Large-scale Knowledge Processing

Lecture 11

Kazuhisa Seto

Today's Lecture

Study Boolean functions and its representations.

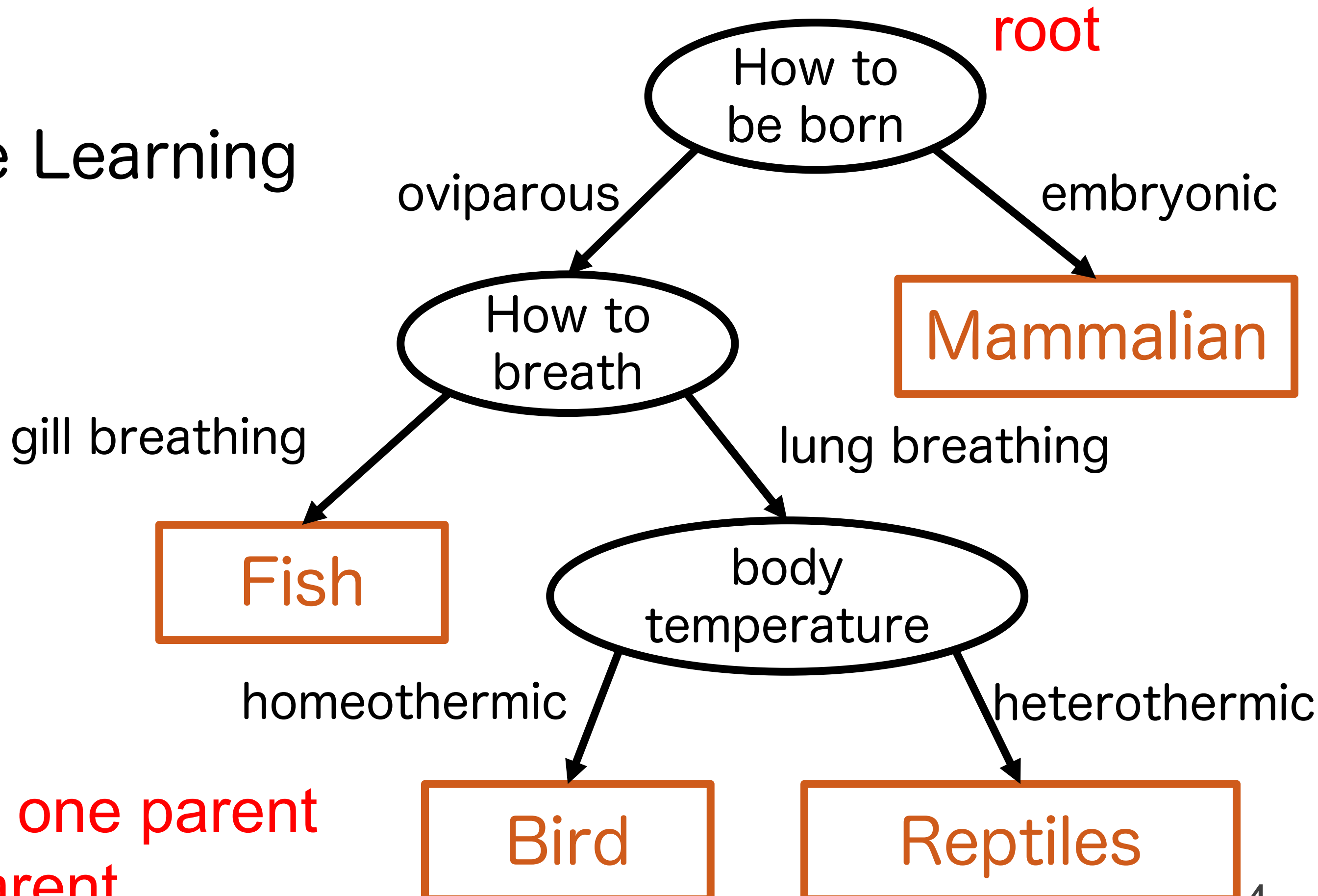
- Boolean Function
- Truth Table
- Boolean Formula（CNF，DNF）
- Karnaugh Map
- Boolean Circuit
- Binary Decision Tree
- BDD（Binary Decision Diagram）

Binary Decision Tree (1)

Decision Tree

A tree categorizing items by explanatory variable

Use for Machine Learning



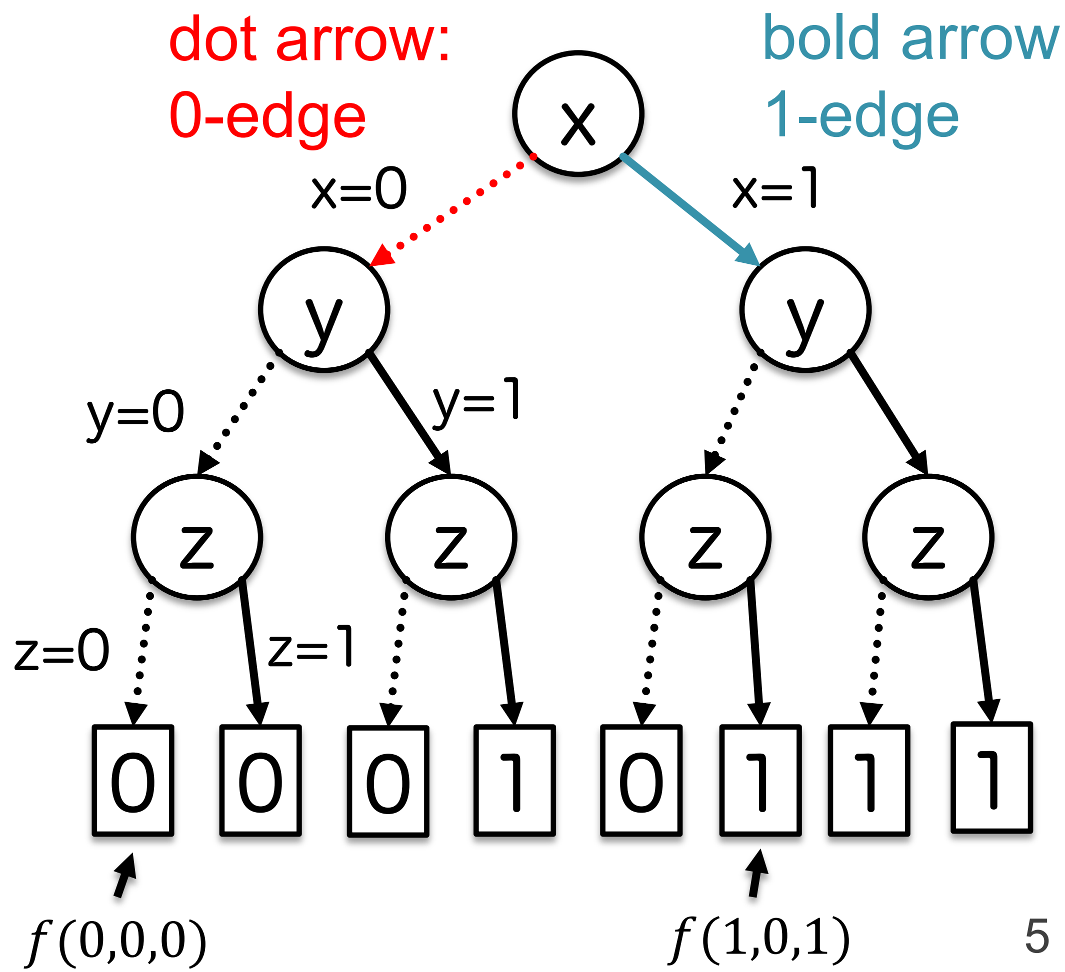
Tree:
Any node has only one parent
The root has no parent

From Truth Table to Binary Decision Tree

We can easily construct a binary decision tree from a truth table.

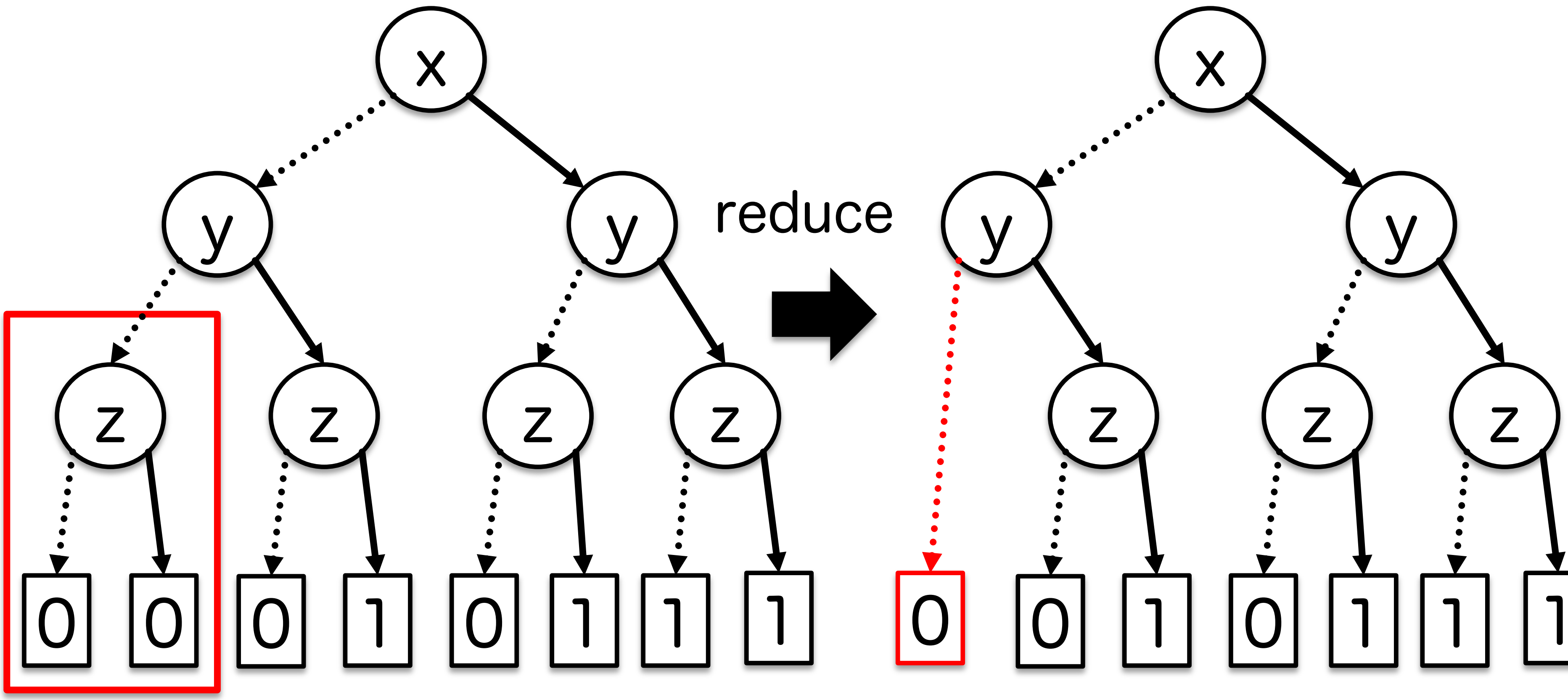
x	y	z	$f(x,y,z)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

