

5.5 Деревя

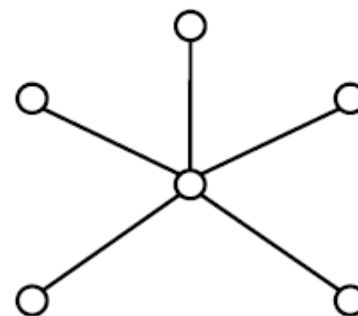
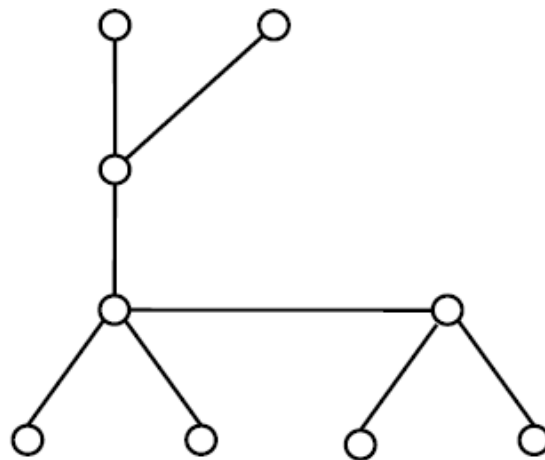
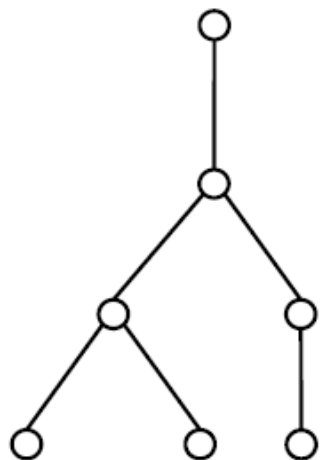
Дерева використовують як інструмент обчислень, зручний спосіб збереження даних, їх сортування чи пошуку.

5.5.1 Основні означення та властивості

Деревом називають зв'язний граф без простих циклів.

Граф, який не містить простих циклів і складається з k компонент зв'язності, називають **лісом** з k дерев.

Приклади



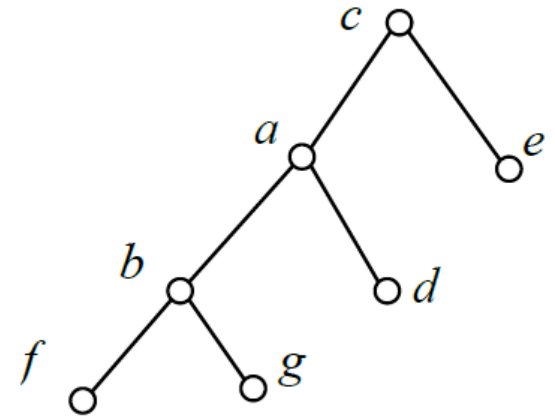
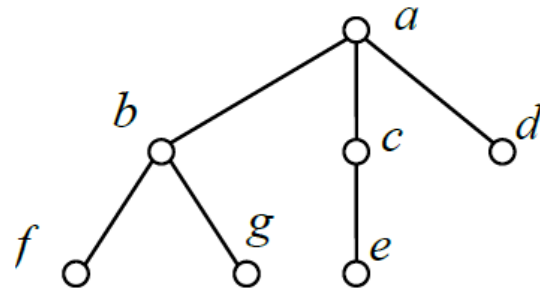
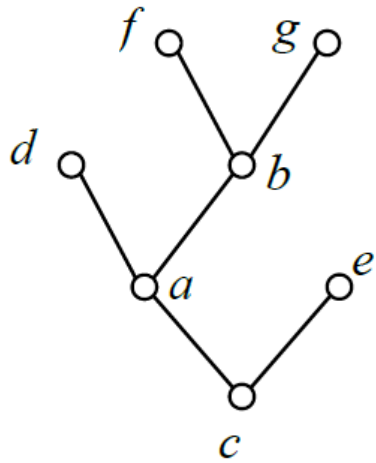
Теорема 1. Нехай граф G має n вершин. Тоді такі твердження еквівалентні:

- 1) граф G — дерево;
- 2) граф G не містить простих циклів і має $(n-1)$ ребро;
- 3) граф G зв'язний і має $(n-1)$ ребро;
- 4) граф G зв'язний, але видалення довільного ребра робить його незв'язним;
- 5) довільні дві вершини графу G з'єднані точно одним простим маршрутом;
- 6) граф G не містить простих циклів, але додавши до нього нове ребро, отримаємо точно один простий цикл.

