*上的字符串长度的定义)

1. Base: $\|\varepsilon\| = 0$;
2. Induction rule: if $\|s\| = n$, then $\forall a \in \Sigma^*$, $\|as\| = n + 1$

Example

试证明: $\forall s, t \in \Sigma^*$, $\|s \cdot t\| = \|s\| + \|t\|$;

Example(1/2)

Definition (Σ^*上的联结运算(Concatenation))

1. Base: $\forall t \in \Sigma^*$, $\varepsilon \cdot t = t$;
2. Induction rule: if $s \cdot t$ is defined, then $\forall a \in \Sigma$, $(as) \cdot t = a(s \cdot t)$

Definition (Σ^*上的字符串长度的定义)

1. Base: $\|\varepsilon\| = 0$;
2. Induction rule: if $\|s\| = n$, then $\forall a \in \Sigma^*$, $\|as\| = n + 1$

Example

试证明: $\forall s, t \in \Sigma^*$, $\|s \cdot t\| = \|s\| + \|t\|$;

Example(1/2)

Definition (Σ^*上的联结运算(Concatenation))

1. Base: $\forall t \in \Sigma^*, \varepsilon \cdot t = t$;
2. Induction rule: if $s \cdot t$ is defined, then $\forall a \in \Sigma$, $(as) \cdot t = a(s \cdot t)$

Definition (Σ^*上的字符串长度的定义)

1. Base: $\|\varepsilon\| = 0$;
2. Induction rule: if $\|s\| = n$, then $\forall a \in \Sigma$, $\|as\| = n + 1$

Example

试证明: $\forall s, t \in \Sigma^*, \ \|s \cdot t\| = \|s\| + \|t\|$;

Example(1/2)

Definition (Σ^*上的联结运算(Concatenation))

1. Base: $\forall t \in \Sigma^*$, $\varepsilon \cdot t = t$;
2. Induction rule: if $s \cdot t$ is defined, then
   $\forall a \in \Sigma$, $(as) \cdot t = a(s \cdot t)$

Definition (Σ^*上的字符串长度的定义)

1. Base: $\|\varepsilon\| = 0$;
2. Induction rule: if $\|s\| = n$, then $\forall a \in \Sigma$, $\|as\| = n + 1$

Example

试证明: $\forall s, t \in \Sigma^*$, $\|s \cdot t\| = \|s\| + \|t\|$;

Example(2/2)

Proof.

对s用归纳法:

1. Base: if $s = \varepsilon$, then
$$\|s \cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s \in \Sigma^*$, if $\|s \cdot t\| = \|s\| + \|t\|$, then $\forall a \in \Sigma$:
$$\begin{aligned}
&\|(as) \cdot t\| \\
=\ & \|a(s \cdot t)\| && (\cdot\text{的定义}) \\
=\ & 1 + \|s \cdot t\| && (\|\ \|\text{的定义}) \\
=\ & 1 + \|s\| + \|t\| && (\text{假设}) \\
=\ & \|as\| + \|t\| && (\|\ \|\text{的定义})
\end{aligned}$$

$\square$

Example(2/2)

Proof.

对s用归纳法:

1. Base: if $s = \varepsilon$, then
   $$\|s \cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s \in \Sigma^*$, if $\|s \cdot t\| = \|s\| + \|t\|$, then $\forall a \in \Sigma$:
   $$\begin{aligned} & \|(as) \cdot t\| \\ = & \|a(s \cdot t)\| && (\cdot\text{的定义}) \\ = & 1 + \|s \cdot t\| && (\|\ \|\text{的定义}) \\ = & 1 + \|s\| + \|t\| && (\text{假设}) \\ = & \|as\| + \|t\| && (\|\ \|\text{的定义}) \end{aligned}$$

$\square$

Example(2/2)

Proof.

对s用归纳法:

1. Base: if $s=\varepsilon$, then
$$\|s\cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s \in \Sigma^*$, if $\|s\cdot t\| = \|s\| + \|t\|$, then $\forall a \in \Sigma$:
$$\begin{aligned} &\|(as)\cdot t\| && \\ =\ & \|a(s\cdot t)\| && (\cdot\text{的定义}) \\ =\ & 1+\|s\cdot t\| && (\|\ \|\text{的定义}) \\ =\ & 1+\|s\|+\|t\| && (\text{假设}) \\ =\ & \|as\|+\|t\| && (\|\ \|\text{的定义}) \end{aligned}$$

$\square$

Example(2/2)

Proof.

对s用归纳法:

1. Base: if $s=\varepsilon$, then
$$\|s\cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s\in\Sigma^*$, if $\|s\cdot t\| = \|s\| + \|t\|$, then $\forall a\in\Sigma$:

$$\begin{aligned}
&\|(as)\cdot t\| \\
&= \|a(s\cdot t)\| && (\cdot\text{的定义}) \\
&= 1+\|s\cdot t\| && (\|\ \|\text{的定义}) \\
&= 1+\|s\|+\|t\| && (\text{假设}) \\
&= \|as\|+\|t\| && (\|\ \|\text{的定义})
\end{aligned}$$

$\square$

Example(2/2)

Proof.

对s用归纳法:

1. Base: if $s = \varepsilon$, then
$$\|s \cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s \in \Sigma^*$, if $\|s \cdot t\| = \|s\| + \|t\|$, then $\forall a \in \Sigma$:

$$\begin{aligned} & \|(as) \cdot t\| & \\ = & \|a(s \cdot t)\| & (\cdot\text{的定义}) \\ = & 1 + \|s \cdot t\| & (\|\ \|\text{的定义}) \\ = & 1 + \|s\| + \|t\| & (\text{假设}) \\ = & \|as\| + \|t\| & (\|\ \|\text{的定义}) \end{aligned}$$

$\square$

Example(2/2)

Proof.

