Лекція 5
План лекції
Відношення порядку

1. Відношення порядку.
1.1. Приклади відношень порядку.
1.2 Визначення відношень порядку.
1.3. Термінологія й позначення.
1.4. Види відношень порядку.
1.5. Основні поняття про впорядковані множини.
1.6. Лінійно впорядковані множини.
1.7. Властивості лінійно впорядкованих множин.
1.8. Цілком упорядкована множина.
1.9. Частково впорядкована множина.
1.10. Розбивка частково впорядкованої множини на ланцюзі.
1.11. Визначення найбільшого елемента множини.
1.12. Визначення максимального елемента множини.
1.13. Визначення найменшого й мінімального елементів множини.
1.14. Визначення верхньої й нижньої граней множини.
1.15. Визначення точної верхньої грані множини.
1.16. Визначення точної нижньої грані множини.
1.17. Діаграми Хассе.

2. Відношення порядку

Існують відношення, що визначають порядок розташування елементів множини.

1. Умову відношення

«t_i раніше t_j» і «t_j пізніше t_i»

використовуємо у випадках, коли елементами множини є стани динамічної системи

$$T = \{t_0, t_1, t_2, t_3, ..., t_{n-1}\},$$

де $t_0 < t_1 < t_2 < t_3 < ... < t_{n-1}$.

Символи «<» «>» використовуються для порівняння величин відрізків часу, вимірюваних від початку відліку.

Запишемо ці відношення у вигляді предиката:

$$R_1 = \left\{ \left(t_i, t_j \right) \middle| t_i < t_j \text{ } при \text{ } i < j \right\}$$

$$R_2 = \left\{ \left(t_j, t_i \right) \middle| t_j > t_i \text{ } при \text{ } j > i \right\}$$

2. Умову відношення

«a_j більше a_i» або «a_i менше a_j»

використовуємо, коли елементами множини є числа або об'єкти, що мають властивість, виражену числом.

$$A = \{a_0, a_1, a_2, ..., a_i, ..., a_j, ..., a_n\},$$

де $a_0 < a_1 < a_2 < ... < a_i < ... < a_j < ... < a_n$.
Символами «>» або «<» користуються для порівняння чисел.

Відношення R у вигляді предиката, задане на $A \times A$.

$$R_1 = \left\{ \left(a_i, a_j \right) \middle| a_i < a_j \ \text{при } i < j \right\}$$
$$R_2 = \left\{ \left(a_j, a_i \right) \middle| a_j > a_i \ \text{при } j > i \right\}$$

3. Умова відношення

«A_i *входить в* A_j», «A_i *строго входить в* A_j»

використовуємо, коли елементами множини є множини.

$$A = \left\{ A_0, A_1, ..., A_i, ..., A_j, ..., A_n \right\}.$$

де $A_0 \subseteq A_1 \subseteq ... \subseteq A_i \subseteq ... \subseteq A_j \subseteq ... \subseteq A_n$ або

$A_0 \subset A_1 \subset ... \subset A_i \subset ... \subset A_j \subset ... \subset A_n$.

Відношення R у вигляді предиката, задане на $A \times A$.

$$R_1 = \left\{ \left(A_i, A_j \right) \middle| A_i \subseteq A_j \ \text{при } i < j \right\}$$
$$R_2 = \left\{ \left(A_i, A_j \right) \middle| A_i \subset A_j \ \text{при } i < j \right\}$$

У всіх випадках можна розташувати елементи множин у деякому порядку або, інакше кажучи, ввести відношення порядку на множині.

Визначення відношень порядку
Відношення порядку на множині *A* поділяють на:
- відношення строгого порядку;
- відношення нестрогого порядку.

Визначення. Відношення R називають **відношенням строгого порядку** на множині A, якщо воно має властивості:
- *антирефлексивності*, тобто якщо xRy то $x \neq y$.
- *антисиметричності,* тобто, якщо xRy і yRx, то $x = y$.
- *транзитивності,* тобто, якщо xRy і yRz, то xRz.

Визначення. Відношення R називають **відношенням нестрогого порядку** на множині A, якщо воно має властивості:
- *рефлексивності*, тобто, xRx .
- *антисиметричності,* тобто, якщо xRy і yRx, то $x = y$
- *транзитивності,* тобто, якщо xRy і yRz, то xRz.

Термінологія й позначення

1.**Відношення нестрогого порядку** позначають символом «$\leq$ » за аналогією з відношенням «менше або дорівнює» на множині дійсних чисел. При цьому, якщо $a \leq b$, то говорять, що елемент a не перевищує b або a підпорядкований b.