У багатьох застосуваннях певну вершину дерева означають як **корінь**.

Дерево разом із виділеним коренем утворює орієнтований граф, який називають **кореневим деревом**.

Приклад



Нехай G — кореневе дерево.

Якщо v — його вершина, відмінна від кореня, то її **батьком** називають єдину вершину u таку, що є орієнтоване ребро (u, v) .

Якщо u — батько, то v — **син**.

Вершини дерева, які не мають синів,
називаються **листками**.

Вершини, які мають синів, називаються
внутрішніми.

Нехай a — вершина дерева.

Піддеревом із коренем a називають підграф, який містить a та всі вершини — нащадки вершини a , а також інцидентні їм ребра.

Кореневе дерево називають ***m*-арним**, якщо кожна його внутрішня вершина має не більше ніж ***m*** синів.

Дерево називають **повним *m*-арним**, якщо кожна його внутрішня вершина має точно ***m*** синів.

У разі **$m=2$** дерево називають **бінарним**.

Кореневе дерево, у якому сини кожної внутрішньої вершини впорядковано, називають **упорядкованим**.

Якщо внутрішня вершина впорядкованого бінарного дерева має двох синів, то першого називають **лівим**, а другого — **правим**.

Рівнем вершини v в кореневому дереві називають довжину простого шляху від кореня до цієї вершини.

Висотою кореневого дерева називають максимальний із рівнів його вершин.

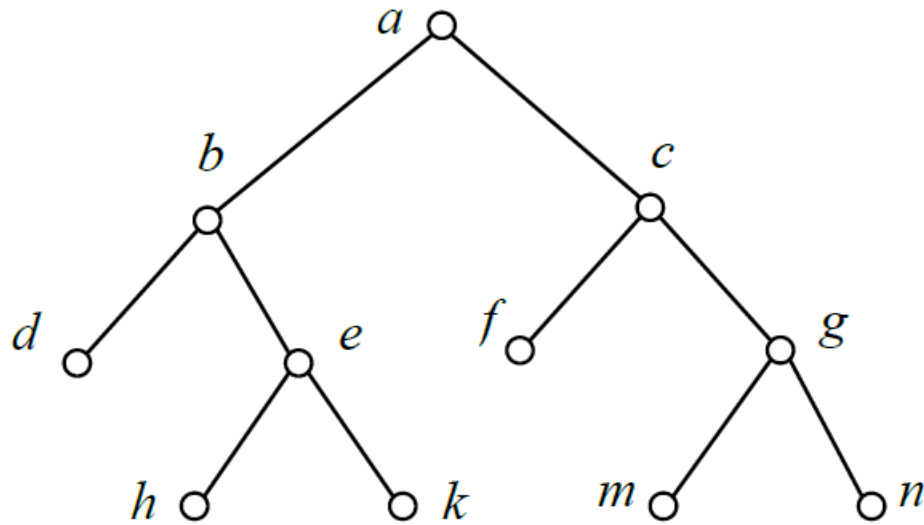
Висота кореневого дерева — це довжина найдовшого простого шляху від кореня до будь-якої вершини.

5.5.2 Обхід дерев

Нехай R — корінь бінарного дерева,

A та B — ліве та праве піддерева.

1. Обхід у прямому порядку (preorder), або згори донизу: R, A, B (корінь відвідують до обходу піддерев).
2. Обхід у внутрішньому порядку (inorder), або зліва направо: A, R, B .
3. Обхід у зворотному порядку (postorder), або знизу догори: A, B, R (корінь відвідують після обходу піддерев).



Приклад.

Обхід у прямому
порядку або згори
донизу:

a, b, d, e, h, k, c, f, g, m, n.