对s用归纳法:

1. Base: if $s = \varepsilon$, then
$$\|s \cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s \in \Sigma^*$, if $\|s \cdot t\| = \|s\| + \|t\|$, then $\forall a \in \Sigma$:

$$\begin{aligned} & \|(as) \cdot t\| \\ = & \|a(s \cdot t)\| && (\cdot\text{的定义}) \\ = & 1 + \|s \cdot t\| && (\|\ \|\text{的定义}) \\ = & 1 + \|s\| + \|t\| && (\text{假设}) \\ = & \|as\| + \|t\| && (\|\ \|\text{的定义}) \end{aligned}$$

$\square$

Example(2/2)

Proof.

对s用归纳法：

1. Base: if $s=\varepsilon$, then
$$\|s\cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s\in\Sigma^*$, if $\|s\cdot t\| = \|s\| + \|t\|$, then $\forall a\in\Sigma$:

$$\begin{aligned}
& \|(as)\cdot t\| \\
= & \|a(s\cdot t)\| && (\cdot\text{的定义}) \\
= & 1+\|s\cdot t\| && (\|\ \|\text{的定义}) \\
= & 1+\|s\|+\|t\| && (\text{假设}) \\
= & \|as\|+\|t\| && (\|\ \|\text{的定义})
\end{aligned}$$

□

Example(2/2)

Proof.

对s用归纳法：

1. Base: if $s=\varepsilon$, then
$$\|s\cdot t\| = \|t\| = \|\varepsilon\| + \|t\|;$$
2. 设$\forall s\in\Sigma^*$, if $\|s\cdot t\| = \|s\| + \|t\|$, then $\forall a\in\Sigma$:

$$\begin{aligned}
& \|(as)\cdot t\| & \\
={}& \|a(s\cdot t)\| & (\cdot\text{的定义}) \\
={}& 1+\|s\cdot t\| & (\|\ \|\text{的定义}) \\
={}& 1+\|s\|+\|t\| & (\text{假设}) \\
={}& \|as\|+\|t\| & (\|\ \|\text{的定义})
\end{aligned}$$

$\square$

序偶、n重组

Definition (序偶 Ordered pair)

序偶是一个包含两个元素的collection，其中一个（称之为第一分量）和另一个元素（称之为第二分量）是可区别的；如果第一分量是a，第二分量是b，记该序偶为：$\langle a, b\rangle$.

序偶、n重组

Definition (序偶 Ordered pair)

序偶是一个包含两个元素的collection，其中一个（称之为第一分量）和另一个元素（称之为第二分量）是可区别的；如果第一分量是a，第二分量是b，记该序偶为：$\langle a, b\rangle$.

Remark

- 序偶本质上是个集合，可以理解为：$\langle a, b\rangle \triangleq \{a, \{a, b\}\}$;
- 序偶不能看成是两个元素平等地在一个集合中：$\langle a, b\rangle \neq \{a, b\}$;
- $\langle a, b\rangle = \langle a', b'\rangle$ iff $a = a' \wedge b = b'$.

序偶、n重组

Definition (序偶 Ordered pair)

序偶是一个包含两个元素的collection，其中一个（称之为第一分量）和另一个元素（称之为第二分量）是可区别的；如果第一分量是a，第二分量是b，记该序偶为：$\langle a, b\rangle$.

Remark

- *序偶本质上是个集合，可以理解为*: $\langle a, b\rangle \triangleq \{a, \{a, b\}\}$;
- *序偶不能看成是两个元素平等地在一个集合中*: $\langle a, b\rangle \neq \{a, b\}$;
- $\langle a, b\rangle = \langle a', b'\rangle$ *iff* $a = a' \wedge b = b'$.

序偶、n重组

Definition (序偶 Ordered pair)

序偶是一个包含两个元素的collection，其中一个（称之为第一分量）和另一个元素（称之为第二分量）是可区别的；如果第一分量是a，第二分量是b，记该序偶为：$\langle a, b\rangle$.

Remark

- *序偶本质上是个集合，可以理解为:* $\langle a, b\rangle \triangleq \{a, \{a, b\}\}$;
- *序偶不能看成是两个元素平等地在一个集合中:* $\langle a, b\rangle \neq \{a, b\}$;
- $\langle a, b\rangle = \langle a', b'\rangle$ *iff* $a = a' \wedge b = b'$.

序偶、n重组

Definition (序偶 Ordered pair)

序偶是一个包含两个元素的collection，其中一个（称之为第一分量）和另一个元素（称之为第二分量）是可区别的；如果第一分量是a，第二分量是b，记该序偶为：$\langle a, b\rangle$.

Remark

- *序偶本质上是个集合，可以理解为：*$\langle a, b\rangle \triangleq \{a, \{a, b\}\}$;
- *序偶不能看成是两个元素平等地在一个集合中：* $\langle a, b\rangle \neq \{a, b\}$;
- $\langle a, b\rangle = \langle a', b'\rangle$ *iff* $a = a' \wedge b = b'$.