2. **Відношення строгого порядку**. Якщо $a \leq b$ і $a \neq b$, то пишуть $a < b$ і говорять, що a менше b або, що елемент a строго підпорядкований b.

3. Загальний випадок відношень. Іноді, щоб відрізнити відношення порядку на деякій множині від відношення порядку «$\leq$ » і «<» на множині дійсних чисел, використовують спеціальні символи «$\prec$» і «$\prec$ ».

Приклад. Відношення порядку в Rn

1. Відношення чисел «$\leq$ » «$\geq$ » задають нестрогий порядок.

$(a_1,\ldots,a_{i-1},a_i,a_{i+1},\ldots,a_n) \leq (b_1,\ldots,b_{i-1},b_i,b_{i+1},\ldots,b_n)$ якщо

$a_1 \leq b_1,\ldots, a_{i-1} \leq b_{i-1}, a_i \leq b_i , a_{i+1} \leq b_{i+1},\ldots, a_n \leq b_n$

2. Відношення чисел «< » «> » задають строгий порядок

$(a_1,\ldots,a_{i-1},a_i,a_{i+1},\ldots,a_n) < (b_1,\ldots,b_{i-1},b_i,b_{i+1},\ldots,b_n)$, якщо

$a_1 < b_1,\ldots,a_{i-1} < b_{i-1},a_i < b_i ,a_{i+1} < b_{i+1},\ldots,a_n < b_n$

Однак для встановлення нестрогого порядку достатньо,
щоб умова $a_i < b_i$ була виконана хоча б по одній координаті, тобто

$(a_1,\ldots,a_{i-1},a_i,a_{i+1},\ldots,a_n) < (b_1,\ldots,b_{i-1},b_i,b_{i+1},\ldots,b_n)$, якщо

$a_1 \leq b_1,\ldots, a_{i-1} \leq b_{i-1}, a_i < b_i , a_{i+1} \leq b_{i+1},\ldots, a_n \leq b_n$

Приклад. Відношення порядку в R^2

$(a_1,a_2) \leq (b_1,b_2)$

Це відношення справедливе, якщо $a_1 \leq b_1$ **і** $a_2 \leq b_2$

$(1,5) \leq (1,7) \rightarrow 1 = 1$ і $5 < 7$; $(1,5)$ **і** $(5,1)$ **непорівнювані**

$(a_1,a_2) < (b_1,b_2)$

$(1,5) < (2,7) \rightarrow (1 < 2)$ **і** $5 < 7$ **;** $(1,5)$ **і** $(7,2)$ **непорівнювані**

Приклад. Відношення порядку в R^3

$$(a_1,a_2,a_3) \leq (b_1,b_2,b_3)$$

Це відношення справедливе, якщо $a_1 \leq b_1$, $a_2 \leq b_2$ і $a_3 \leq b_3$

$(1,2,3) \leq (1,2,4) \rightarrow 1 = 1, 2 = 2, 3 < 4$

$$(a_1, a_2, a_3) < (b_1, b_2, b_3)$$

Це відношення справедливе, якщо $a_1 < b_1$, $a_2 < b_2$ і $a_3 < b_3$

$(1,2,3) < (2,3,4) \rightarrow 1 < 2, 2 < 3, 3 < 4$

Види відношень порядку:

1. Строгий повний порядок
Відношення строгого порядку задане **на всіх** елементах упорядкованої множини.

2. Строгий частковий порядок
Відношення строгого порядку задане **не на всіх** елементах упорядкованої множини.

3. Нестрогий повний порядок
Відношення нестрогого порядку задане **на всіх** елементах упорядкованої множини.

4. Нестрогий частковий порядок
Відношення нестрогого порядку задане **не на всіх** елементах упорядкованої множини.

Основні поняття про впорядковані множини

Упорядковані множини утворюють один з фундаментальних типів математичних структур.

Визначення. *Упорядкованою множиною* називають непусту множину X разом із заданим на ньому бінарним нестрогим «$\le$» *відношенням порядку,* яке за визначенням: 1) рефлексивне: $a \le a$;
2) антисиметричне: $a \le b \le a \Rightarrow a = b$(для будь-яких a, b, X).
3) транзитивне: $a \le b \le c \Rightarrow a \le c$;
або строгим «$<$» *відношенням порядку,* яке за визначенням:
1) антирефлексивне: $a < b \Rightarrow a \ne b$;
2) антисиметричне: $a < b \wedge b < a \Rightarrow a = b$
3) транзитивне: $a \le b \le c \Rightarrow a \le c$;

Лінійно впорядковані множини

Визначення порівнюваності. Елементи a і b упорядкованої множини називають *порівнюваними,* якщо $a < b$, $a = b$ або $a > b$. Знаки $<$, $=$ і $>$ мають звичайний зміст.