→



Reducing Binary Decision Tree

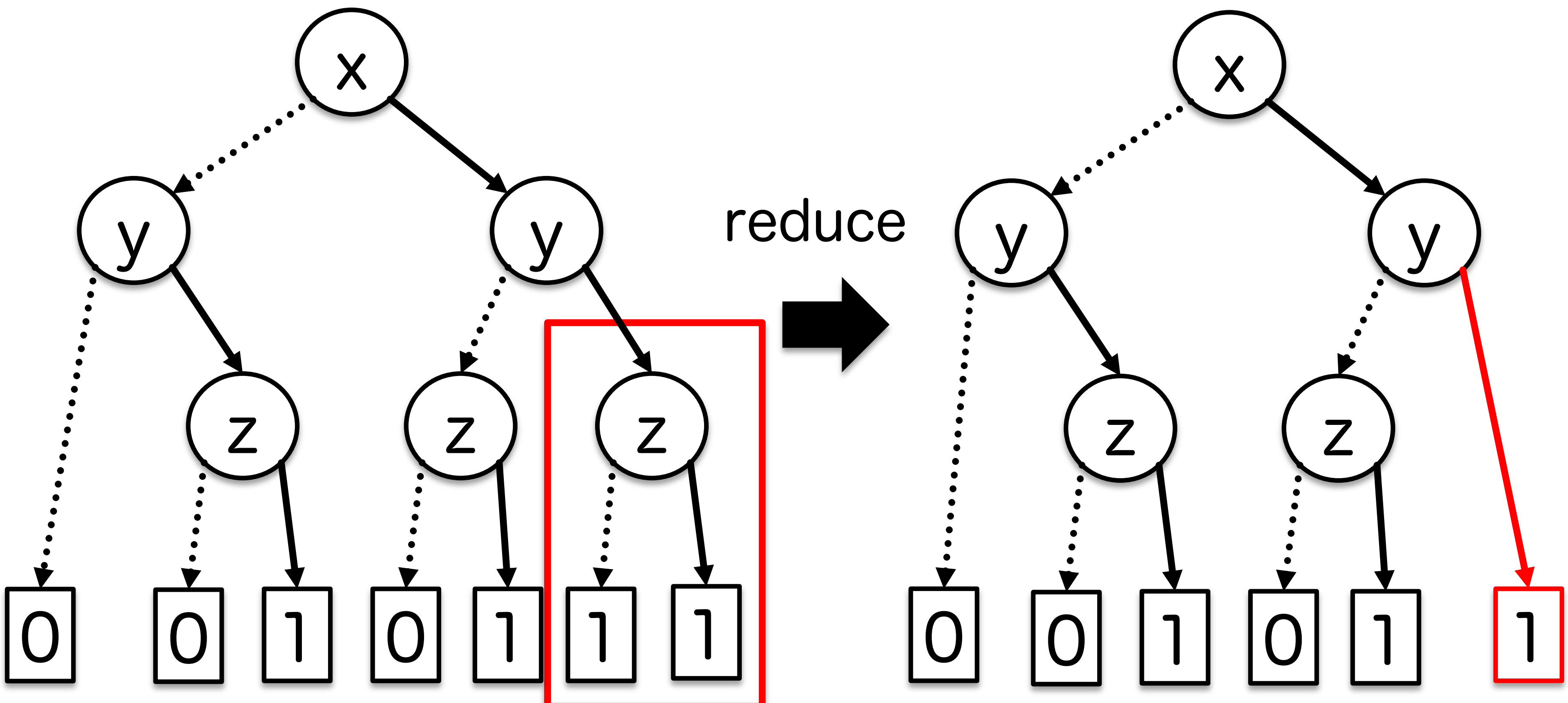
We can often reduce a binary decision tree.



Both case that z =0 and z=1, the output is 0 → we remove the variable z.

Reducing Binary Decision Tree

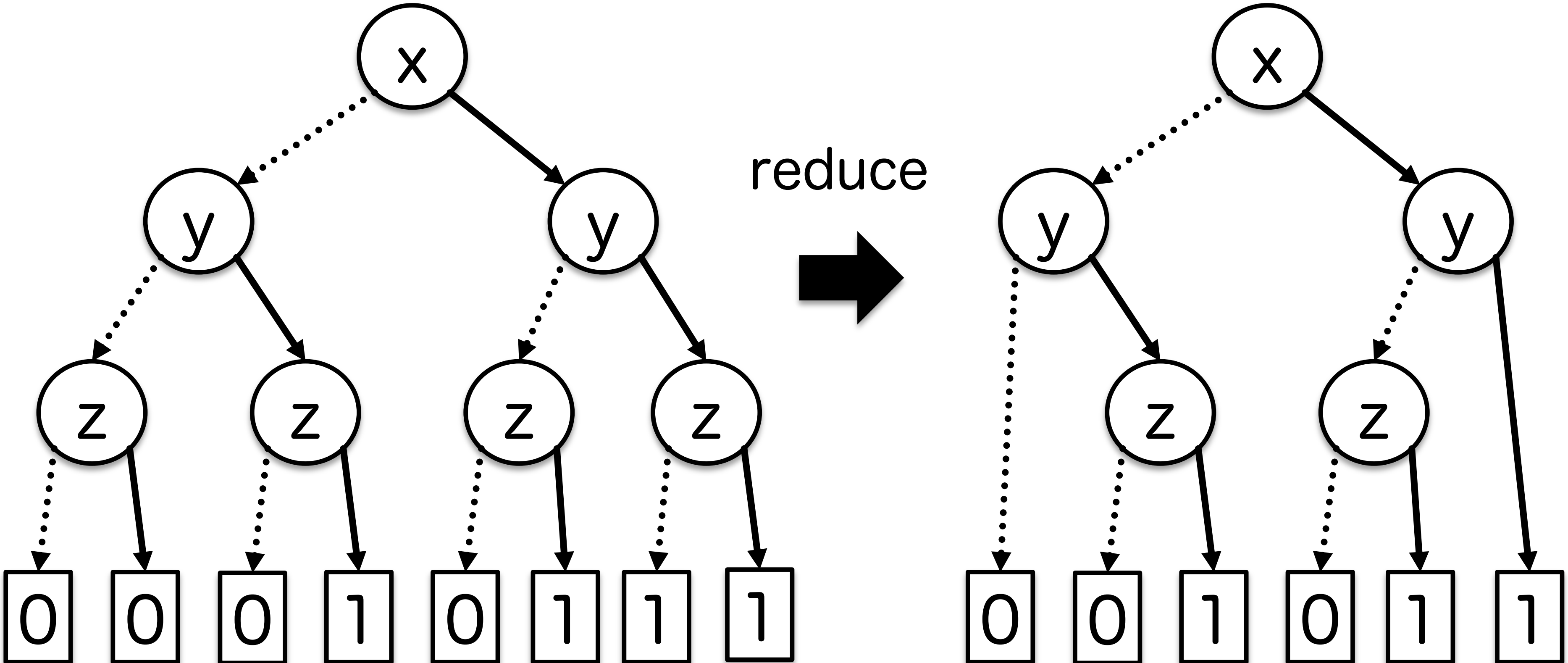
We can often reduce a binary decision tree.



Both case that z =0 and z=1, the output is 1 → we remove the variable z.

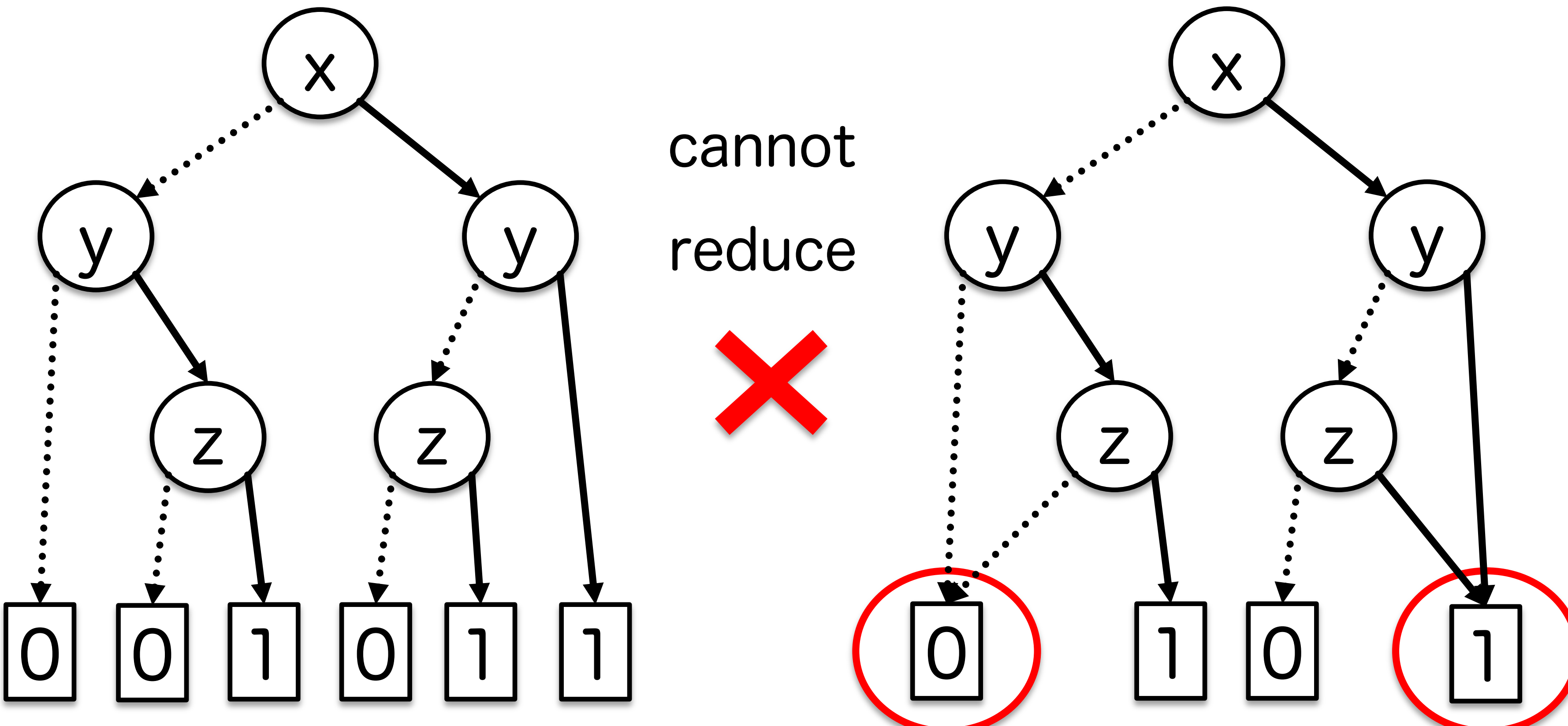
Reducing Binary Decision Tree

We can often reduce a binary decision tree.



Decision Tree must be tree

Since red nodes has two parents,
this is not tree!



Exercise1

Construct decision trees corresponding to the following two Boolean functions.

AND function

x	y	z	$x \wedge y \wedge z$
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

OR function

x	y	z	$x \vee y \vee z$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Exercise 1 (Answer)