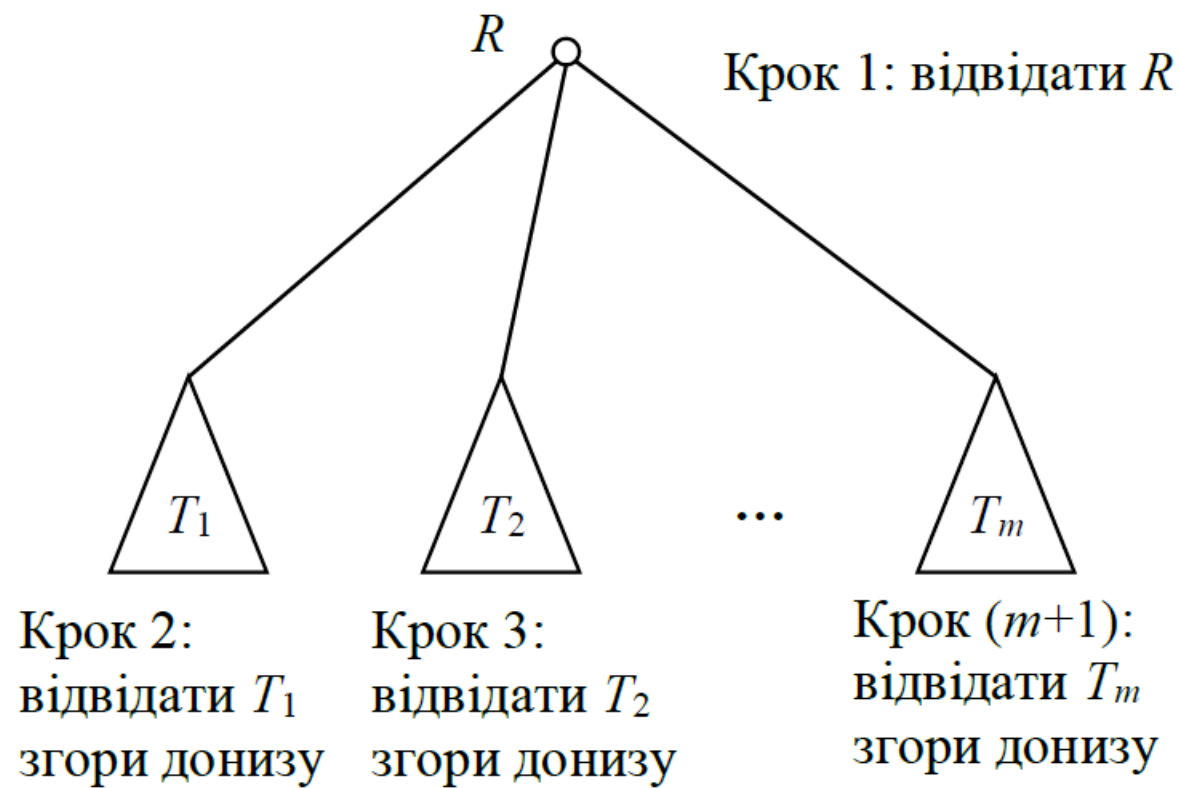
Обхід у внутрішньому порядку або зліва
направо:

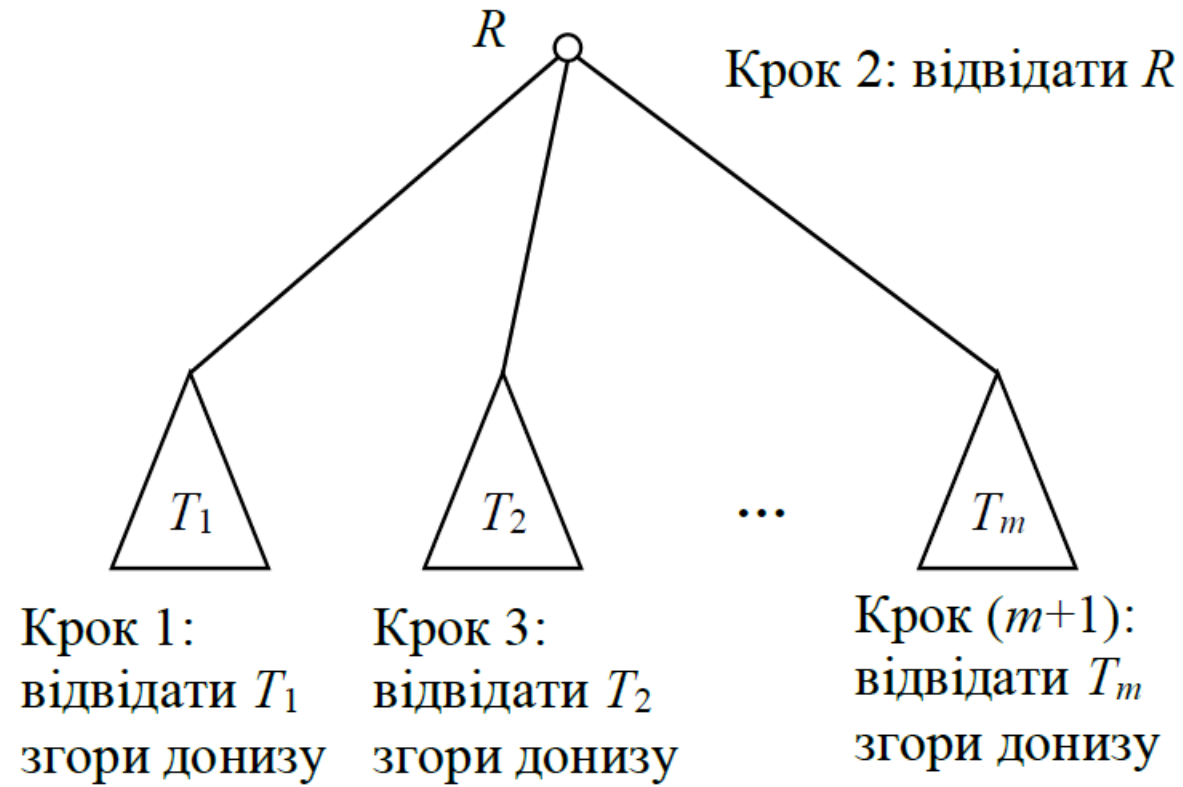
d, b, h, e, k, a, f, c, m, g, n.

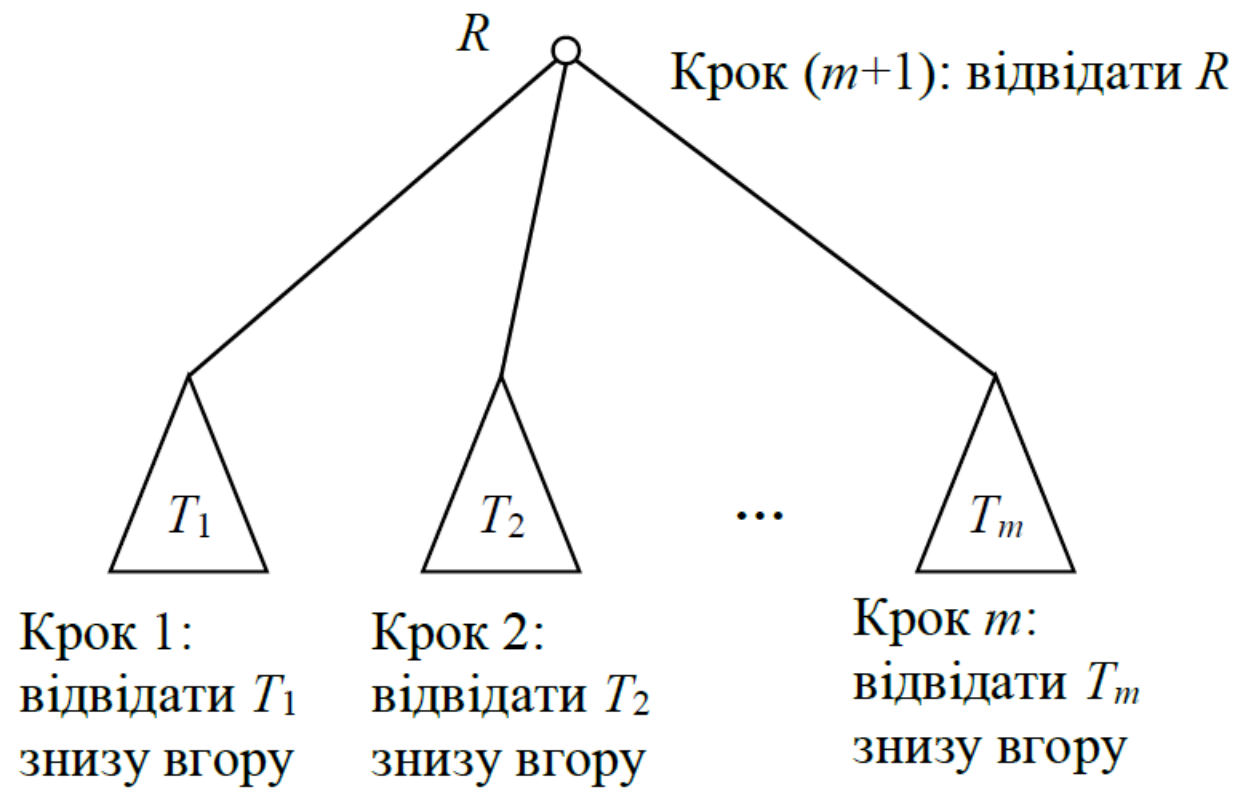
Обхід у зворотному порядку або знизу догори:

d, h, k, e, b, f, m, n, g, c, a.

Способи обходу бінарних дерев можна
узагальнити й на довільні m -арні дерева.



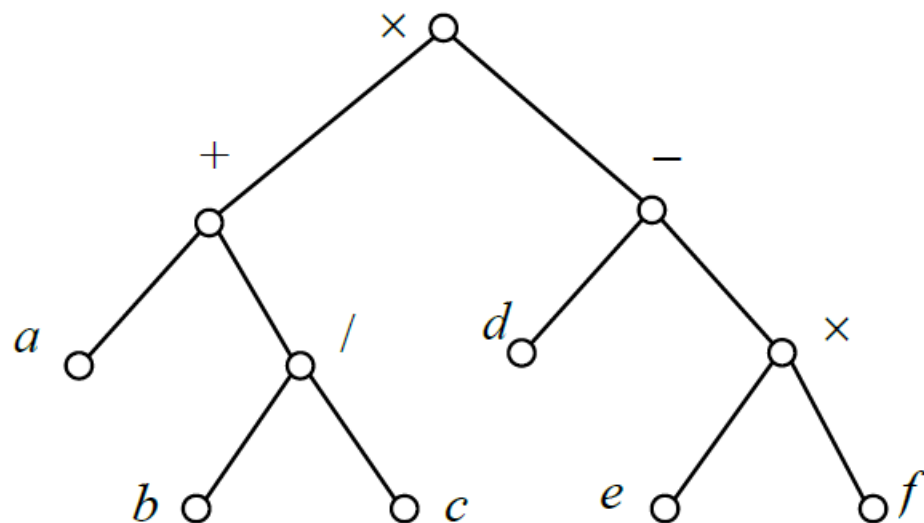




5.5.3 Приклад застосування дерев:

зіставлення виразам дерев і побудова на цій основі різних форм запису виразів

Приклад. $\left(a + \frac{b}{c}\right) \times (d - e \times f)$



Обхід в прямому порядку — **префіксний**
(польський) запис:

$$\times + a / bc - d \times ef.$$

Обхід у внутрішньому порядку — **інфіксний**
запис:

$$a + b / c \times d - e \times f.$$

Обхід в зворотному порядку — **постфіксний**
(зворотний польський) запис:

$$abc / + def \times - \times.$$

Інфіксна форми запису:

$$((a + (b / c)) \times (d - (e \times f))).$$

Для обчислення значення виразу в польському записі його проглядають справа наліво та знаходять два операнди разом зі знаком операції перед ними. Ці операнди та знак операції вилучають із запису, виконують операцію, а її результат записують на місце вилучення символів.

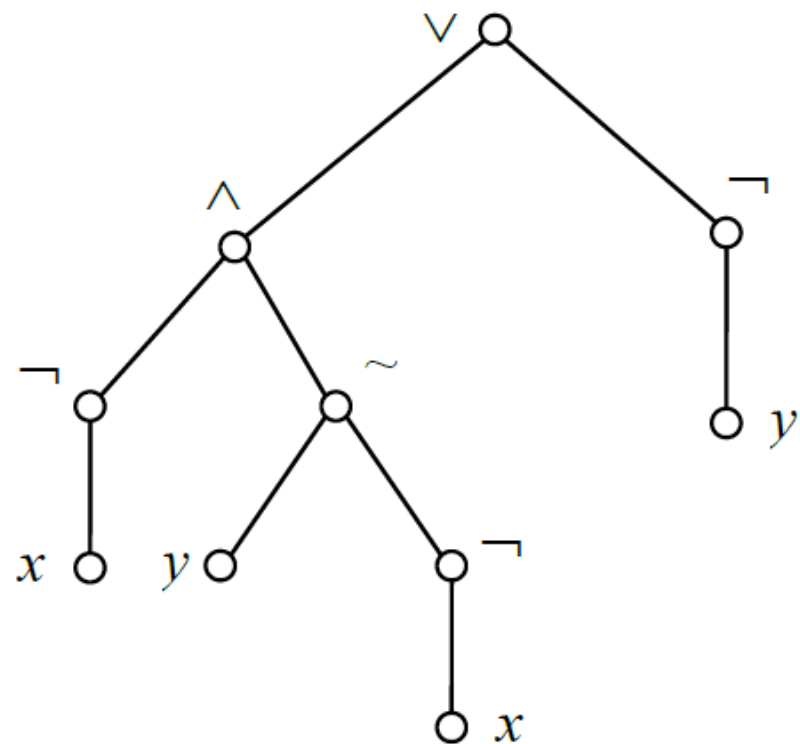
Приклад. Обчислити: « $+ \times - 4 2 5 ^ 2 / 9 3$ »