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶;
2. 任个$n(n \geq 3)$个元素$a_1, a_2, \ldots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义, 并记为:$\langle a_1, a_2, \ldots, a_{n-1} \rangle$, 则, n重组定义为: $\langle a_1, a_2, \ldots a_n \rangle \triangleq \langle \langle a_1, a_2, \ldots a_{n-1} \rangle, a_n \rangle$

Remark

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶;
2. 任个$n(n \geq 3)$个元素$a_1, a_2, \ldots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义, 并记为:$\langle a_1, a_2, \ldots, a_{n-1} \rangle$, 则, n重组定义为: $\langle a_1, a_2, \ldots a_n \rangle \triangleq \langle \langle a_1, a_2, \ldots a_{n-1} \rangle, a_n \rangle$

Remark

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶;
2. 任个$n(n \geqslant 3)$个元素$a_1, a_2, \ldots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义, 并记为:$\langle a_1, a_2, \ldots, a_{n-1} \rangle$, 则, n重组定义为: $\langle a_1, a_2, \ldots a_n \rangle \triangleq \langle \langle a_1, a_2, \ldots a_{n-1} \rangle, a_n \rangle$

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶;
2. 任个$n(n \geqslant 3)$个元素$a_1, a_2, \ldots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义, 并记为:$\langle a_1, a_2, \ldots, a_{n-1}\rangle$, 则，$n$重组定义为: $\langle a_1, a_2, \ldots a_n\rangle \triangleq \langle\langle a_1, a_2, \ldots a_{n-1}\rangle, a_n\rangle$

Remark

- $\langle a_1, a_2, a_3\rangle \triangleq \langle\langle a_1, a_2\rangle, a_3\rangle \neq \langle a_1, \langle a_2, a_3\rangle\rangle$;
- $\langle a_1, a_2, \ldots, a_n\rangle = \langle a_1', a_2', \ldots, a_n'\rangle$ *iff* $a_i = a_i'$, $i = 1, 2, \ldots, n$;
- *函数式程序设计语言将tuple作为内建的数据类型，称为乘积类型;*
- *序偶也是一个重要的数据结构, 如C++和Java的标准库都有的Vector container.*

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶;
2. 任个$n(n \geqslant 3)$个元素$a_1, a_2, \ldots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义, 并记为:$\langle a_1, a_2, \ldots, a_{n-1} \rangle$, 则, n重组定义为: $\langle a_1, a_2, \ldots a_n \rangle \triangleq \langle \langle a_1, a_2, \ldots a_{n-1} \rangle, a_n \rangle$

Remark

- $\langle a_1, a_2, a_3 \rangle \triangleq \langle \langle a_1, a_2 \rangle, a_3 \rangle \neq \langle a_1, \langle a_2, a_3 \rangle \rangle$;
- $\langle a_1, a_2, \ldots, a_n \rangle = \langle a_1', a_2', \ldots, a_n' \rangle$ *iff* $a_i = a_i'$, $i = 1, 2, \ldots, n$;
- *函数式程序设计语言将tuple作为内建的数据类型，称为乘积类型;*
- *序偶也是一个重要的数据结构, 如C++和Java的标准库都有的Vector container.*

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶；
2. 任个$n(n \geqslant 3)$个元素$a_1, a_2, \dots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义，并记为:$\langle a_1, a_2, \dots, a_{n-1}\rangle$, 则，$n$重组定义为：$\langle a_1, a_2, \dots a_n\rangle \triangleq \langle\langle a_1, a_2, \dots a_{n-1}\rangle, a_n\rangle$

Remark

- $\langle a_1, a_2, a_3\rangle \triangleq \langle\langle a_1, a_2\rangle, a_3\rangle \neq \langle a_1, \langle a_2, a_3\rangle\rangle$;
- $\langle a_1, a_2, \dots, a_n\rangle = \langle a_1', a_2', \dots, a_n'\rangle$ *iff* $a_i = a_i'$, $i = 1, 2, \dots, n$;
- *函数式程序设计语言将tuple作为内建的数据类型，称为乘积类型;*
- *序偶也是一个重要的数据结构，如C++和Java的标准库都有的Vector container.*

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶;
2. 任个$n(n \geqslant 3)$个元素$a_1, a_2, \ldots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义, 并记为:$\langle a_1, a_2, \ldots, a_{n-1}\rangle$, 则, n重组定义为: $\langle a_1, a_2, \ldots a_n\rangle \triangleq \langle\langle a_1, a_2, \ldots a_{n-1}\rangle, a_n\rangle$

Remark

- $\langle a_1, a_2, a_3\rangle \triangleq \langle\langle a_1, a_2\rangle, a_3\rangle \neq \langle a_1, \langle a_2, a_3\rangle\rangle$;
- $\langle a_1, a_2, \ldots, a_n\rangle = \langle a_1', a_2', \ldots, a_n'\rangle$ *iff* $a_i = a_i'$, $i = 1, 2, \ldots, n$;
- *函数式程序设计语言将tuple作为内建的数据类型, 称为乘积类型;*
- *序偶也是一个重要的数据结构, 如C++和Java的标准库都有的Vector container.*

n重组

Definition (n重组(元组) n-tuple, 归纳定义)

1. 二重组即序偶；
2. 任个$n(n \geqslant 3)$个元素$a_1, a_2, \ldots, a_n$, 设前$n-1$个元素的$n-1$重组已经定义，并记为:$\langle a_1, a_2, \ldots, a_{n-1} \rangle$, 则，$n$重组定义为：$\langle a_1, a_2, \ldots a_n \rangle \triangleq \langle \langle a_1, a_2, \ldots a_{n-1} \rangle, a_n \rangle$

Remark

- $\langle a_1, a_2, a_3 \rangle \triangleq \langle \langle a_1, a_2 \rangle, a_3 \rangle \neq \langle a_1, \langle a_2, a_3 \rangle \rangle$;
- $\langle a_1, a_2, \ldots, a_n \rangle = \langle a_1', a_2', \ldots, a_n' \rangle$ *iff* $a_i = a_i'$, $i = 1, 2, \ldots, n$;
- *函数式程序设计语言将tuple作为内建的数据类型，称为乘积类型;*
- *序偶也是一个重要的数据结构，如C++和Java的标准库都有的Vector container.*