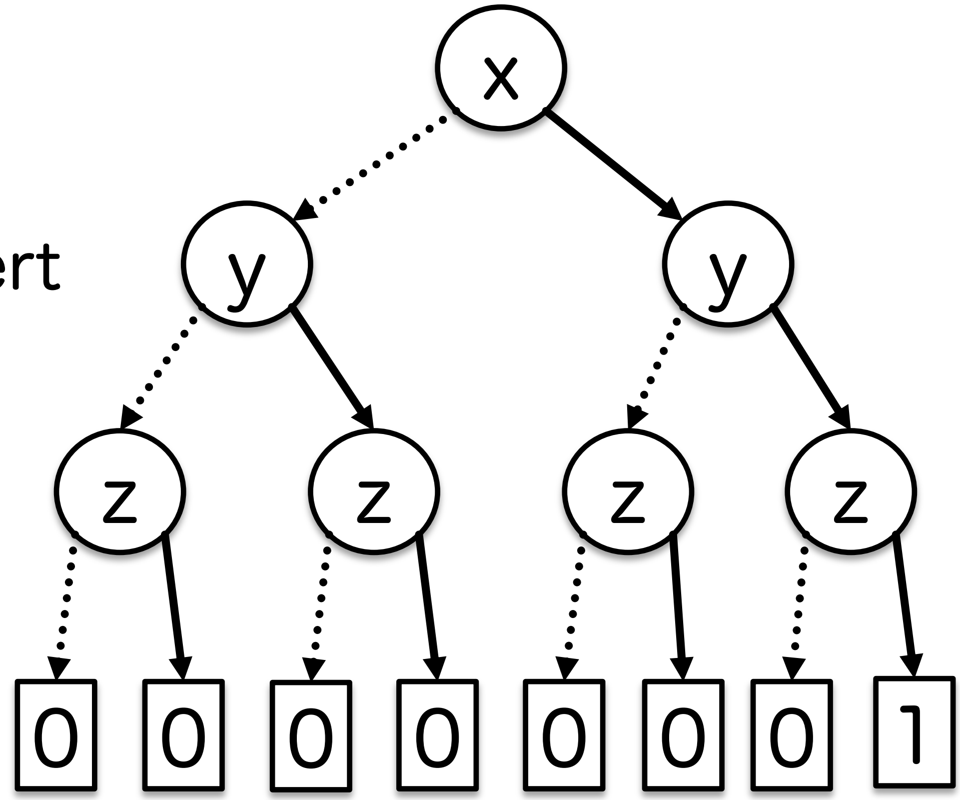
AND function

dot arrow : 0-edge

bold arrow : 1-edge

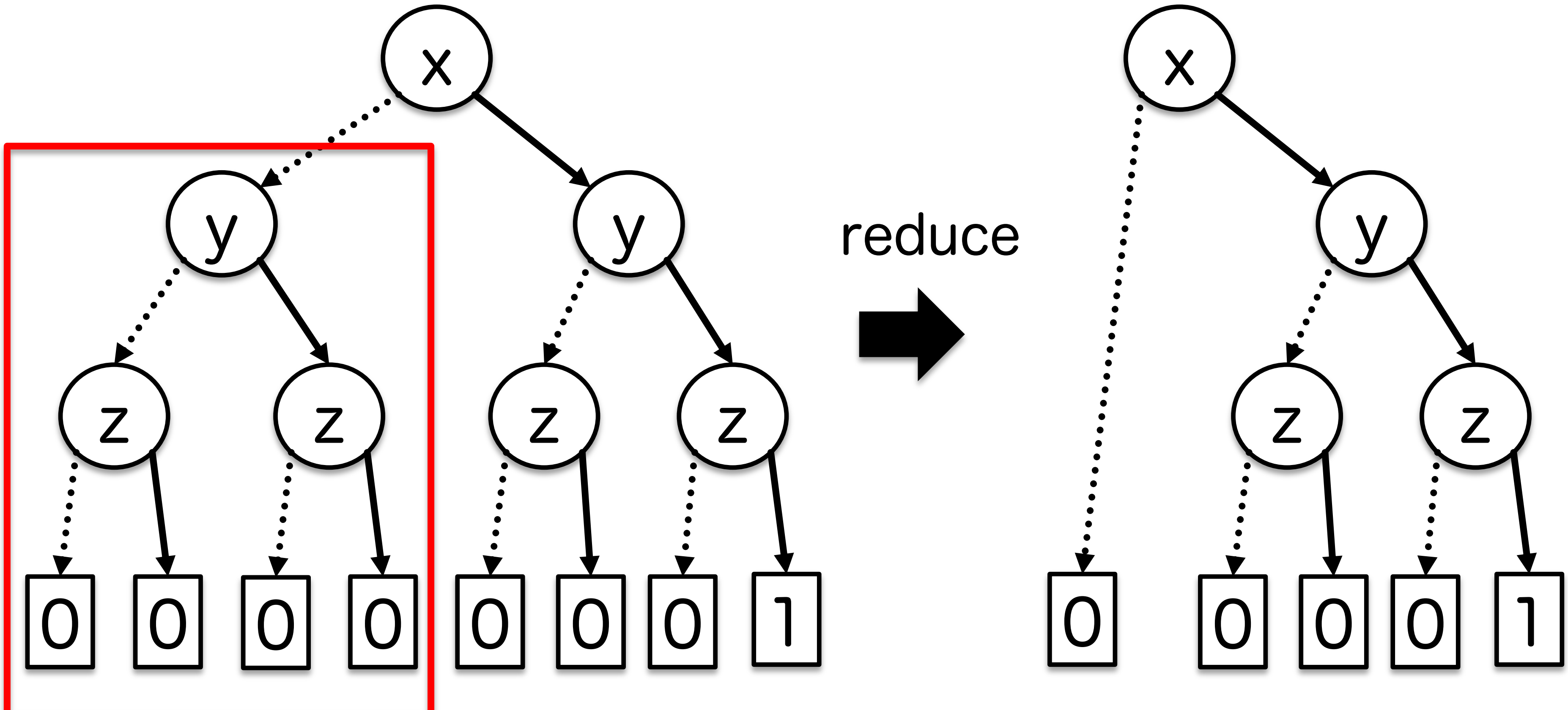
x	y	z	$x \wedge y \wedge z$
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

convert



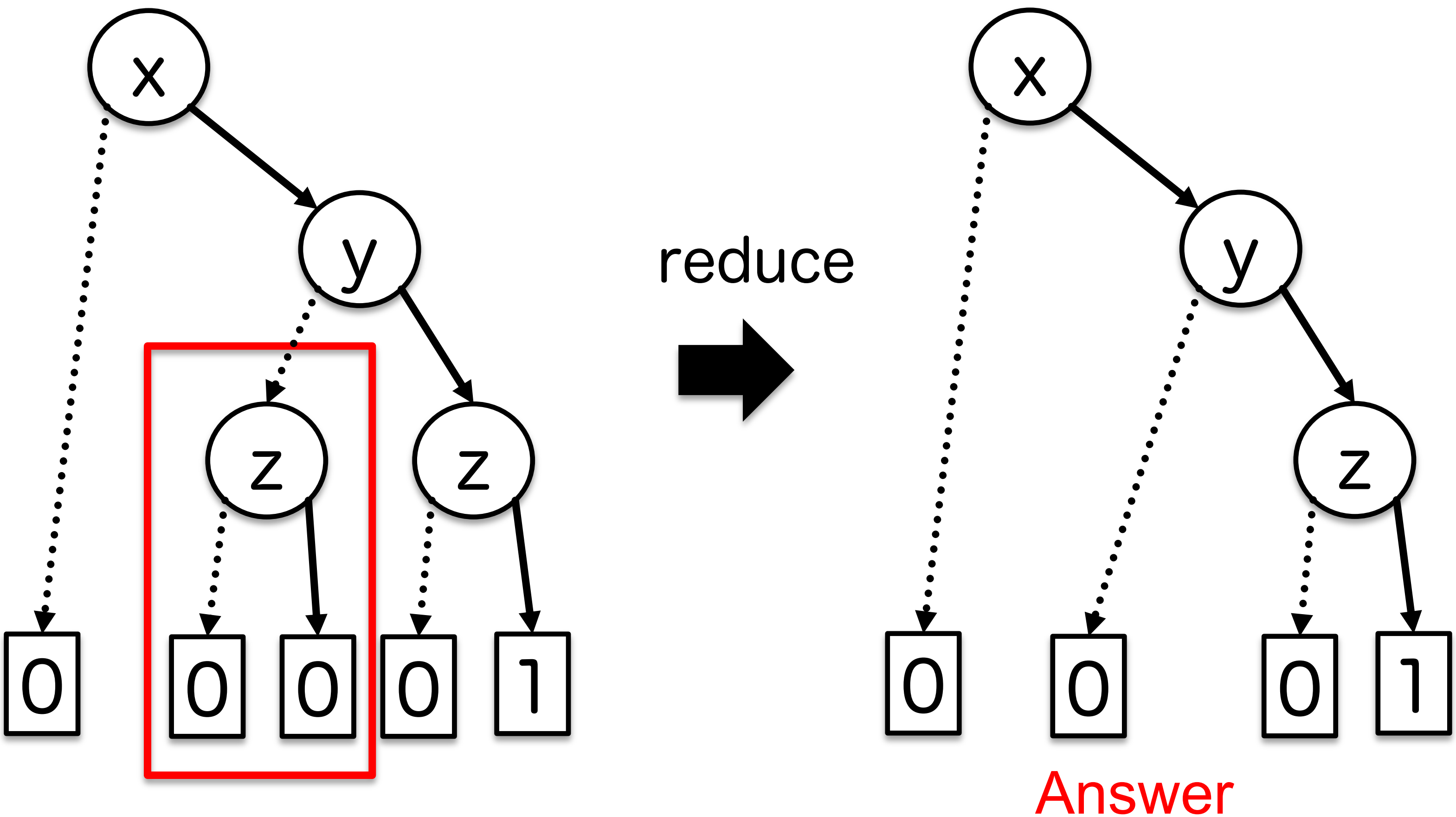
Exercise 1 (Answer)

Reduce Phase 1



Exercise 1 (Answer)

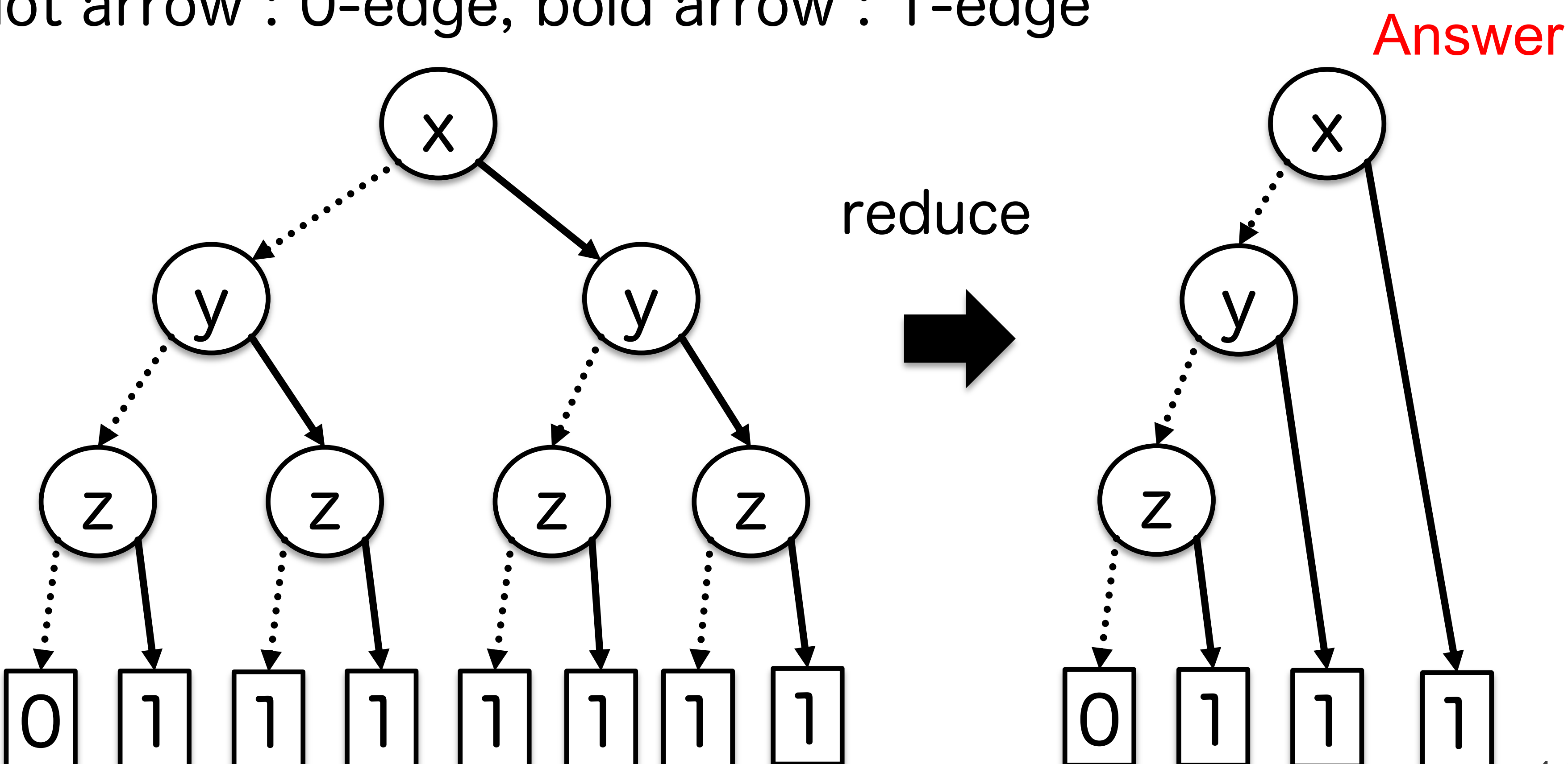
Reduce Phase 2



Exercise 1（Answer)

OR function: Almost similar construction of AND function.