Визначення (через порівнюваність). Упорядкована множина X називається *лінійно впорядкованою,* або *ланцюгом,* якщо будь-які два його елементи порівнювані.

Визначення (через відношення лінійного порядку).

Упорядковану множину X називають *лінійно впорядкованою,* або *ланцюгом,* якщо на ній задане відношення лінійного порядку.

Визначення. Бінарне відношення *R* на множині X називають *відношенням лінійного порядку,* якщо для будь-яких $a \in X$ і $b \in X$ (для будь-яких двох елементів множини X) або aRb, або bRa.

Лінійно впорядковані множини

(продовження)

Іншими словами, відношення порядку R на множині X називають лінійним, якщо будь-які два елементи цієї множини перебувають у відношенні R.

Приклад ланцюга. Множина всіх дійсних чисел зі звичайним порядком – це ланцюг.

Відмітимо, що *антиподами ланцюгів* є *антиланцюги.*

Визначення.
Антиланцюг – упорядкована множина, у якій жодні два різні елементи не є порівнюваними.

Властивості лінійно впорядкованих множин

1. **Покриття.**
Нехай X – довільний ланцюг. Якщо $a < b$ в X і не існує елемента $c \in X$ з умовою $a < c < b$ (який розміщений між a і b), то співвідношення $a < b$ називають *покриттям.*

2. **Взаємне розташування елементів**
Елемент a *називають попереднім* для b
Елемент b називають *наступним* за a.
Елемент ланцюга, у якого немає попереднього елемента, називають *граничним елементом.*

3. **Щільний ланцюг**
Ланцюг називають *щільним,* якщо в ньому немає покриттів. У щільних ланцюгах між будь-якими елементами $a < b$ лежить нескінченна кількість елементів.

4. **Щільний підланцюг**
Підланцюг A ланцюга X називають *щільним в X,* якщо між будь-якими елементами $a < b$ з X обов'язково найдеться елемент з A.

5. **Повний зверху ланцюг**
Ланцюг називають *повним зверху,* якщо його довільна непуста підмножина має sup (супремум).

6. **Повний знизу ланцюг**
Ланцюг називають *повним знизу,* якщо його довільна непуста підмножина має inf (інфімум).

7. **Повний ланцюг**
Ланцюг називають *повним,* якщо він повний зверху і знизу одночасно.

Цілком упорядкована множина

Найважливіший клас ланцюгів утворюють цілком упорядковані множини.

Визначення. Ланцюг називають *цілком упорядкованою множиною,* якщо будь-яка її непуста підмножина має найменший елемент.

Приклад. Ланцюг усіх натуральних чисел N і скінченні ланцюги є прикладами цілком упорядкованих множин.

Приклад. Будь-яка непуста підмножина цілком упорядкованої множини цілком упорядкована.

Частково впорядкована множина

Визначення (через відношення часткового порядку).
Упорядковану множину X називають *частково впорядкованою,* якщо на ній задане відношення часткового порядку.

Визначення. Бінарне відношення R на множині X називають відношенням ***часткового порядку****,* якщо для деяких $a \in X$, $b \in X$ не виконується відношення ні aRb, ні відношення bRa.

Якщо відношення R на X є відношення *нестрогого часткового порядку*, то воно

рефлексивне - $\forall a\left(aRa\right)$,

антисиметричне - $\forall a,b\left(aRb\right)\wedge\left(bRa\right)\Rightarrow a=b$

транзитивне - $\forall a,b,c\left(aRb\right)\wedge\left(bRc\right)\Rightarrow aRc$.

Якщо відношення R на X є відношенням *строгого часткового порядку*, то воно

антирефлексивне - $\forall a,b\left(aRb\right)\Rightarrow a\neq b$, антисиметричне - $\forall a,b\big(aRb\big)\wedge\big(bRa\big)\Rightarrow a=b$

транзитивне - $\forall a,b,c\big(aRb\big)\wedge\big(bRc\big)\Rightarrow aRc$.

Приклад частково впорядкованої множини

Приклад 1. Нехай задане:

1. Множина T додатних дільників числа 30.
$$T=\{1,2,3,5,6,10,15,30\}.$$
2. Відношення «≤ », згідно якого *m* і *n є порівнюваними*: *m≤n за умови, що m* ділить *n* націло.