乘积集合

Definition (叉积 笛卡儿乘积集合 Cartesian Product)

$$A \times B \triangleq \{\langle a, b\rangle \mid a \in A \wedge b \in B\}$$

$$\begin{aligned} A_1 \times A_2 \times \cdots \times A_n &\triangleq (A_1 \times A_2 \times \cdots \times A_{n-1}) \times A_n \\ &= \{\langle a_1, a_2, \ldots, a_n\rangle \mid a_i \in A_i,\ i = 1, 2, \ldots, n\} \end{aligned}$$

Remark

乘积集合

Definition (叉积 笛卡儿乘积集合 Cartesian Product)

$$A \times B \triangleq \{\langle a, b\rangle \mid a \in A \wedge b \in B\}$$

$$\begin{aligned} A_1 \times A_2 \times \cdots \times A_n &\triangleq (A_1 \times A_2 \times \cdots \times A_{n-1}) \times A_n \\ &= \{\langle a_1, a_2, \ldots, a_n\rangle \mid a_i \in A_i,\ i = 1, 2, \ldots, n\} \end{aligned}$$

Remark

- 记: $A_1 \times A_2 \times \cdots \times A_n \triangleq \prod_{i=1}^{n} A_i$;　记: $\prod_{i=1}^{n} A \triangleq A^n$;
- 无交换率: 设$A \neq B$, 则: $A \times B \neq B \times A$, 如:
  $\{\langle 1,2\rangle\} = \{1\} \times \{2\} \neq \{2\} \times \{1\} = \{\langle 2,1\rangle\}$
- 无结合律: $A \times B \times C \triangleq (A \times B) \times C \neq A \times (B \times C)$;
- 但是上述两乘积集合是等价的; 即存在一个映射将两个集合的元素一一对应起来.

乘积集合

Definition (叉积 笛卡儿乘积集合 Cartesian Product)

$$A \times B \triangleq \{\langle a, b\rangle \mid a \in A \wedge b \in B\}$$

$$\begin{aligned} A_1 \times A_2 \times \cdots \times A_n &\triangleq (A_1 \times A_2 \times \cdots \times A_{n-1}) \times A_n \\ &= \{\langle a_1, a_2, \ldots, a_n\rangle \mid a_i \in A_i,\ i = 1, 2, \ldots, n\} \end{aligned}$$

Remark

- 记: $A_1 \times A_2 \times \cdots \times A_n \triangleq \prod_{i=1}^{n} A_i$; 记: $\prod_{i=1}^{n} A \triangleq A^n$;
- 无交换率: 设$A \neq B$, 则: $A \times B \neq B \times A$, 如:
  $\{\langle 1,2\rangle\} = \{1\} \times \{2\} \neq \{2\} \times \{1\} = \{\langle 2,1\rangle\}$
- 无结合律: $A \times B \times C \triangleq (A \times B) \times C \neq A \times (B \times C)$;
- 但是上述两乘积集合是等价的; 即存在一个映射将两个集合的元素一一对应起来.

乘积集合

Definition (叉积 笛卡儿乘积集合 Cartesian Product)

$$A \times B \triangleq \{\langle a, b\rangle \mid a \in A \wedge b \in B\}$$

$$\begin{aligned} A_1 \times A_2 \times \cdots \times A_n &\triangleq (A_1 \times A_2 \times \cdots \times A_{n-1}) \times A_n \\ &= \{\langle a_1, a_2, \ldots, a_n\rangle \mid a_i \in A_i,\, i = 1, 2, \ldots, n\} \end{aligned}$$

Remark

- 记: $A_1 \times A_2 \times \cdots \times A_n \triangleq \prod_{i=1}^{n} A_i$; 记: $\prod_{i=1}^{n} A \triangleq A^n$;
- 无交换率: 设$A \neq B$, 则: $A \times B \neq B \times A$, 如:
  $\{\langle 1, 2\rangle\} = \{1\} \times \{2\} \neq \{2\} \times \{1\} = \{\langle 2, 1\rangle\}$
- 无结合律: $A \times B \times C \triangleq (A \times B) \times C \neq A \times (B \times C)$;
- 但是上述两乘积集合是等价的; 即存在一个映射将两个集合的元素一一对应起来.

乘积集合

Definition (叉积 笛卡儿乘积集合 Cartesian Product)

$$A \times B \triangleq \{\langle a, b\rangle \mid a \in A \wedge b \in B\}$$

$$\begin{aligned} A_1 \times A_2 \times \cdots \times A_n &\triangleq (A_1 \times A_2 \times \cdots \times A_{n-1}) \times A_n \\ &= \{\langle a_1, a_2, \ldots, a_n\rangle \mid a_i \in A_i,\ i = 1, 2, \ldots, n\} \end{aligned}$$

Remark

- 记: $A_1 \times A_2 \times \cdots \times A_n \triangleq \prod_{i=1}^{n} A_i$; 记: $\prod_{i=1}^{n} A \triangleq A^n$;
- 无交换率: 设$A \neq B$, 则: $A \times B \neq B \times A$, 如:
  $\{\langle 1, 2\rangle\} = \{1\} \times \{2\} \neq \{2\} \times \{1\} = \{\langle 2, 1\rangle\}$
- 无结合律: $A \times B \times C \triangleq (A \times B) \times C \neq A \times (B \times C)$;
- 但是上述两乘积集合是等价的; 即存在一个映射将两个集合的元素一一对应起来.