dot arrow : 0-edge, bold arrow : 1-edge

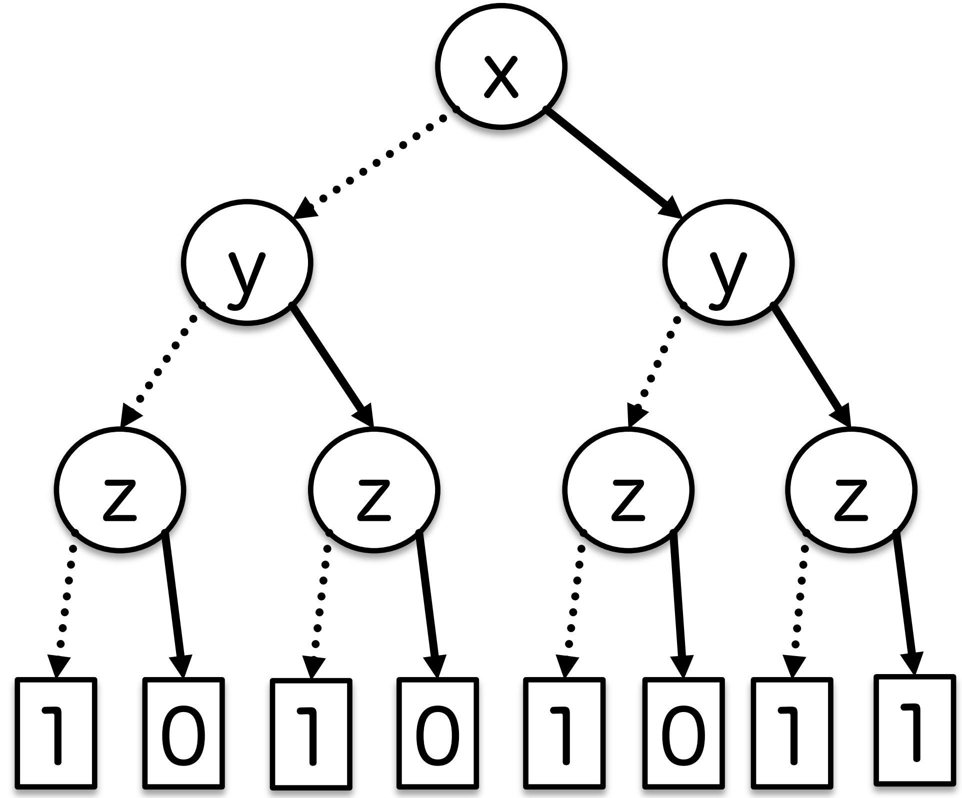


Binary Decision Tree (2)

Variable Ordering is important

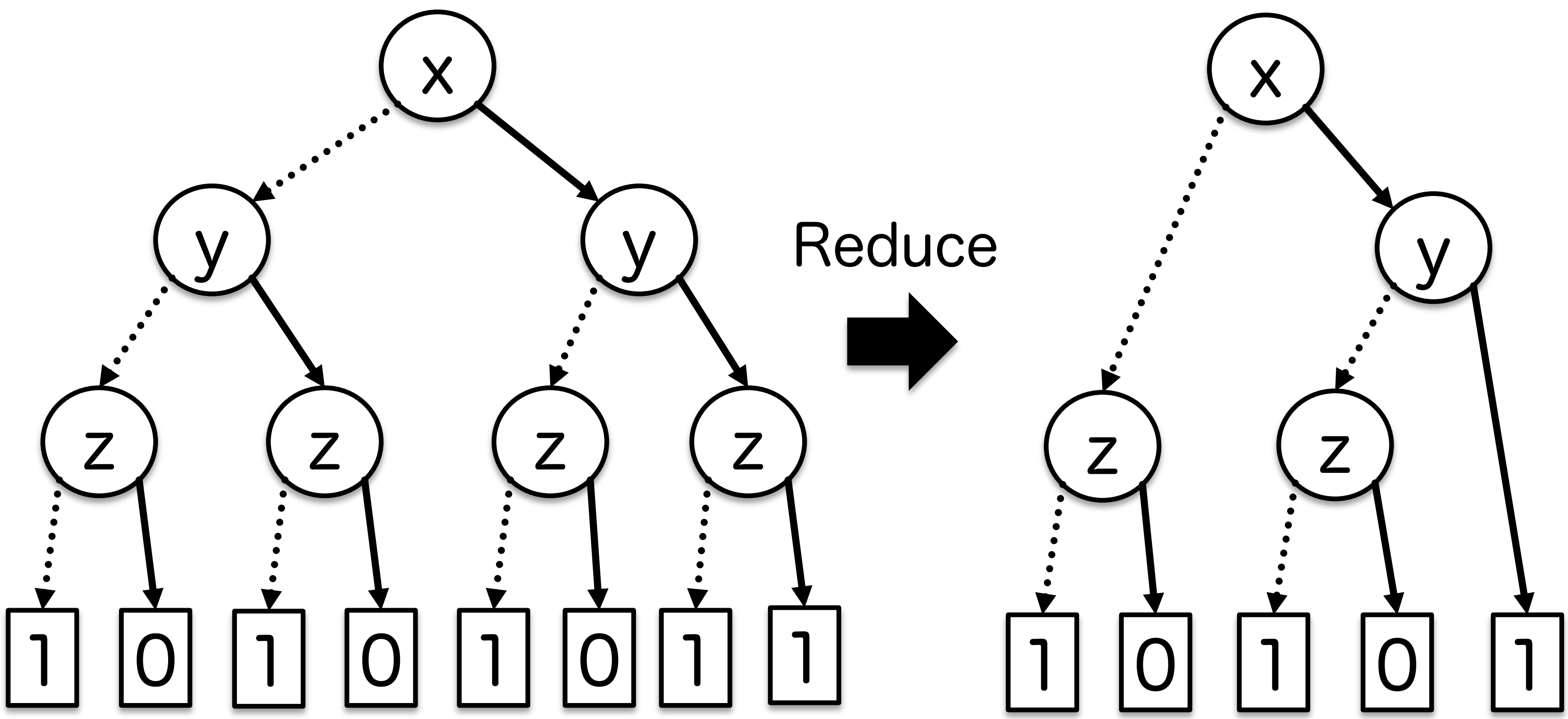
We construct a decision tree of the following truth table that checks the values of the variables in x, y, z.

x	y	z	f
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



Variable Ordering is important

After reducing, the number of nodes labeled with variable is 4.

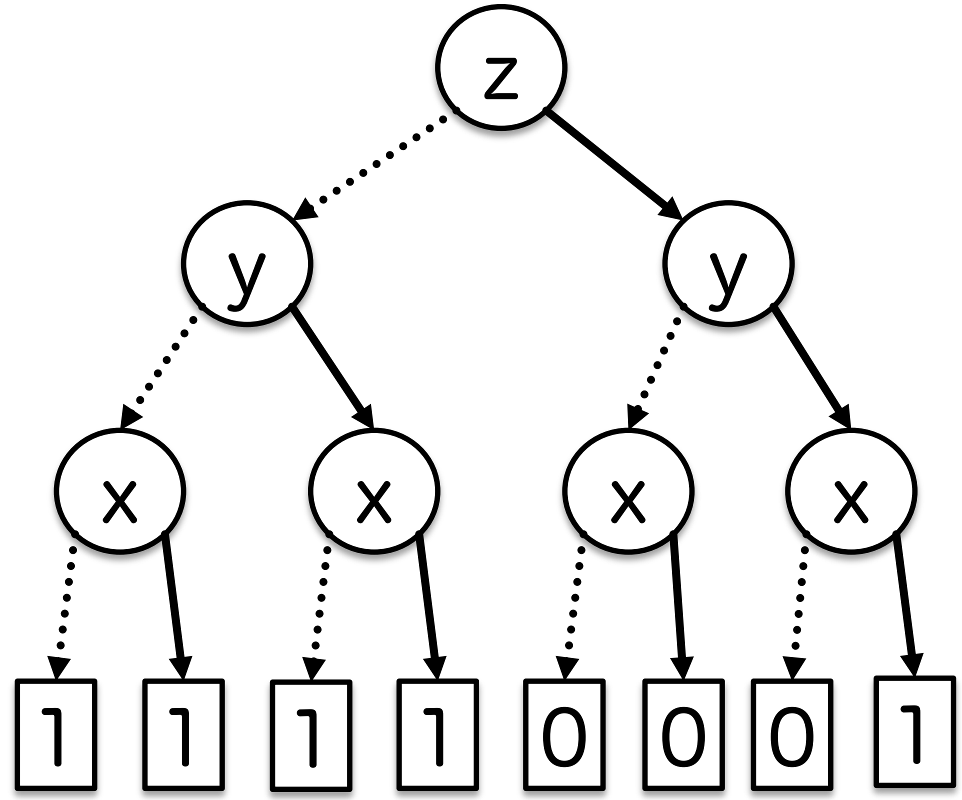


Variable Ordering is important

We change the order x, y, z to z, y, x.

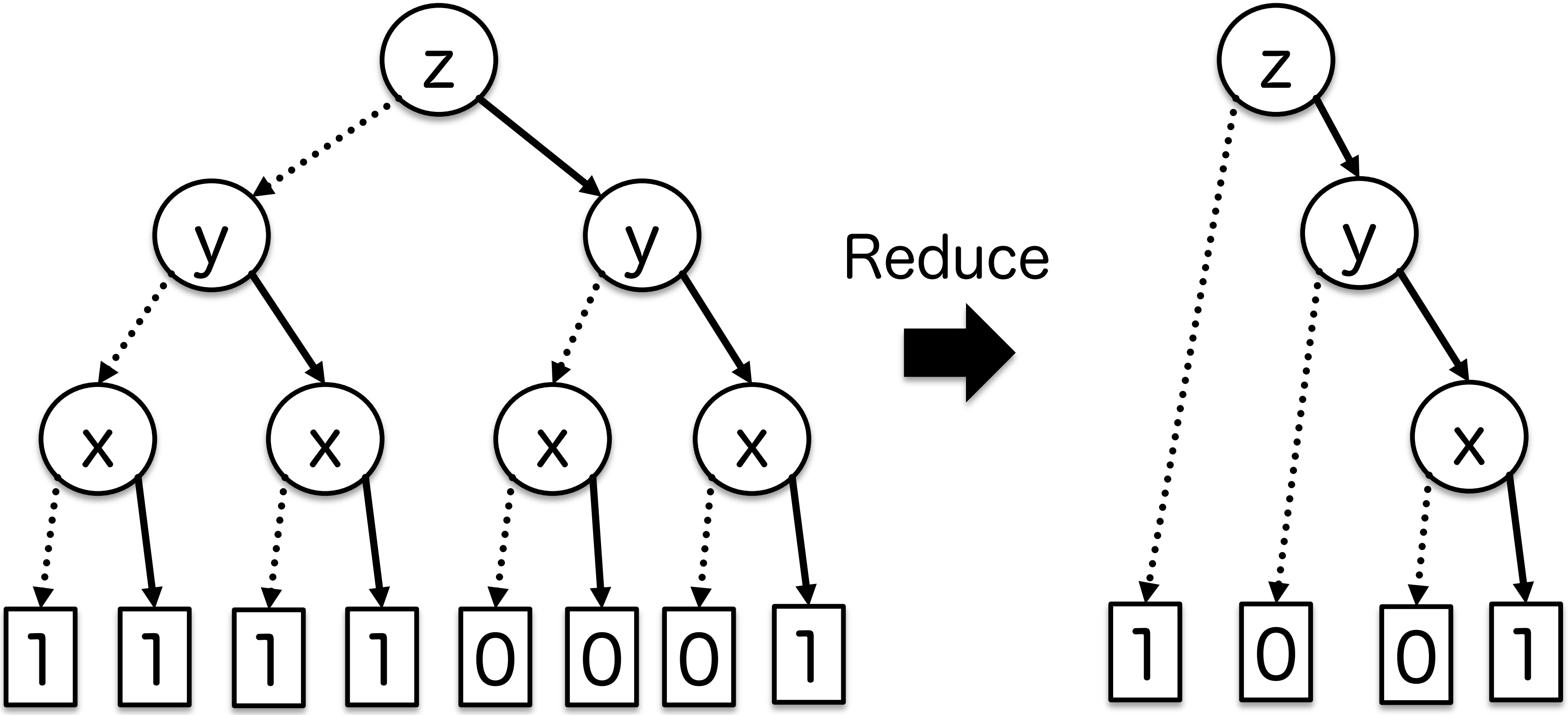
The decision tree changes to the following.