Нехай *n*=15 і *m*=5. Тоді *n* і *m* – є порівнюваними, оскільки 5 ділить 15 націло.
Нехай *n*=6 і *m*=5. Тоді *n* і *m* – непорівнянні, оскільки 5 не ділить 6 націло.

Висновок.

1. Задане відношення порядку «≤ » на множині T є відношенням часткового порядку.
2. Множина T є *частково упорядкованою* на заданому відношенні.

Розбиття частково впорядкованої множини на ланцюзі

Нехай є деяка множина A. Говорять, що множина A *розбита* на підмножини $A_1, A_2, A_3, \ldots A_m$, якщо:

1. $A_i \neq \varnothing,\ \ (i=1,2,...,m)$;
2. $A_i \cap A_j = \varnothing$, якщо $i \neq j$ для всіх $i,j \in \{1,2,3,...,m\}$;
3. $A=\bigcup\limits_{i=1}^{m} A_i$

Нехай A є частково впорядкованою множиною. Розбиття множини A на ланцюзі називають *найменшим*, якщо воно має найменше число елементів m у порівнянні з іншими розбиттями A на ланцюзі. Таке розбиття також називають *мінімальним ланцюговим розбиттям* (*МЛР*) множини A.

Приклад мінімального ланцюгового розбиття (МЛР)

Нехай дана множина A:
$$A=\{1,a,\angle,б,2,7,в,\triangleleft,1245,\square,д\},$$
на якій задані такі відношення часткового порядку:
$$R_1=\{(a,b)\,|\,a\le b\}$$
$$R_2=\{(a,b)\,|\,\text{"}a\ \text{слідує в алфавітному порядку за}\ b\text{"}\}$$
$$R_3=\{(a,b)\,|\,\text{"}a\ \text{має больше вуглів,}\ \textit{ніж}\ b\text{"}\}$$
Побудуємо розбиття цієї множини на ланцюзі

$A_1 = \{1,2\}; A_2 = \{7,1245\}; A_3 = \{а,б\}; A_4 = \{в,д\}; A_5 = \{\angle, \triangleleft, \square\}$

$m = 5 \quad A = \bigcup_{i=1}^{5} A_i, A_i \neq \varnothing (i = 1,...,5),$

$A_i \cap A_j = \varnothing$ якщо $i \neq j$ для всіх $i, j \in \{1,2,3,4,5\}$.

$m = 3 \quad A_1 = \{1,2,7,1245\}; A_2 = \{а,б,в,д\}; A_3 = \{\angle, \triangleleft, \square\}$

Отже, розбиття A_1, A_2, A_3 - МЛР

Визначення найбільшого елемента множини

Найбільшим елементом лінійно впорядкованої множини X відносно строгого «$<$» або нестрогого «$\leq$» упорядкування будемо називати такий елемент $a \in X$, що для будь-якого $x \in X$ вірно $x < a$ або $x \leq a$.

Теорема про єдиність найбільшого елемента.

Якщо існує найбільший елемент лінійно впорядкованої множини, то він є єдиним.

Доведення. Припустимо, що існують два найбільші елементи a і a'. Тоді для будь-якого x виконується $x \leq a$ і $x \leq a'$. Зокрема, $a \leq a'$ або $a' \leq a$.

Оскільки всі відношення порядку мають властивість антисиметричності, то з $(aRa') \wedge (a'Ra)$ слідує $a = a'$.

З $a = a'$ слідує, що якщо в лінійно впорядкованій множині існує найбільший елемент, то він єдиний. Тому, якщо говорять про найбільший елемент множини, то мають на увазі **цілком визначений** її елемент.

Приклад. Необхідно знайти найбільший елемент лінійно впорядкованої множини $X = \{1,2,15,18\}$, заданої на відношенні нестрогого порядку $a \leq b$.

Згідно з визначенням:

1. **Усі** елементи даної множини **повинні бути меншими** або дорівнювати найбільшому.
2. Найбільший елемент – **єдиний**.

Порівняємо елементи множини X:

1) $1 \geq 1$, $1 \ngeq 2$, $1 \ngeq 15$, $1 \ngeq 18$.

2) $2 \geq 1$, $2 \geq 2$, $2 \ngeq 15$, $2 \ngeq 18$.

3) $15 \geq 1$, $15 \geq 2$, $15 \geq 15$, $15 \ngeq 18$.

4) $18 \geq 1$, $18 \geq 2$, $18 \geq 15$, $18 \geq 18$.

Необхідним умовам відповідає тільки елемент 18.