乘积集合

Definition (叉积 笛卡儿乘积集合 Cartesian Product)

$$A \times B \triangleq \{\langle a, b\rangle \mid a \in A \wedge b \in B\}$$

$$A_1 \times A_2 \times \cdots \times A_n \triangleq (A_1 \times A_2 \times \cdots \times A_{n-1}) \times A_n$$
$$= \{\langle a_1, a_2, \ldots, a_n\rangle \mid a_i \in A_i,\ i = 1, 2, \ldots, n\}$$

Remark

- *记:* $A_1 \times A_2 \times \cdots \times A_n \triangleq \prod_{i=1}^{n} A_i$; *记:* $\prod_{i=1}^{n} A \triangleq A^n$;
- *无交换率: 设*$A \neq B$, *则:* $A \times B \neq B \times A$, *如:*
  $\{\langle 1, 2\rangle\} = \{1\} \times \{2\} \neq \{2\} \times \{1\} = \{\langle 2, 1\rangle\}$
- *无结合律:* $A \times B \times C \triangleq (A \times B) \times C \neq A \times (B \times C)$;
- *但是上述两乘积集合是等价的; 即存在一个映射将两个集合的元素一一对应起来.*

Example

Example

- n维向量空间: $\mathbb{R}^n$;
- n个原子$P_1, P_2, \ldots, P_n$的所有指派的集合是: $\{0, 1\}^n$;
- n元谓词公式$P(x_1, x_2, \ldots, x_n)$的解释I是D^n到$\{0, 1\}$的函数;
- 元素类型为T的长度为n的数组: T^n;
- 函数式程序设计语言或Python的元组(tuple), Ex:
  `t = ("zhangsan", 1992, [1,2,3])`
- C语言的struct类型本质上也是乘积集合，只不过n重组的位置序号在struct中用分量名表示，为每个不同位置的分量加上了标签，称之为Named Cartesian Product, 如：`struct {int x ; char * s}`对应的乘积集合是:
  $$INT \times POINTER(CHAR).$$

Example

Example

- n维向量空间：$\mathbb{R}^n$;
- n个原子$P_1, P_2, \ldots, P_n$的所有指派的集合是：$\{0,1\}^n$;
- n元谓词公式$P(x_1, x_2, \ldots, x_n)$的解释I是D^n到$\{0,1\}$的函数;
- 元素类型为T的长度为n的数组：T^n;
- 函数式程序设计语言或Python的元组(tuple), Ex:
  `t = ("zhangsan", 1992, [1,2,3])`
- C语言的struct类型本质上也是乘积集合，只不过n重组的位置序号在struct中用分量名表示，为每个不同位置的分量加上了标签，称之为Named Cartesian Product, 如：`struct {int x ; char * s}`对应的乘积集合是：

  $$INT \times POINTER(CHAR).$$

Example

Example

- n维向量空间：$\mathbb{R}^n$；
- n个原子$P_1, P_2, \ldots, P_n$的所有指派的集合是：$\{0, 1\}^n$；
- n元谓词公式$P(x_1, x_2, \ldots, x_n)$的解释I是$\mathcal{D}^n$到$\{0, 1\}$的函数；
- 元素类型为T的长度为n的数组：T^n；
- 函数式程序设计语言或Python的元组(tuple)，Ex:
  `t = ("zhangsan", 1992, [1,2,3])`
- C语言的struct类型本质上也是乘积集合，只不过n重组的位置序号在struct中用分量名表示，为每个不同位置的分量加上了标签，称之为Named Cartesian Product，如：`struct {int x ; char * s}`对应的乘积集合是：

  $$INT \times POINTER(CHAR).$$

Example

Example

- n维向量空间：$\mathbb{R}^n$；
- n个原子$P_1, P_2, \ldots, P_n$的所有指派的集合是：$\{0,1\}^n$；
- n元谓词公式$P(x_1, x_2, \ldots, x_n)$的解释I是$\mathcal{D}^n$到$\{0,1\}$的函数；
- 元素类型为T的长度为n的数组：T^n；
- 函数式程序设计语言或Python的元组(tuple), Ex:
  `t = ("zhangsan", 1992, [1,2,3])`
- C语言的struct类型本质上也是乘积集合，只不过n重组的位置序号在struct中用分量名表示，为每个不同位置的分量加上了标签，称之为Named Cartesian Product, 如：`struct {int x ; char * s}`对应的乘积集合是：

  $INT \times POINTER(CHAR)$.

Example

Example

- n维向量空间：$\mathbb{R}^n$;
- n个原子$P_1, P_2, \ldots, P_n$的所有指派的集合是：$\{0, 1\}^n$;
- n元谓词公式$P(x_1, x_2, \ldots, x_n)$的解释I是$\mathcal{D}^n$到$\{0, 1\}$的函数;
- 元素类型为T的长度为n的数组：T^n;
- 函数式程序设计语言或Python的元组(tuple)，Ex:
  `t = ("zhangsan", 1992, [1,2,3])`
- C语言的struct类型本质上也是乘积集合，只不过n重组的位置序号在struct中用分量名表示，为每个不同位置的分量加上了标签，称之为Named Cartesian Product，如：`struct {int x ; char * s}`对应的乘积集合是：

  $INT \times POINTER(CHAR)$.

Example

Example

- n维向量空间：$\mathbb{R}^n$;
- n个原子$P_1, P_2, \ldots, P_n$的所有指派的集合是：$\{0, 1\}^n$;
- n元谓词公式$P(x_1, x_2, \ldots, x_n)$的解释I是$\mathcal{D}^n$到$\{0, 1\}$的函数;
- 元素类型为T的长度为n的数组：T^n;
- 函数式程序设计语言或Python的元组(tuple), Ex:
  `t = ("zhangsan", 1992, [1,2,3])`
- C语言的struct类型本质上也是乘积集合，只不过n重组的位置序号在struct中用分量名表示，为每个不同位置的分量加上了标签，称之为Named Cartesian Product，如：`struct {int x ; char * s}`对应的乘积集合是：

  $$INT \times POINTER(CHAR).$$

Example

Example

- n维向量空间: $\mathbb{R}^n$;
- n个原子$P_1, P_2, \ldots, P_n$的所有指派的集合是: $\{0, 1\}^n$;
- n元谓词公式$P(x_1, x_2, \ldots, x_n)$的解释I是$\mathcal{D}^n$到$\{0, 1\}$的函数;
- 元素类型为T的长度为n的数组: T^n;
- 函数式程序设计语言或Python的元组(tuple), Ex:
  `t = ("zhangsan", 1992, [1,2,3])`
- C语言的`struct`类型本质上也是乘积集合，只不过n重组的位置序号在`struct`中用分量名表示，为每个不同位置的分量加上了标签，称之为Named Cartesian Product，如: `struct {int x ; char * s}`对应的乘积集合是:
  $$INT \times POINTER(CHAR).$$