x	y	z	f
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



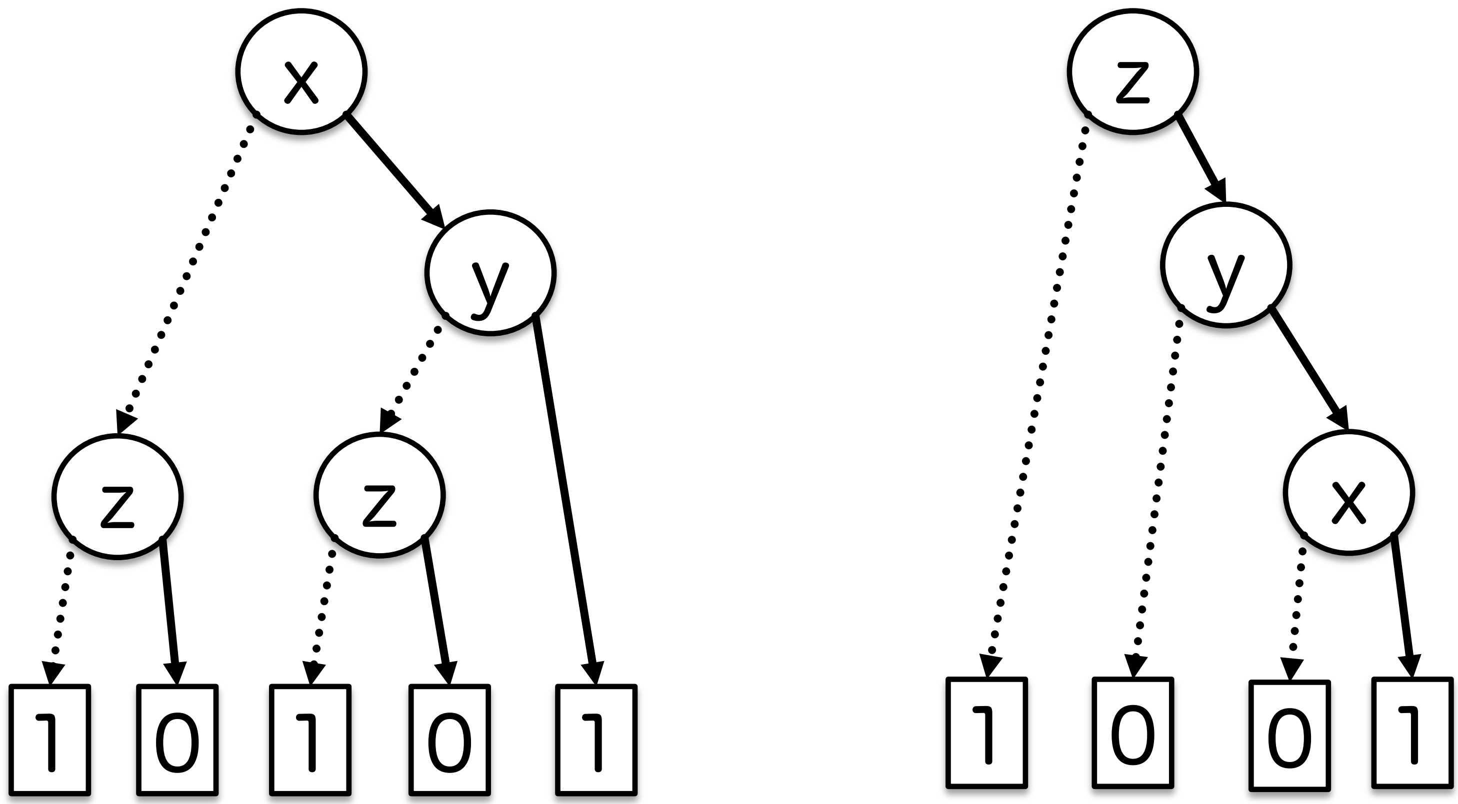
Variable Ordering is important

After reducing, the number of nodes labeled with variable is 3.



Variable Ordering is important

The following two binary decision trees represent the same Boolean function, but the size is different !



Variable Ordering is important

Decision Trees strongly depends on Variable Ordering.

- It is important to construct a decision tree with small size as possible.
  - ✓ Saving the memory
- However, it is very difficult.
  - ✓ How to compute the optimal variable ordering ?

Exercise 2

The following truth table is representing a Boolean function f. Construct a decision tree of f with a small size as possible.

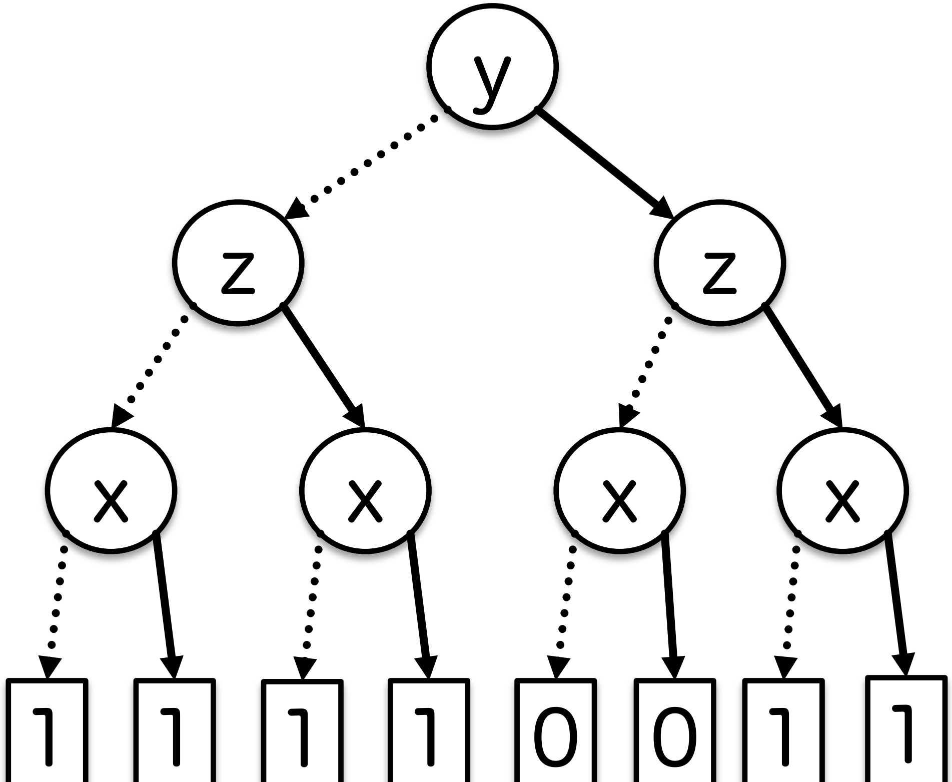
x	y	z	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Exercise 2 (Answer)

Try to construct decision trees for all orders of variables.

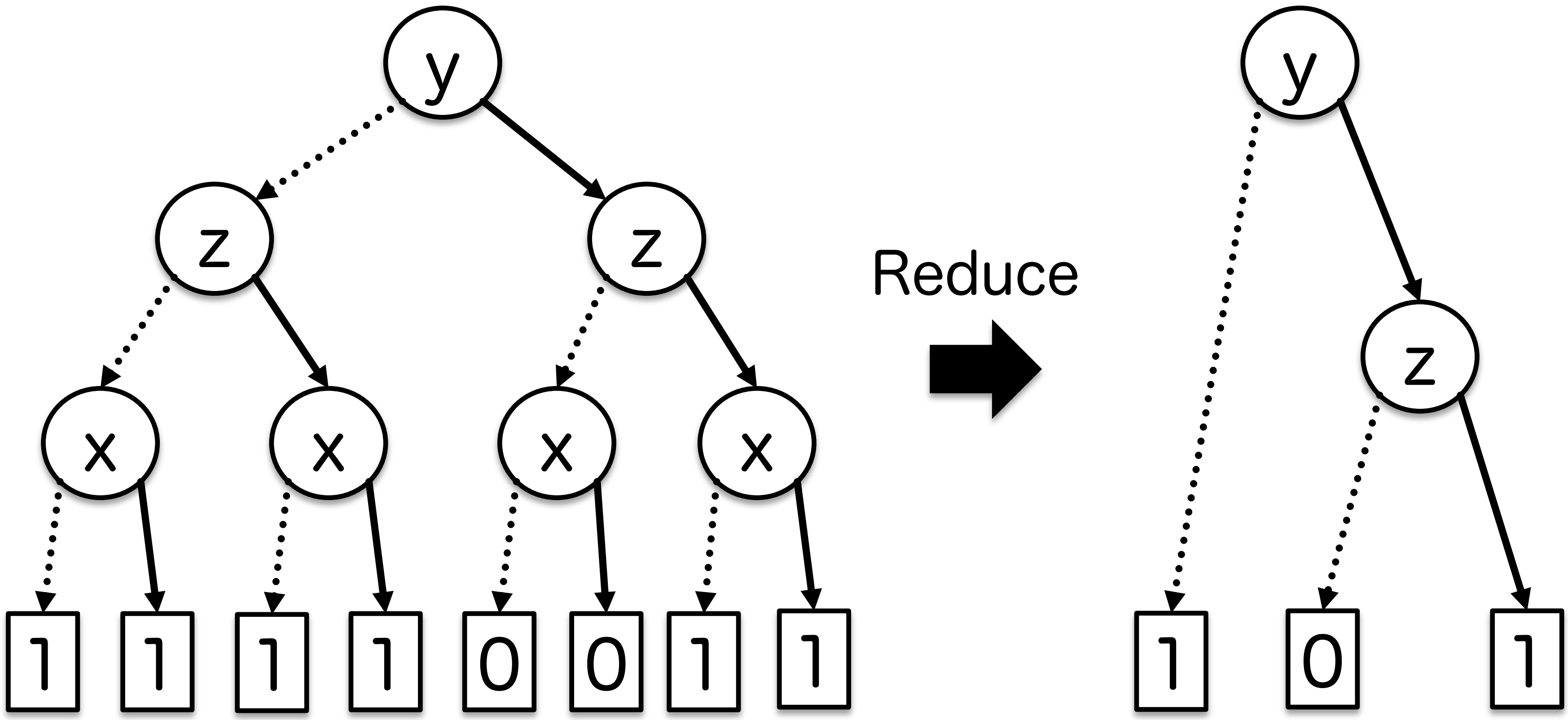
The order y, z, x makes the smallest decision tree.

x	y	z	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1



Exercise 2 (Answer)

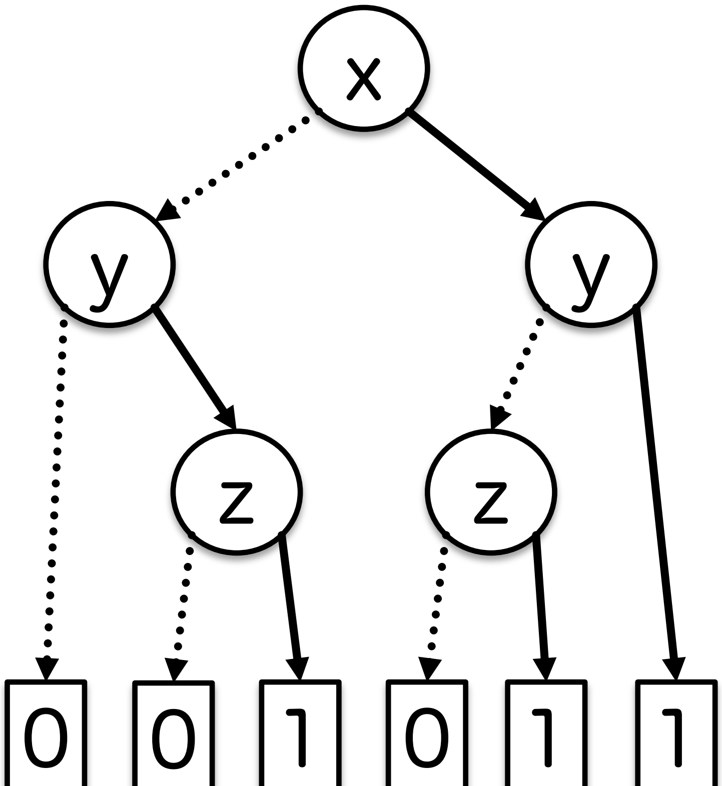
Reducing the constructed decision tree.



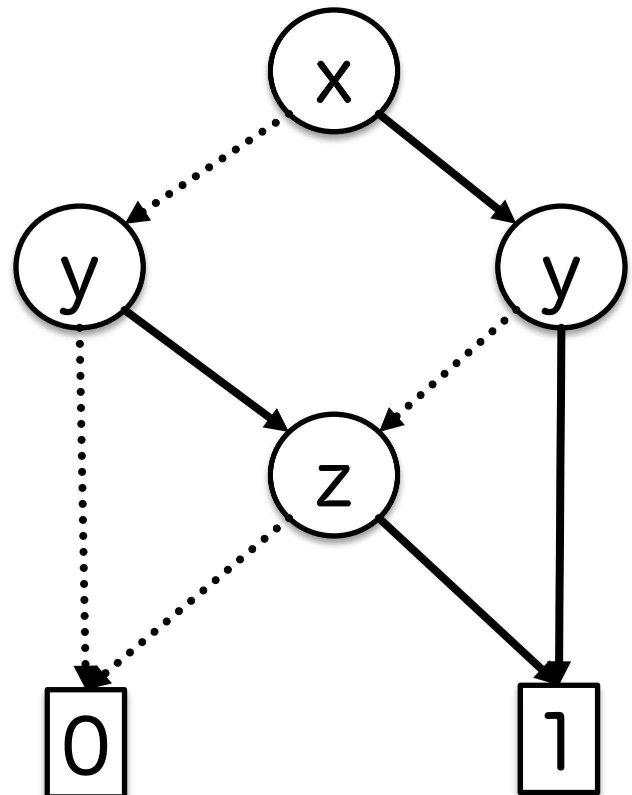
Binary Decision Diagram : BDD (1)

Binary Decision Diagram: BDD

A Binary Decision Diagram is represented by a directed acyclic graph.