Крок	Вираз	Виділені символи	Виконання операції
1	$+ \times - 4 2 5 ^ 2 / 9 3$	$/ 9 3$	$9 / 3 = 3$
2	$+ \times - 4 2 5 ^ 2 3$	$^ 2 3$	$2 ^ 3 = 8$
3	$+ \times - 4 2 5 8$	$- 4 2$	$4 - 2 = 2$
4	$+ \times 2 5 8$	$\times 2 5$	$2 \times 5 = 10$
5	$+ 10 8$	$+ 10 8$	$10 + 8 = 18$
6	18		

Для обчислення значення виразу в зворотному польському записі його переглядають зліва направо та виділяють два операнди разом зі знаком операції після них. Ці операнди та знак операції вилучають, а її результат записують на місце вилучених символів.

Польські записи широко використовують у
комп'ютерних науках, особливо для
конструювання компіляторів.

Приклад. $(\neg x \wedge (y \sim \neg x)) \vee \neg y$



Польський запис: « $\vee \wedge \neg x \sim y \neg x \neg y$ ».

Зворотній польський запис: « $x \neg y x \neg \sim \wedge y \neg \vee$ ».

Правило обчислення результату булевих виразів для прямої польської форми

Переглядаємо послідовність справа наліво та виділяємо один операнд та операцію заперечення « \neg », яка стоїть перед цим операндом, або два операнди та операцію, відмінну від заперечення та яка стоїть перед цими операндами.

Вибрані операнд(и) та операцію замінюємо в послідовності на результат застосування операції до операнда(ів).

Аналогічно формулюється правило обчислення результату булевих виразів у зворотній польській формі.

Приклад. Обчислити значення функції на наборі (1, 0), використовуючи прямий польський запис.

Крок	Вираз	Виділені символи	Виконання операції
1	$\vee \wedge \neg 1 \sim 0 \neg 1 \neg 0$	$\neg 0$	$\neg 0 = 1$
2	$\vee \wedge \neg 1 \sim 0 \neg 1 1$	$\neg 1$	$\neg 1 = 0$
3	$\vee \wedge \neg 1 \sim 0 0 1$	$\sim 0 0$	$0 \sim 0 = 1$
4	$\vee \wedge \neg 1 1 1$	$\neg 1$	$\neg 1 = 0$
5	$\vee \wedge 0 1 1$	$\wedge 0 1$	$0 \wedge 1 = 0$
6	$\vee 0 1$	$\vee 0 1$	$0 \vee 1 = 1$
7	1		

Приклад. Обчислити значення функції на наборі (0, 0), використовуючи зворотній польський запис.

Крок	Вираз	Виділені символи	Виконання операції
1	$0 \neg 0 0 \neg \sim \wedge 0 \neg \vee$	$0 \neg$	$\neg 0 = 1$
2	$1 0 0 \neg \sim \wedge 0 \neg \vee$	$0 \neg$	$\neg 0 = 1$
3	$1 0 1 \sim \wedge 0 \neg \vee$	$0 1 \sim$	$0 \sim 1 = 0$
4	$1 0 \wedge 0 \neg \vee$	$1 0 \wedge$	$1 \wedge 0 = 0$
5	$0 0 \neg \vee$	$0 \neg$	$\neg 0 = 1$
6	$0 1 \vee$	$0 1 \vee$	$0 \vee 1 = 1$
7	1		