Визначення максимального елемента множини

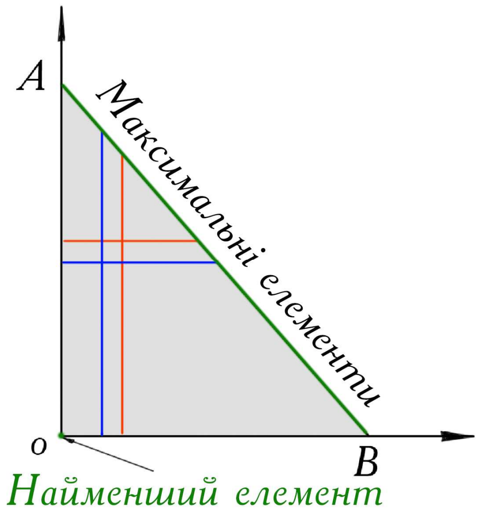
Максимальним елементом частково впорядкованої множини X відносно строгого «<» (нестрогого «$\leq$ ») впорядкування називають такий його елемент $a \in X$, для якого наявна одна із двох ситуацій:
- або $x < a$ ($x \leq a$),
- або a і x – непорівнювані.

Зауваження
На одній і тій же множині можуть бути задані **різні відношення** порядку.
За одним з них множина може бути лінійно впорядкованою, а за іншим – частково впорядкованою.
Тоді за першим відношенням будемо говорити **про найбільший елемент**, а за другим – **про максимальний**.

Визначення найменшого і мінімального елементів множини

Найменшим елементом лінійно впорядкованої множини X відносно строгого «<» (нестрогого «$\leq$ ») впорядкування будемо називати такий елемент $a \in X$, що для всіх $x \in X$ вірно $a < x$ ($a \leq x$).
Мінімальним елементом частково впорядкованої множини X відносно строгого «<» або нестрогого «$\leq$ » впорядкування називають такий його елемент $a \in X$, для якого наявна одна із двох ситуацій:
- або $a < x$, ($a \leq x$)
- або a і x – непорівнювані.

Зауваження. Якщо на множині існує найменший елемент, то він є єдиним мінімальним. Аналогічно, якщо на множині існує найбільший елемент, то він є єдиним максимальним.



Приклад. Розглянемо множину X точок трикутника OAB з наступним відношенням порядку: $(a,b) \leq (c,d)$ тоді і тільки тоді, коли $a \leq c$ і $b \leq d$.
Точка (0,0) є найменшим елементом даної множини.

Мінімальний елемент множини X – єдиний і збігається з найменшим елементом.

Максимальними елементами множини X є всі точки, що лежать на стороні AB трикутника OAB.

Найбільший елемент множини X *не існує*.

Визначення верхньої і нижньої граней множини

Визначення верхньої грані

Якщо A є частково впорядкована множина і $B \subseteq A$, то елемент $a \in A$ називають **верхньою гранню** множини B, якщо для кожного $b \in B$ існує нерівність $b \leq a$.

$A = \{1,2,3,4,5,6,7,8,9\}$, $B = \{1,2,3,4,5\}$. Верхні грані: 6,7,8,9

Визначення нижньої грані

Якщо A є частково впорядкована множина і $B \subseteq A$, то елемент $a \in A$ називають нижньою гранню множини B, якщо для кожного $b \in B$ існує нерівність $a \leq b$.

$A = \{1,2,3,4,5,6,7,8,9\}$, $B = \{5,6,7,8,9\}$. Нижні грані: 1,2,3,4

Визначення точної верхньої грані множини

$A = \{1,2,3,4,5,6,7,8,9\}$, $B = \{1,2,3,4,5\}$.

Елемент $a \in A$ називають **найменшою верхньою гранню**, якщо $a = \min\limits_i a_i$, де a_i – довільна верхня грань множини *B*. $a = \min\{6,7,8,9\} = 6$

Найменший елемент a множини всіх верхніх граней називають точною верхньою гранню або *супремумом* і позначають $\sup B$.

$A = \{1,2,3,4,5,6,7,8,9\}$, $B = \{1,2,3,4,5\}$. $\sup B = 6$

Іншими словами, найменшою верхньою гранню є така верхня грань, яка є нижньою гранню множини всіх верхніх граней.

Визначення точної нижньої грані множини

$A = \{1,2,3,4,5,6,7,8,9\}$, $B = \{5,6,7,8,9\}$.

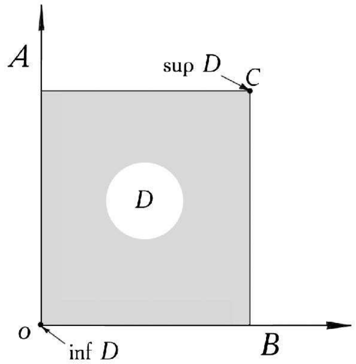
Елемент $a \in A$ називають **найбільшою нижньою гранню,** якщо $a = \max\limits_i a_i$, де a_i – довільна нижня грань множини *B*. $a = \max\{1,2,3,4\}$=4

Найбільший елемент множини всіх нижніх граней називають точною нижньою гранню або *інфімумом* і позначають $\inf B$.