Binary Decision Tree



BDD

Binary Decision Diagram: BDD

BDD satisfies the following conditions.

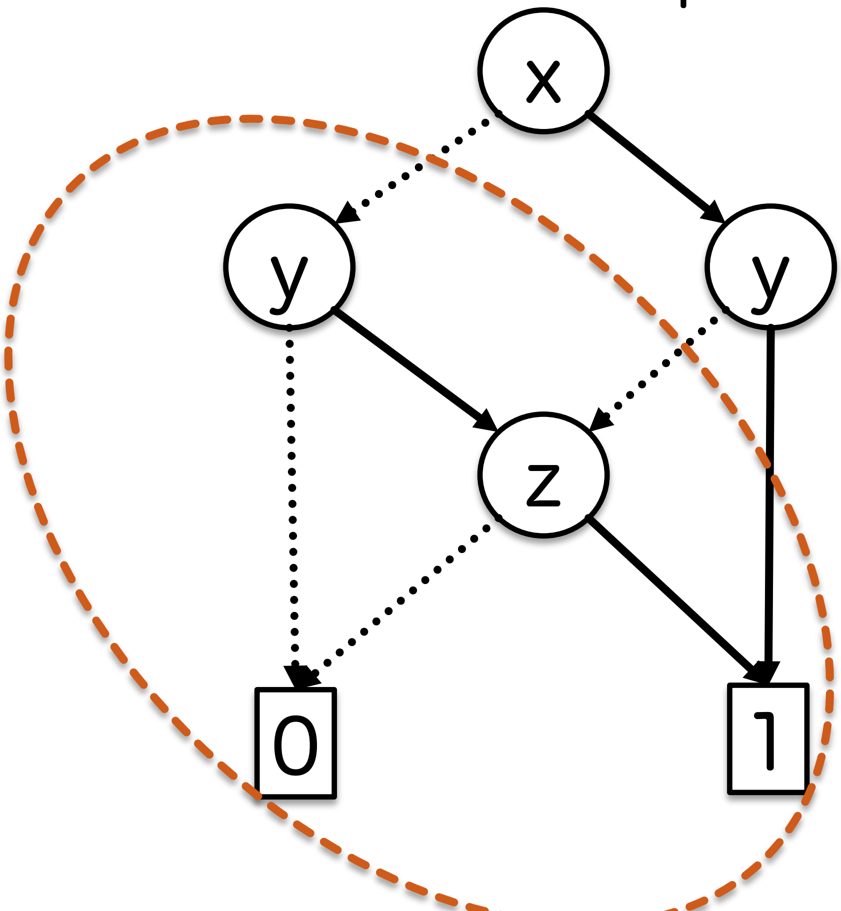
- consists of constant nodes（0,1）and variable nodes.
- directed acyclic graph
- no edge from constant node
- one 0-edge and one 1-edge from variable node
- one of variable nodes is the root node with degree 0.
- the size of BDD is the number of nodes

Characterizations of BDDs

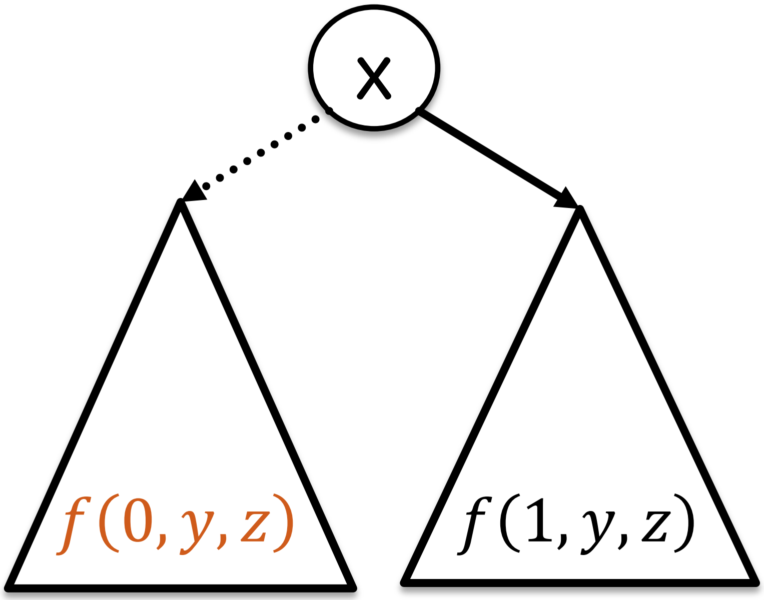
- Many practical Boolean functions can be represented small BDDs.
- For a variable ordering, BDD has unique representation.
- There exist efficient operations for BDD [Bryant 1986]
- A subgraph of a BDD represents the subfunction of the Boolean function represented by the BDD.
- Recently, BDDs have many applications.

Characterizations of BDDs

A subgraph of a BDD represents the subfunction of the Boolean function represented by the BDD.



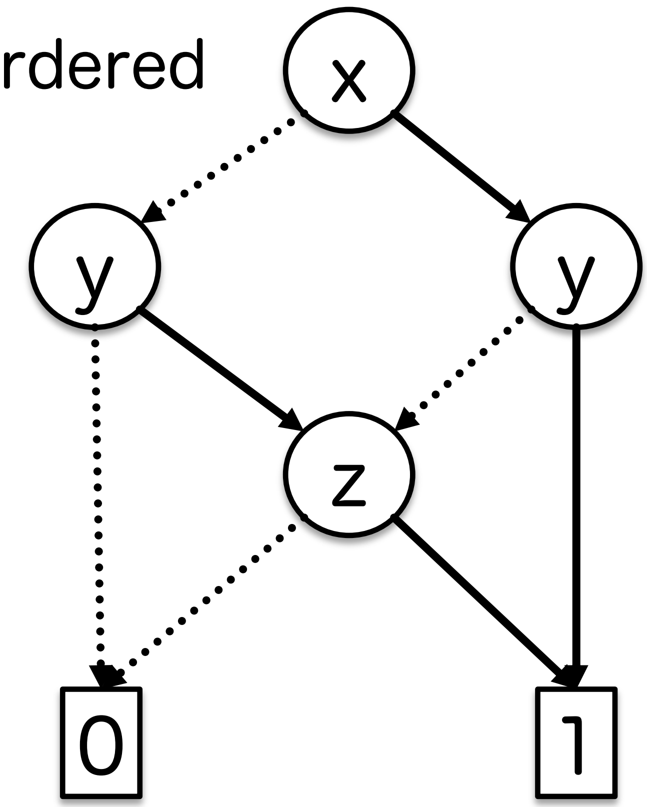
$$\bar{x}(yz) \vee x(\bar{y}z \vee y)$$
$$= \bar{x}f(0, y, z) \vee xf(1, y, z)$$



The definition of OBDDs（Ordered BDD）

The variable appears according to the total order.

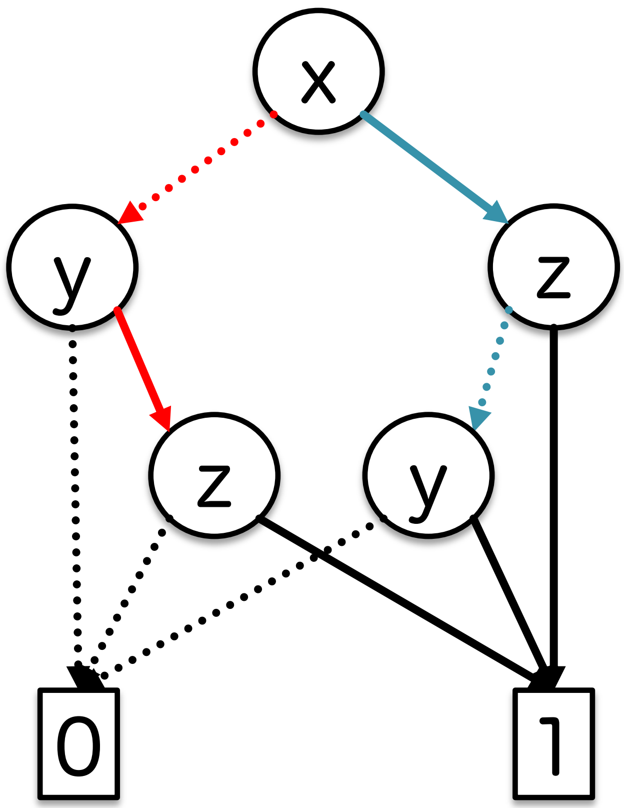
Ordered



In any path from the root node to a leave, we check the value of x, y, z in that order.

The definition of OBDDs（Ordered BDD)

This example is not OBDD because the variable's order in the red path is different from that in the blue path.



The size of OBDDs can be larger than that of non-OBDDs, but OBDDs are often easier to handle.

In this class, we handle OBDDs

The definition of ROBDDs（Reduced Ordered BDD）

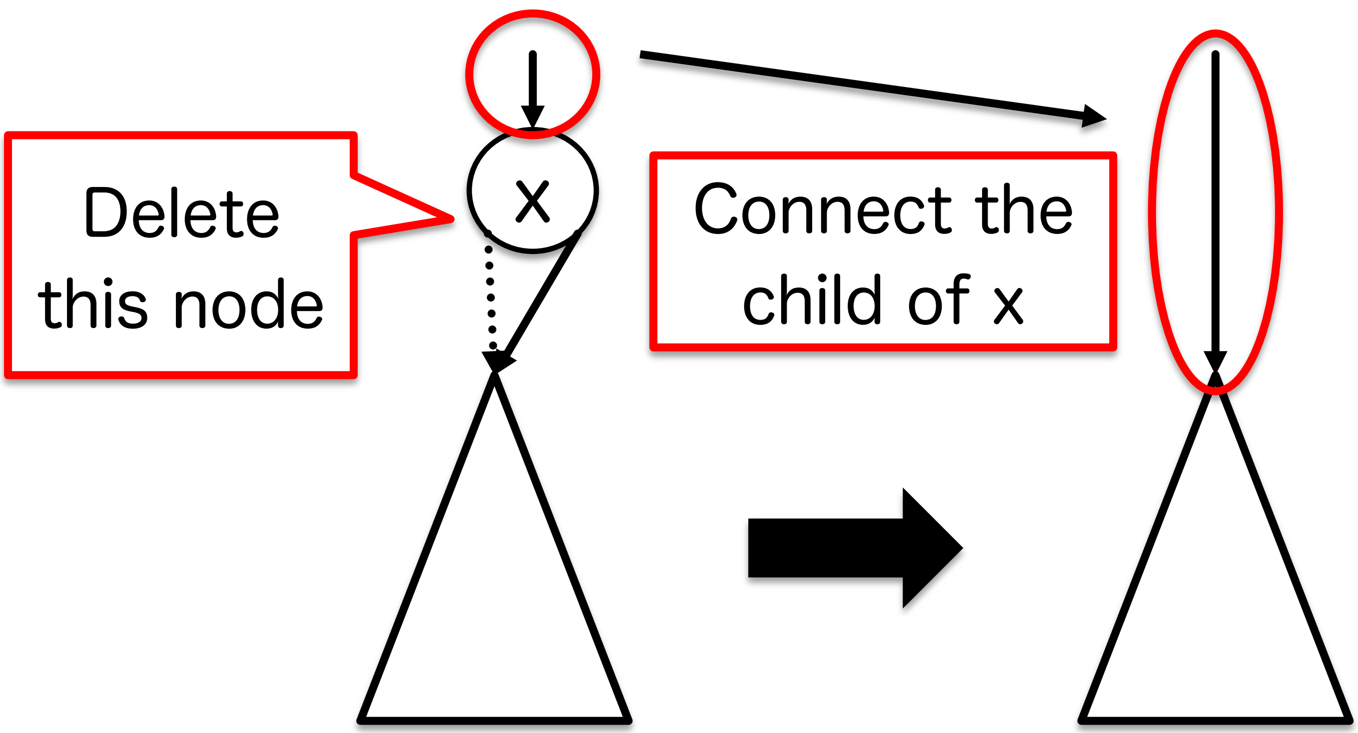
A ROBDD is obtained by applying the following two reducing rules to a BDD