相关性质

Theorem

- $\emptyset \times A = \emptyset$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- $if\ A_i(i = 1, 2 \ldots, n)$是有限集合，则:

$$\left|\prod_{i=1}^{n} A_i\right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0, 1\}^n| = 2^n$$

相关性质

Theorem

- $\varnothing \times A = \varnothing$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- *if* $A_i (i = 1, 2 \ldots, n)$ 是有限集合，则:

$$\left| \prod_{i=1}^{n} A_i \right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0, 1\}^n| = 2^n$$

相关性质

Theorem

- $\varnothing \times A = \varnothing$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- $if\ A_i(i = 1, 2 \ldots, n)$是有限集合，则:

$$\left| \prod_{i=1}^{n} A_i \right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0, 1\}^n| = 2^n$$

相关性质

Theorem

- $\varnothing \times A = \varnothing$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- *if* $A_i (i = 1, 2 \ldots, n)$是有限集合，则:

$$\left| \prod_{i=1}^{n} A_i \right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0, 1\}^n| = 2^n$$

相关性质

Theorem

- $\varnothing \times A = \varnothing$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- if $A_i (i = 1, 2 \ldots, n)$是有限集合，则：

$$\left| \prod_{i=1}^{n} A_i \right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0, 1\}^n| = 2^n$$

相关性质

Theorem

- $\varnothing \times A = \varnothing$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- if $A_i(i = 1, 2 \ldots, n)$是有限集合，则:

$$\left|\prod_{i=1}^{n} A_i\right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0,1\}^n| = 2^n$$

相关性质

Theorem

- $\emptyset \times A = \emptyset$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- *if* $A_i (i = 1, 2 \ldots, n)$是有限集合，则：

$$\left| \prod_{i=1}^{n} A_i \right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0, 1\}^n| = 2^n$$

相关性质

Theorem

- $\emptyset \times A = \emptyset$;
- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;
- $(A \cup B) \times C = (A \times C) \cup (B \times C)$;
- $(A \cap B) \times C = (A \times C) \cap (B \times C)$;
- *if* $A_i (i = 1, 2 \ldots, n)$*是有限集合，则:*

$$\left| \prod_{i=1}^{n} A_i \right| = \prod_{i=1}^{n} |A_i|$$

Example

$$|\{0, 1\}^n| = 2^n$$

容斥原理

Description (容斥原理, inclusion-exclusion Principle)

设下述讨论的集合均为有限集合.

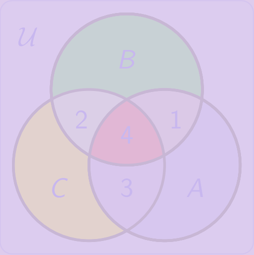
- $|A \cap B| \leqslant \min(|A|, |B|)$;
- $|A \cup B| = |A| + |B| - |A \cap B|$;
- $|A \cup B \cup C| =$
  $|A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C|$;
- ......
- $\left|\bigcup_{i=1}^{n} A_i\right| = \sum_{i=1}^{n} |A_i| - \sum_{i,j:i\neq j} |A_i \cap A_j| + \sum_{i,j,k:i\neq j\neq k\neq i} |A_i \cap A_j \cap A_k| - \cdots \pm |A_1 \cap A_2 \cdots \cap A_n|$

容斥原理的直观解释

Description

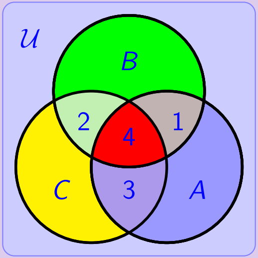
容斥原理的直观解释

Description



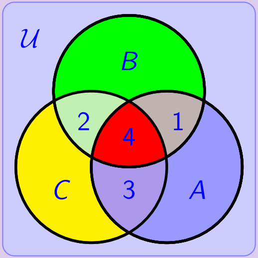
容斥原理的直观解释

Description



容斥原理的直观解释

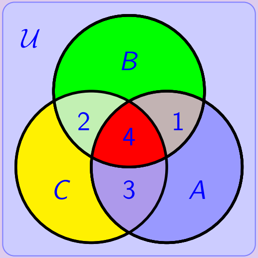
Description



- $|A|+|B|+|C|$ 中①, ②和③ 计数2次，而④计数3次，∴
- $|A\cup B\cup C| = |A|+|B|+|C| - |①| - |②| - |③| - 2|④|$ (i)
- $|A\cap B|+|B\cap C|+|A\cap C|$ 中④计数3次，∴
- $|A\cap B|+|B\cap C|+|A\cap C| = |①| + |②| + |③| + 3|④|$ (ii)
- 由(i)和(ii)得: $|A\cup B\cup C| = |A|+|B|+|C| - |A\cap B| - |B\cap C| - |A\cap C| + |④|$