$A = \{1,2,3,4,5,6,7,8,9\}$, $B = \{5,6,7,8,9\}$. $\inf B = 4$.

Іншими словами, найбільшою нижньою гранню є така нижня грань, яка є верхньою гранню множини всіх нижніх граней.

Приклад. Розглянемо множину *D* точок прямокутника *OACB* із заданим відношенням порядку:
$(a,b) \le (c,d)$ тоді і тільки тоді, коли $a \le c$ і $b \le d$.

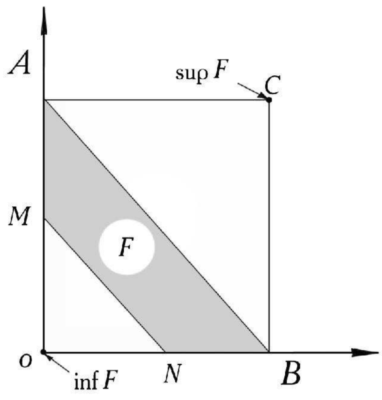


Точка *O* є точною нижньою гранню $\inf D \in D$.

Точка *C* є точною верхньою гранню $\sup D \in D$.

З рисунка видно, що обидві точки належать множині *D*.

Приклад. Розглянемо множину *F* точок трапеції *ABNM* із заданим відношенням порядку: $(a,b) \le (c,d)$ тоді і тільки тоді, коли $a \le c$ і $b \le d$.



Приклад показує, що існує точна верхня грань $\sup F$ і точна нижня грань $\inf F$. Однак жодна з граней не належить множині F.

Діаграма Хассе

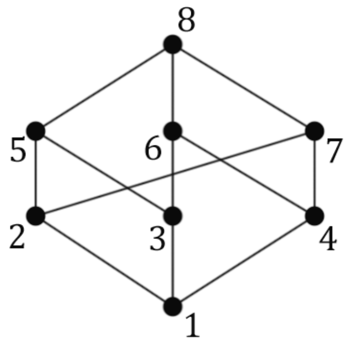
Для графічного представлення впорядкованої множини R використовують *діаграму Хассе.* Цю діаграму будують у такий спосіб.

Кожному елементу множини X ставлять у відповідність точку (кружок) на площині, причому, якщо aRb, точку, яка відповідає елементові a, розташовують нижче точки, яка відповідає елементові b. Точки $a \in X$ і $b \in X$ з'єднують лінією (ребром), якщо aRb і не існує елемента $c \in X$ такого, що aRc й cRb.

Приклад. Нехай дана множина $X = \{1,2,3,4,5,6,7,8\}$, на якій задано відношення

$$R = \{(1,2),(1,3),(1,4),(\mathbf{1,5}),(\mathbf{1,6}),(\mathbf{1,7}),(\mathbf{1,8}),$$
$$(2,5),(2,7),(\mathbf{2,8}),(3,5),(3,6),(\mathbf{3,8}),(4,6),(4,7),(\mathbf{4,8}),$$
$$(5,8),(6,8),(7,8)\}$$

Діаграма Хассе даного відношення представлена на рисунку.

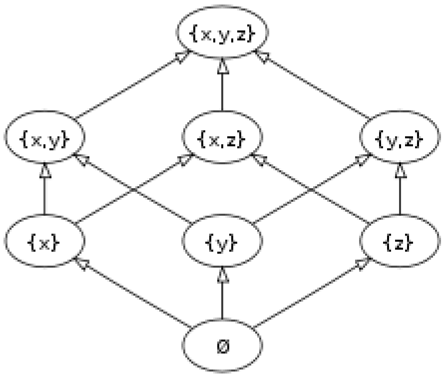


Приклад.

Нехай $C = \{x, y, z\}$,

а X - **булеан** множини З:

$$X = \{\varnothing, \{x\}, \{y\}, \{z\}, \{x,y\}, \{y,z\}, \{z,x\}, \{z,y,z\}\}$$



Визначимо відношення R на X за допомогою $(T,V) \in R$, якщо $T \subseteq V$.