- Delete redundant nodes
- Merge equivalent nodes

In this class, BDDs means ROBDDs

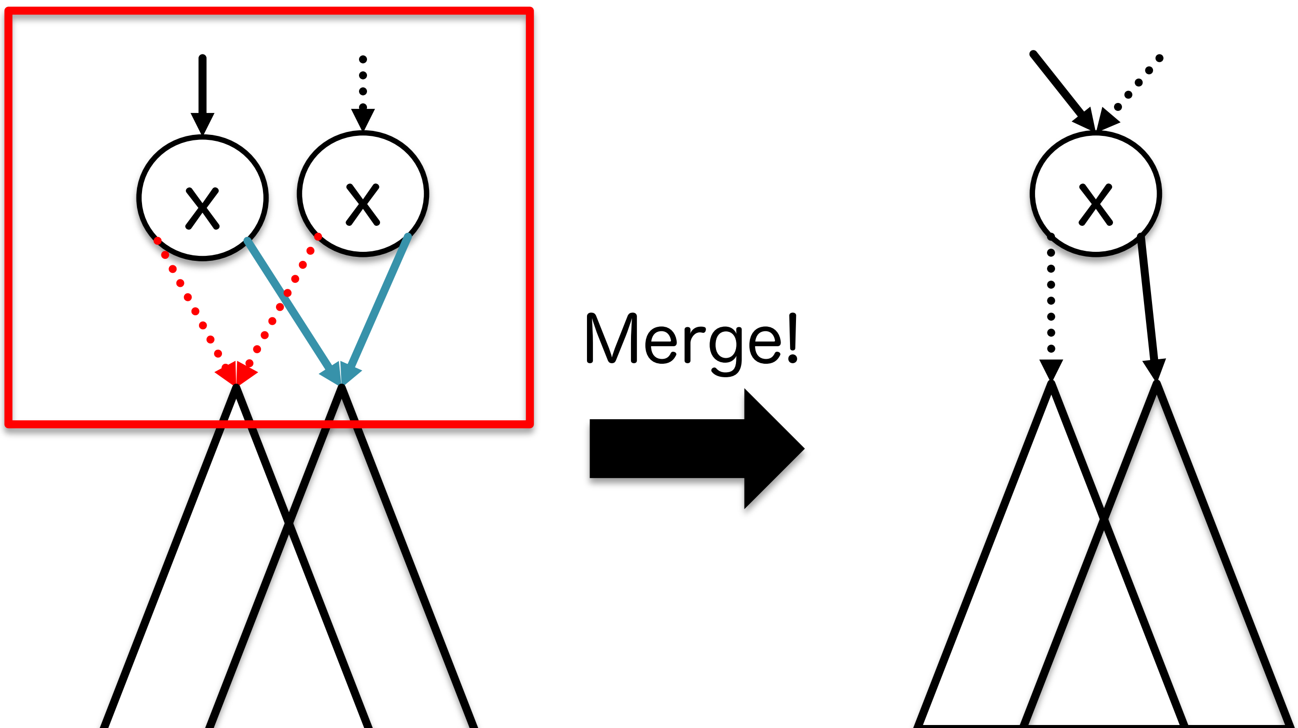
Delete redundant nodes

In the following, the node labeled with x is redundant because 0-edge and 1-edge from the node is connected to the same node. In this case, we can delete the node x.



Merge equivalent node

In the following, 0-edges of two nodes labeled with the same variable x are connected to the same child and 1-edges are also connected to the same childe. In this case, we can merge these nodes.



Exercise 3

Represent the following Boolean functions by BDDs

(1) $x_1 \land x_2 \land x_3 \land x_4$

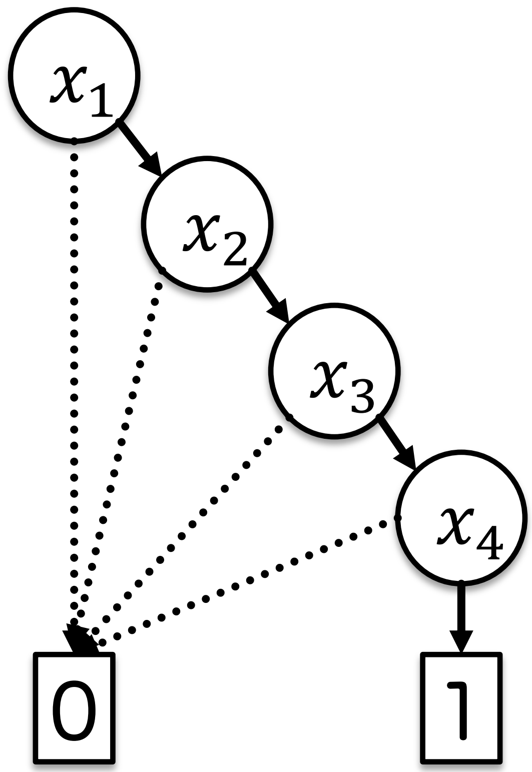
(2) $x_1 \lor x_2 \lor x_3 \lor x_4$

(3) $(x_1 \lor x_2) \land x_3$

(4) $x_1 \oplus x_2 \oplus x_3 \oplus x_4$

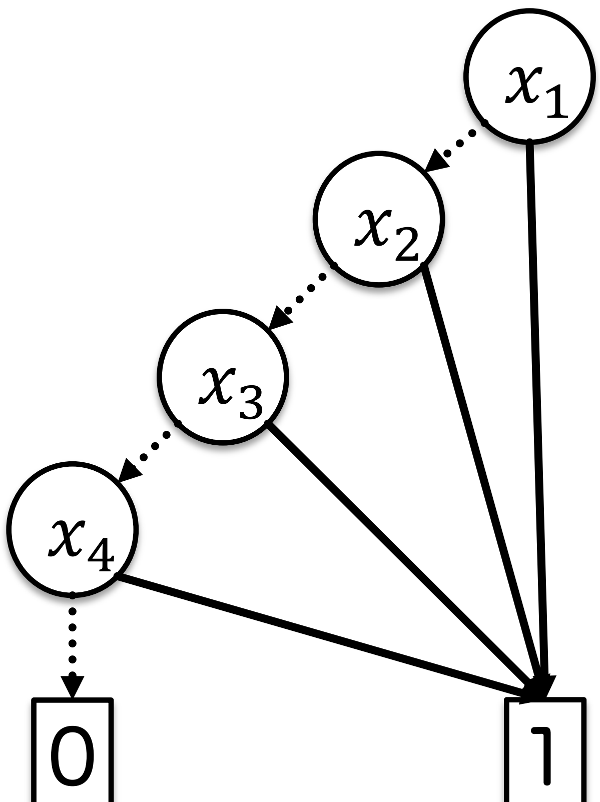
Exercise 3 (Answer)

(1) $x_1 \wedge x_2 \wedge x_3 \wedge x_4$



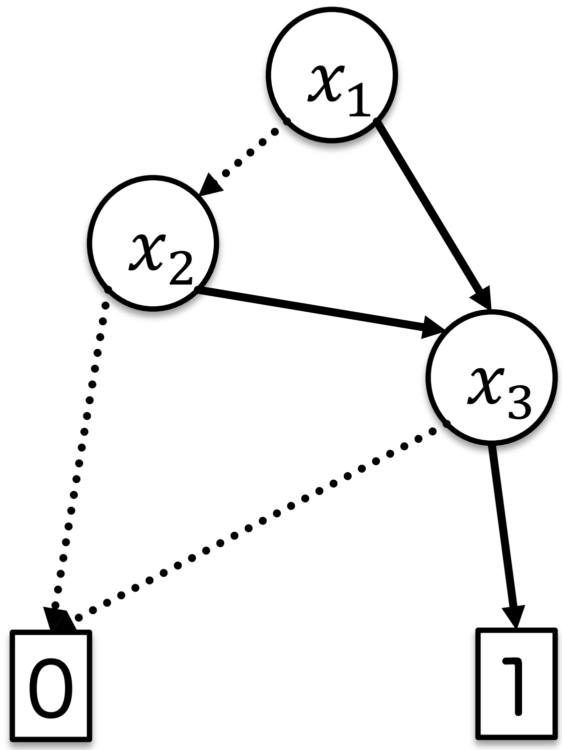
Exercise 3 (Answer)

(2) $x_1 \lor x_2 \lor x_3 \lor x_4$



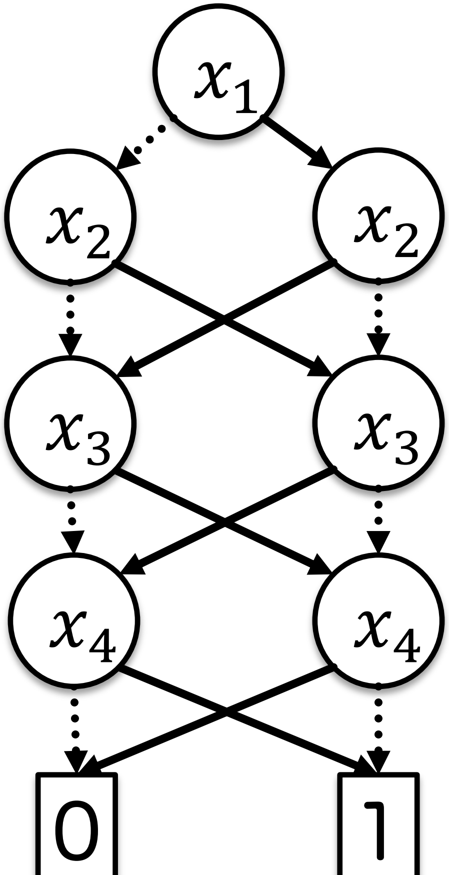
Exercise 3 (Answer)

(3) $(x_1 \lor x_2) \land x_3$



Exercise 3 (Answer)

(4) $x_1 \oplus x_2 \oplus x_3 \oplus x_4$



Binary Decision Diagram : BDD (2)

Variable Ordering is also important for BDD

As with decision trees, the shape of the BDD depens on the order in which the variables are read.

Can any Boolean function be represented by a small BDD?

- There exists a small BDD for any variable ordering → Happy!
- There exists no small BDD for any variable ordering → Bad!

Variable Ordering is also important for BDD

- In some cases, there exists a variable ordering for a small BDD
  - ✓ Computing a good variable ordering is important !
  - ✓ However, computing an optimal variable ordering is NP-hard [Tani, et al 1993]

A small BDD in any variable ordering

The following Boolean function can be represented by a small BDD in any variable ordering.

Boolean function	The size of BDD
AND	$O(n)$
OR	$O(n)$
EXOR	$O(n)$
Majority	$O(n^2)$
Symmetric	$O(n^2)$

※ Symmetric : The output value only depends on the number of one's of input variables

No small BDD in any variable ordering

The following functions have no small BDD in any variable ordering.

- Arithmetic Multiplication [Bryant 1991] : $\Omega(2^{n/5})$
- Arithmetic Division [Horiyama, Yajima 1998] : $\Omega(2^{n/8})$
- Random Functions

An example of function that has a small BDD in some variable ordering.

Example： $x_1 y_1 \vee x_2 y_2$

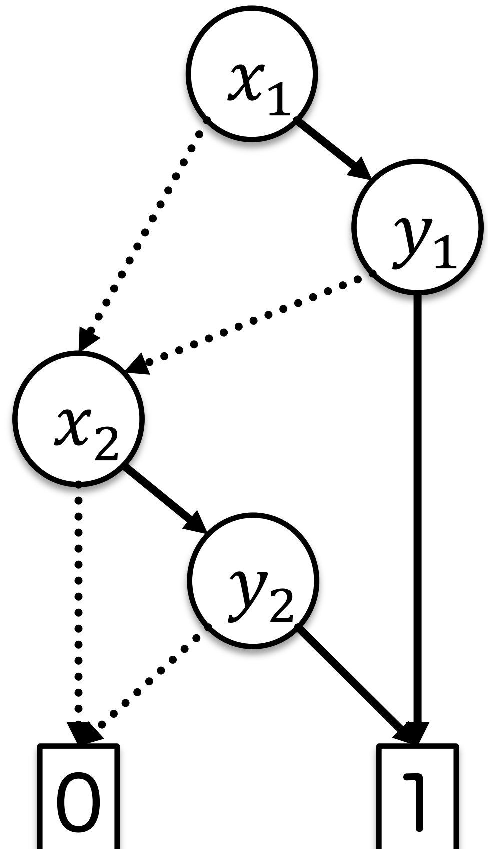
Consider two variable ordering

- x_1, y_1, x_2, y_2
- x_1, x_2, y_1, y_2

An example of function that has a small BDD in some variable ordering.