容斥原理的直观解释

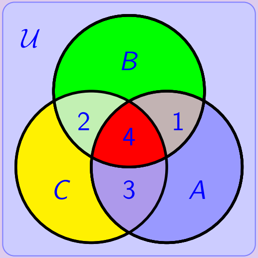
Description



- $|A|+|B|+|C|$ 中①，②和③ 计数2次，而④计数3次，$\therefore$
- $|A\cup B\cup C| = |A|+|B|+|C| - |①|-|②|-|③|-2|④| \quad (i)$
- $|A\cap B|+|B\cap C|+|A\cap C|$ 中④计数3次，$\therefore$
- $|A\cap B|+|B\cap C|+|A\cap C| = |①|+|②|+|③|+3|④| \quad (ii)$
- 由(i)和(ii)得：$|A\cup B\cup C| = |A|+|B|+|C| - |A\cap B|-|B\cap C|-|A\cap C|+|④|$

容斥原理的直观解释

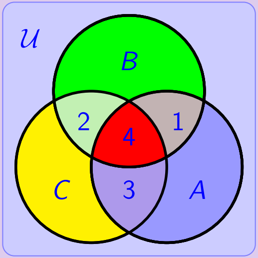
Description



- $|A|+|B|+|C|$ 中①, ②和③ 计数2次，而④计数3次，$\therefore$
- $|A\cup B\cup C| = |A|+|B|+|C|-|①|-|②|-|③|-2|④|$ $\quad (i)$
- $|A\cap B|+|B\cap C|+|A\cap C|$ 中④计数3次，$\therefore$
- $|A\cap B|+|B\cap C|+|A\cap C| = |①|+|②|+|③|+3|④|$ $\quad (ii)$
- 由(i)和(ii)得：
  $|A\cup B\cup C| = |A|+|B|+|C|-|A\cap B|-|B\cap C|-|A\cap C|+|④|$

容斥原理的直观解释

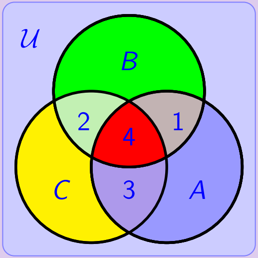
Description



- $|A|+|B|+|C|$中①, ②和③ 计数2次，而④计数3次，$\therefore$
- $|A\cup B\cup C| = |A|+|B|+|C|-|①|-|②|-|③|-2|④| \qquad (i)$
- $|A\cap B|+|B\cap C|+|A\cap C|$中④计数3次，$\therefore$
- $|A\cap B|+|B\cap C|+|A\cap C| = |①|+|②|+|③|+3|④| \qquad (ii)$
- 由(i)和(ii)得：$|A\cup B\cup C| = |A|+|B|+|C|-|A\cap B|-|B\cap C|-|A\cap C|+|④|$

容斥原理的直观解释

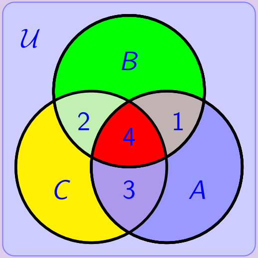
Description



- $|A|+|B|+|C|$中①，②和③ 计数2次，而④计数3次，∴
- $|A\cup B\cup C| = |A|+|B|+|C| - |①| - |②| - |③| - 2|④|$ *(i)*
- $|A\cap B|+|B\cap C|+|A\cap C|$中④计数3次，∴
- $|A\cap B|+|B\cap C|+|A\cap C| = |①|+|②|+|③|+3|④|$ *(ii)*
- 由(i)和(ii)得：$|A\cup B\cup C| = |A|+|B|+|C| - |A\cap B| - |B\cap C| - |A\cap C| + |④|$

容斥原理的直观解释

Description



- $|A| + |B| + |C|$ 中①, ②和③ 计数2次，而④计数3次, $\therefore$
- $|A \cup B \cup C| = |A| + |B| + |C| - |①| - |②| - |③| - 2|④|$ (i)
- $|A \cap B| + |B \cap C| + |A \cap C|$ 中④计数3次, $\therefore$
- $|A \cap B| + |B \cap C| + |A \cap C| = |①| + |②| + |③| + 3|④|$ (ii)
- 由(i)和(ii)得: $|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |④|$

Example

Problem

某教研组有教师30人($|T|=30$)，其中会英语的15人($|E|=15$)，其中会法语的8人($|F|=8$)，其中会德语的6人($|G|=6$)，三门均会的有3人($|E\cap F\cap G|=3$)，问最少有多少人三门外语均不会(N)?

Solution

Example

Problem

某教研组有教师30人($|T|=30$)，其中会英语的15人($|E|=15$)，其中会法语的8人($|F|=8$)，其中会德语的6人($|G|=6$)，三门均会的有3人($|E\cap F\cap G|=3$)，问最少有多少人三门外语均**不会**(N)?

Solution

1. $N=T-(E\cup F\cup G)$，求$|N|$的最小值等价于求$|E\cup F\cup G|$的最大值；
2. $$\begin{aligned}&|E\cup F\cup G|\\&=|E|+|F|+|G|-|E\cap F|-|F\cap G|-|E\cap G|+|E\cap F\cap G|\\&=32-(|E\cap F|+|F\cap G|+|E\cap G|)\end{aligned}$$
3. 而，$E\cap F\supseteq E\cap F\cap G$，$\therefore |E\cap F|\geqslant|E\cap F\cap G|=3$
4. $\therefore |E\cup F\cup G|\leqslant 32-3\times 3=23$
5. $|N|=|T|-|E\cup F\cup G|\geqslant 30-23=7$

Example

Problem

某教研组有教师30人($|T| = 30$)，其中会英语的15人($|E| = 15$)，其中会法语的8人($|F| = 8$)，其中会德语的6人($|G| = 6$)，三门均会的有3人($|E \cap F \cap G| = 3$)，问最少有多少人三门外语均不会(N)?