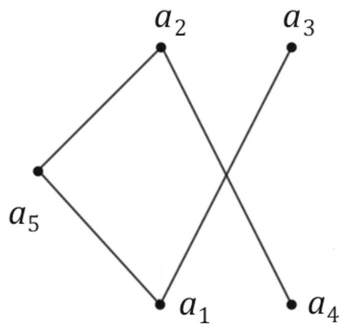
Наприклад, $(\{y\},\{x,y\}) \in R$, оскільки $\{y\} \subseteq \{x,y\}$.

Однак $(\{y,z\},\{z\}) \notin R$, оскільки $\{y,z\} \not\subset \{z\}$.

Побудувавши відношення R, можна легко перевірити, що (X, R) — частково упорядкована множина.

Приклад.
На рисунку представлений частковий порядок, породжений бінарним відношенням

$$R = \{(a_1, a_2), (a_1, a_3), (a_1, a_5), (a_4, a_2), (a_5, a_2)\}.$$



Діаграма Хассе допомагає краще розуміти взаємозв'язок елементів, що належать одній і тій же впорядкованій множині (наприклад, приналежність одного і того ж ланцюга або одного і того ж антиланцюга).

Приклад створення несуперечливих відношень S і R

Варіант 0. aSb якщо a сестра b і aRb якщо a дружина b.
Із двох відношень першим будуємо більш строге.

aRb – більш строге в порівнянні з aSb:
- $a_i -$ жінка;
- $a_i -$ може входити тільки 1 раз в $a_i R b_j$
- $b_j -$ чоловік;
- $b_j -$ може входити тільки 1 раз в $a_i R b_j$

aSb – менш строге в порівнянні з aRb:
- $a_i -$ жінка;
- $a_i -$ може входити багато разів в $a_i R b_j$
- $b_j -$ чоловік або жінка;
- $b_j -$ може входити багато разів в $a_i R b_j$

Опис структури даних

A: Array [0..19] of String;

B: Array [0..19] of String;
масив елементів множин A и B

Gender_A: Array [0..19] of Byte;
Gender_B: Array [0..19] of Byte;
масив гендерної приналежності: 0-жінка; 1-чоловік.

Alen і Blen кількості елементів у множинах A и B.

R_relation[i,j] матриця відношення R.
S_relation[i,j] матриця відношення S.

```
procedure R_create; (*Варіант 0 - a дружина b*)
var k:integer;
   used:Boolean;
begin
 For i:=0 to Alen-1 do     (* Переглядаємо A  *)
 If Gender_A[i]=0 then     (* Вибираємо жінку *)
 For j:=0 to Blen-1 do     (* Переглядаємо B  *)
 If Gender_B[j]=1 then    (* Вибираємо чоловіка *)
 begin
  used:=false;
  For k:=0 to Alen-1 do  (*Перевіряємо на зайнятість*)
  If R_relation[k,j]=1 then used:=true;
  If Not used then
  begin  R_relation[i,j]:=1;   Break; end;
 end;
end;
```

```
procedure S_create; (* Варіант 0 - a сестра b*)
begin
 For i:=0 to Alen-1 do    (* Переглядаємо A  *)
 If Gender_A[i]=0 then   (* Вибираємо жінку *)
 For j:=0 to Blen-1 do   (* Переглядаємо B  *)
 (* Якщо жінка, то вибираємо відразу її сестрою a  *)
 (* Якщо чоловік і він не чоловік a, вибираємо його братом a  *)
 If Gender_B[0]=0 then S_Relation[i,j]:=1 else
 If R_relation[i,j]=0 then S_Relation[i,j]:=1;
end;
```

Варіант 1.   aSb якщо а мати b і aRb якщо а внучка b.

Із двох відношень першим будуємо більш строге.

aRb – більш строге в порівнянні з aSb:
- a_i – жінка;
- a_i – може входити тільки 2 рази в a_iRb_j;
- b_j – жінка;
- b_j – може входити багато разів в. a_iRb_j

aSb – менш строге в порівнянні з aRb:
- a_i – жінка;
- a_i – може входити багато раз в a_iRb_j ;
- b_j – чоловік або жінка;
- b_j – може входити багато разів в. a_iRb_j

```
procedure R_create; (*Варіант 1 - а внучка b*)
var k:integer;
   used:Byte;
begin
 For i:=0 to Alen-1 do    (* Переглядаємо А  *)
 If Gender_A[i]=0 then    (* Вибираємо жінку *)
 For j:=0 to Blen-1 do    (* Переглядаємо В  *)
 If Gender_B[j]=0 then    (* Вибираємо жінку *)
 begin
  used:=0;
  For k:=0 to Alen-1 do    (*Тільки дві бабусі*)
  If R_relation[k,j]=1 then Inc(used);
  If used<2 then
  begin  R_relation[i,j]:=1;   Break; end;
 end;
end;
```

```
procedure S_create; (*Варіант 1 - а мати b*)
var used:Boolean;
begin
 For i:=0 to Alen-1 do   (* Переглядаємо А  *)
 If Gender_A[i]=0 then   (* Вибираємо жінку *)
 For j:=0 to Blen-1 do   (* Переглядаємо В  *)
 begin
```

```
  used:=false;
  For k:=0 to Alen-1 do    (*b уже має матір?*)
  If s_relation[k,j]=1 then used:=true;
  If not used then
  If (Gender_B[0]=1) then
  begin S_Relation[i,j]:=1; Break; end  else
  If R_relation[i,j]=0 then
  begin S_Relation[i,j]:=1; Break; end;
 end;
end;
```