Example： $x_1 y_1 \vee x_2 y_2$

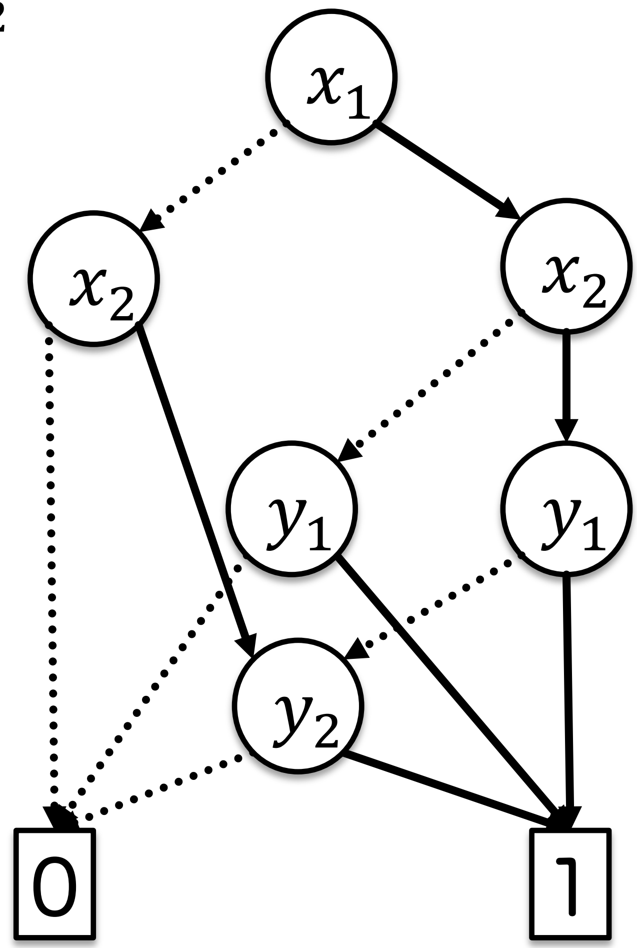
- x_1, y_1, x_2, y_2



An example of function that has a small BDD in some variable ordering.

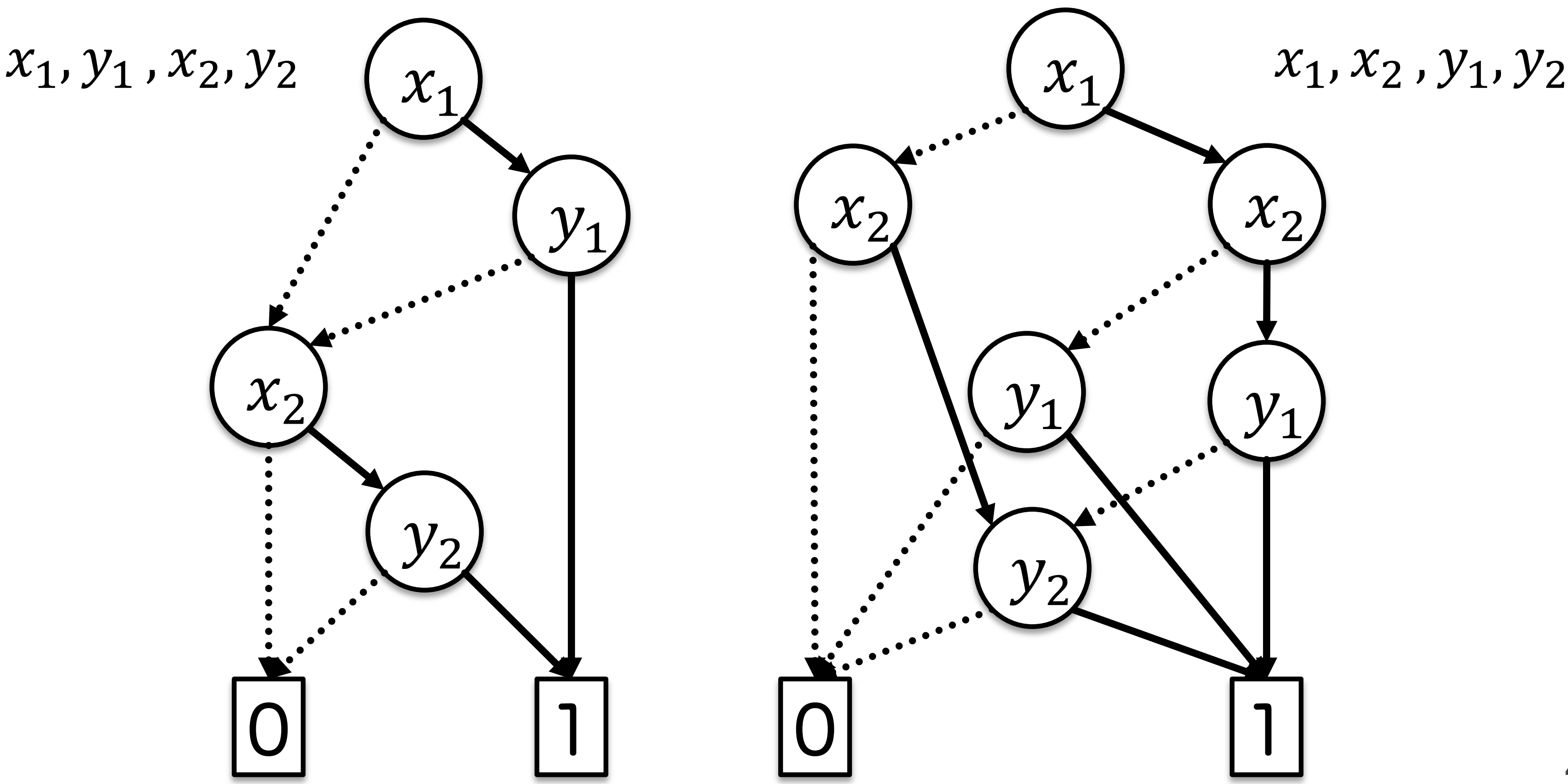
Example： $x_1 y_1 \vee x_2 y_2$

➢ x_1, x_2, y_1, y_2



An example of function that has a small BDD in some variable ordering.

Example： $x_1 y_1 \vee x_2 y_2$



An example of function that has a small BDD in some variable ordering.

Example： $x_1 y_1 \vee x_2 y_2 \vee \cdots \vee x_n y_n$

Consider two variable ordering

- $x_1,\ y_1,\ x_2,\ y_2, \ldots, x_n,\ y_n \rightarrow$ size of BDD：O(n)
- $x_1,\ x_2, \ldots, x_n,\ y_1,\ y_2 \ldots, y_n \rightarrow$ size of BDD：$O(2^n)$

You consider why this large difference happens !

Exercise 4

Consider two 2-bits binary number $a: a_1 a_0$ and $b: b_1 b_0$

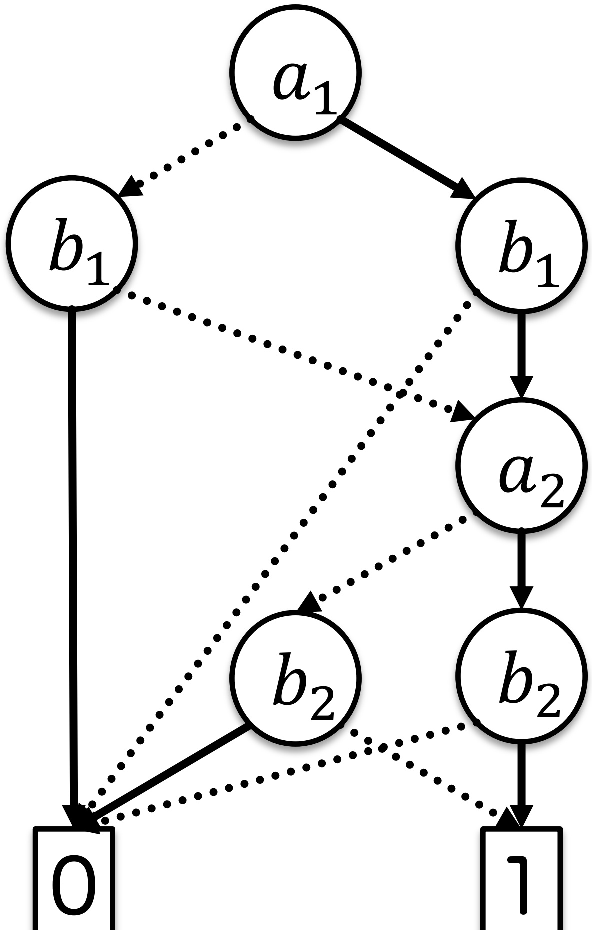
$f(a_1, a_0, b_1, b_0)$ outputs 1 when $a_1 = b_1$ and $a_0 = b_0$,

otherwise, outputs 0.

1. Show the variable ordering such that the size of BDD for f is the smallest.
2. Show the variable ordering such that the size of BDD for f is the largest.

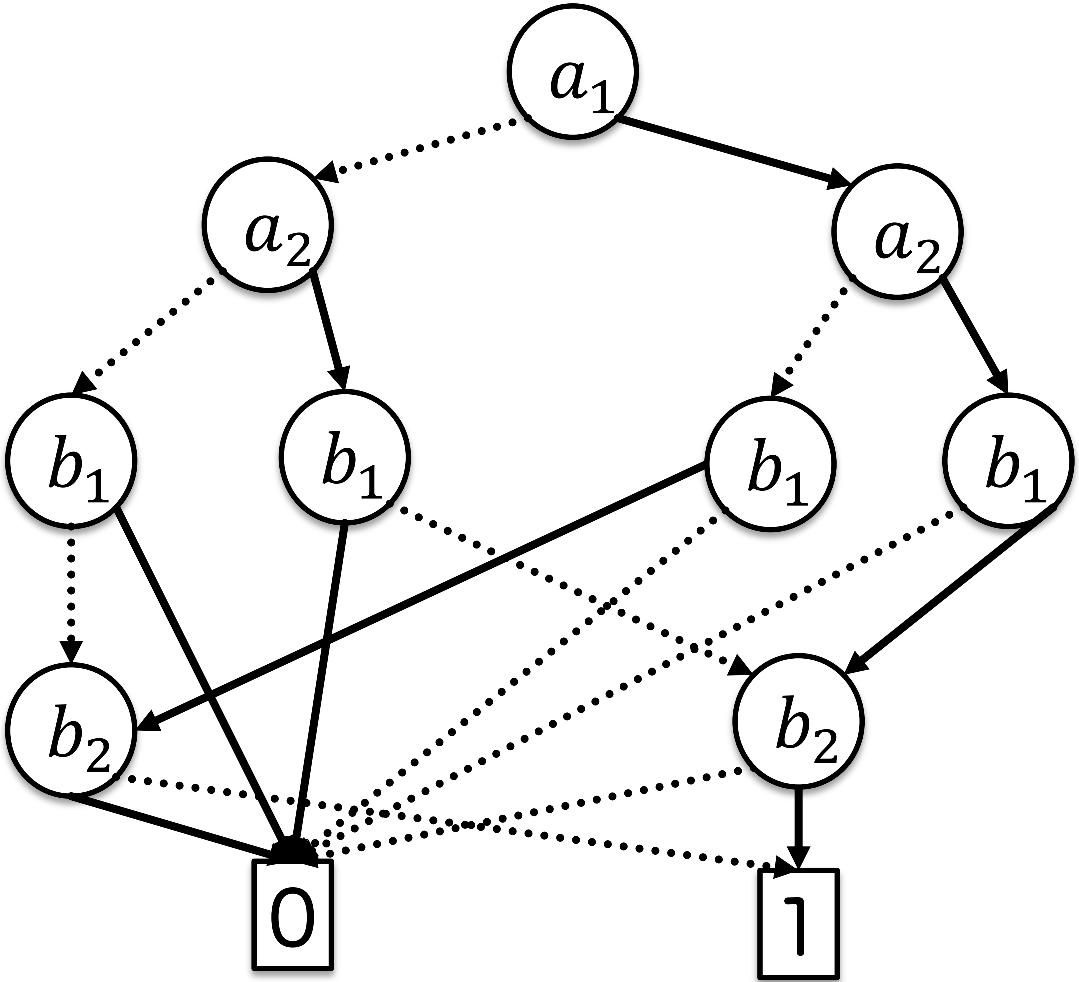
Exercise 4（Answer）

The variable ordering such that the size is smallest.



Exercise 4（Answer）

The variable ordering such that the size is largest.



Summary

Introduce Binary Decision Trees and Binary Decision Diagrams

- Variable ordering may affect the size of BDDs.
- It is difficult to computing the variable ordering that minimizes the size of BDD.