Solution

1. $N = T - (E \cup F \cup G)$，求$|N|$的最小值等价于求$|E \cup F \cup G|$的最大值；
2. $$\begin{aligned} & |E \cup F \cup G| \\ = & |E| + |F| + |G| - |E \cap F| - |F \cap G| - |E \cap G| + |E \cap F \cap G| \\ = & 32 - (|E \cap F| + |F \cap G| + |E \cap G|) \end{aligned}$$
3. 而，$E \cap F \supseteq E \cap F \cap G$, $\therefore |E \cap F| \geqslant |E \cap F \cap G| = 3$
4. $\therefore |E \cup F \cup G| \leqslant 32 - 3 \times 3 = 23$
5. $|N| = |T| - |E \cup F \cup G| \geqslant 30 - 23 = 7$

Example

Problem

某教研组有教师30人($|T| = 30$)，其中会英语的15人($|E| = 15$)，其中会法语的8人($|F| = 8$)，其中会德语的6人($|G| = 6$)，三门均会的有3人($|E \cap F \cap G| = 3$)，问最少有多少人三门外语均不会(N)?

Solution

1. $N = T - (E \cup F \cup G)$，求$|N|$的最小值等价于求$|E \cup F \cup G|$的最大值；
2. $$\begin{aligned} & |E \cup F \cup G| \\ = & |E| + |F| + |G| - |E \cap F| - |F \cap G| - |E \cap G| + |E \cap F \cap G| \\ = & 32 - (|E \cap F| + |F \cap G| + |E \cap G|) \end{aligned}$$
3. 而，$E \cap F \supseteq E \cap F \cap G$，$\therefore |E \cap F| \geqslant |E \cap F \cap G| = 3$
4. $\therefore |E \cup F \cup G| \leqslant 32 - 3 \times 3 = 23$
5. $|N| = |T| - |E \cup F \cup G| \geqslant 30 - 23 = 7$

Example

Problem

某教研组有教师30人($|T| = 30$)，其中会英语的15人($|E| = 15$)，其中会法语的8人($|F| = 8$)，其中会德语的6人($|G| = 6$)，三门均会的有3人($|E \cap F \cap G| = 3$)，问最少有多少人三门外语均不会(N)?

Solution

1. $N = T - (E \cup F \cup G)$，求$|N|$的最小值等价于求$|E \cup F \cup G|$的最大值；
2. $$\begin{aligned} & |E \cup F \cup G| \\ = & |E| + |F| + |G| - |E \cap F| - |F \cap G| - |E \cap G| + |E \cap F \cap G| \\ = & 32 - (|E \cap F| + |F \cap G| + |E \cap G|) \end{aligned}$$
3. 而，$E \cap F \supseteq E \cap F \cap G$，$\therefore |E \cap F| \geqslant |E \cap F \cap G| = 3$
4. $\therefore |E \cup F \cup G| \leqslant 32 - 3 \times 3 = 23$
5. $|N| = |T| - |E \cup F \cup G| \geqslant 30 - 23 = 7$

Example

Problem

某教研组有教师30人($|T| = 30$)，其中会英语的15人($|E| = 15$)，其中会法语的8人($|F| = 8$)，其中会德语的6人($|G| = 6$)，三门均会的有3人($|E \cap F \cap G| = 3$)，问最少有多少人三门外语均不会(N)?

Solution

1. $N = T - (E \cup F \cup G)$，求$|N|$的最小值等价于求$|E \cup F \cup G|$的最大值；
2. $$\begin{aligned} & |E \cup F \cup G| \\ = & |E| + |F| + |G| - |E \cap F| - |F \cap G| - |E \cap G| + |E \cap F \cap G| \\ = & 32 - (|E \cap F| + |F \cap G| + |E \cap G|) \end{aligned}$$
3. 而，$E \cap F \supseteq E \cap F \cap G$, $\therefore |E \cap F| \geqslant |E \cap F \cap G| = 3$
4. $\therefore |E \cup F \cup G| \leqslant 32 - 3 \times 3 = 23$
5. $|N| = |T| - |E \cup F \cup G| \geqslant 30 - 23 = 7$

Example

Problem

某教研组有教师30人($|T| = 30$)，其中会英语的15人($|E| = 15$)，其中会法语的8人($|F| = 8$)，其中会德语的6人($|G| = 6$)，三门均会的有3人($|E \cap F \cap G| = 3$)，问最少有多少人三门外语均不会(N)?

Solution

1. $N = T - (E \cup F \cup G)$，求$|N|$的最小值等价于求$|E \cup F \cup G|$的最大值；
2. $$\begin{aligned} & |E \cup F \cup G| \\ = & |E| + |F| + |G| - |E \cap F| - |F \cap G| - |E \cap G| + |E \cap F \cap G| \\ = & 32 - (|E \cap F| + |F \cap G| + |E \cap G|) \end{aligned}$$
3. 而，$E \cap F \supseteq E \cap F \cap G$，$\therefore |E \cap F| \geqslant |E \cap F \cap G| = 3$
4. $\therefore |E \cup F \cup G| \leqslant 32 - 3 \times 3 = 23$
5. $|N| = |T| - |E \cup F \cup G| \geqslant 30 - 23 = 7$

本章小节

1. 基本概念
   - 精确的集合语言
   - 什么是集合
   - 集合的包含关系
2. 集合上的运算
   - 运算的定义
   - 集合运算的恒等式和不等式
   - 补运算
   - 文氏图和范式的关系
   - 集合运算的扩充
   - 无限交和无限并
   - 集合的程序实现
3. 集合的构造
   - 幂集合
   - 自然数 — 集合的递归构造
   - 乘积集合
   - 字符串集合
   - 容斥原理