Варіант 2. aSb якщо a дружина b і aRb якщо a мати b.

aSb – більш строге в порівнянні з aRb:

- a_i – жінка;
- a_i – може входити тільки 1 раз в a_iRb_j
- b_j – чоловік;
- b_j – може входити тільки 1 раз в a_iRb_j

aRb – менш строге в порівнянні з aSb:

- a_i – жінка;
- a_i – може входити багато разів в a_iRb_j
- b_j – чоловік або жінка;
- b_j – може входити багато разів в a_iRb_j

```
procedure S_create; (*Варіант 2 - а дружина b*)

var k:integer;
   used:Boolean;
begin
 For i:=0 to Alen-1 do    (* Переглядаємо A  *)
 If Gender_A[i]=0 then    (* Вибираємо жінку *)
 For j:=0 to Blen-1 do    (* Переглядаємо B  *)
 If Gender_B[j]=1 then    (* Вибираємо чоловіка *)
 begin
  used:=false;
  For k:=0 to Alen-1 do  (*Перевіряємо на зайнятість*)
  If S_relation[k,j]=1 then used:=true;
  If Not used then
```

```
  begin  S_relation[i,j]:=1;   Break; end;
 end;
end;

procedure R_create; (*Варіант 1 - а мати b*)
var used:Boolean;
begin
 For i:=0 to Alen-1 do    (* Переглядаємо A  *)
 If Gender_A[i]=0 then   (* Вибираємо жінку *)
 For j:=0 to Blen-1 do   (* Переглядаємо B  *)
 begin
  used:=false;
  For k:=0 to Alen-1 do     (*b уже має матір?*)
  If R_relation[k,j]=1 then used:=true;
  If not used then
  If (Gender_B[0]=1) then
  begin R_Relation[i,j]:=1; Break; end  else
  If S_relation[i,j]=0 then
  begin R_Relation[i,j]:=1; Break; end;
 end;
end;
```

Варіант 9. aSb **якщо а свекруха b,** aRb **якщо а батько b.**

Примітка. Свекруха – мати чоловіка.

aSb – більш строге в порівнянні з aRb:

- a_i – жінка;
- a_i – може входити багато раз в a_iRb_j
- b_j – жінка;
- b_j – може входити тільки 1 раз в a_iRb_j

aRb – менш строге в порівнянні з aSb:

- a_i – чоловік;
- a_i – може входити багато раз в a_iRb_j
- b_j – чоловік або жінка;
- b_j – може входити 1 раз в a_iRb_j

```
procedure S_create; (*Варіант 9 - а свекруха b*)
(*Розглянутий варіант-у свекрухи один син*)
var k:integer;
```

```
   used:Boolean;
begin
 For i:=0 to Alen-1 do     (* Переглядаємо A  *)
 If Gender_A[i]=0 then     (* Вибираємо жінку *)
 For j:=0 to Blen-1 do     (* Переглядаємо B  *)
 If Gender_B[j]=0 then     (* Вибираємо жінку *)
 begin
  used:=false;
  For k:=0 to Alen-1 do  (*Перевіряємо на зайнятість*)
  If S_relation[k,j]=1 then used:=true;
  If Not used then
  begin  S_relation[i,j]:=1;   Break; end;
 end;
end;

procedure R_create; (*Варіант 9 - a батько b*)
var used:Boolean;
begin
 For i:=0 to Alen-1 do    (* Переглядаємо A  *)
 If Gender_A[i]=1 then   (* Вибираємо чоловіка *)
 For j:=0 to Blen-1 do   (* Переглядаємо B  *)
 begin
  used:=false;
  For k:=0 to Alen-1 do     (*b уже має батька?*)
  If R_relation[k,j]=1 then used:=true;
  If not used then
  begin R_Relation[i,j]:=1; Break; end;
 end;
end;
```