

# 3-1 Dynamic Programming

## (Part II: “Theory”)

Hengfeng Wei

hfwei@nju.edu.cn

September 27, 2018



## Definition (Optimal Substructure)

A problem exhibits ***optimal substructure*** if an optimal solution to the problem contains within it optimal solutions to subproblems.

## Definition (Optimal Substructure)

A problem exhibits ***optimal substructure*** if an optimal solution to the problem contains within it optimal solutions to subproblems.

*Who exhibits . . . ?*

## Definition (Optimal Substructure)

A problem exhibits ***optimal substructure*** if an optimal solution to the problem contains within it optimal solutions to subproblems.

*Who* exhibits . . . ?

*How to prove “YES”?*

## Definition (Optimal Substructure)

A problem exhibits ***optimal substructure*** if an optimal solution to the problem contains within it optimal solutions to subproblems.

*Who* exhibits ... ?

*How to prove "YES"?*

*How to use "Cut-and-Paste"?*

## Definition (Optimal Substructure)

A problem exhibits ***optimal substructure*** if an optimal solution to the problem contains within it optimal solutions to subproblems.

*Who* exhibits ... ?

*How to prove "YES"?*

*How to use "Cut-and-Paste"?*

*How to show "NO"?*

## Definition (Optimal Substructure)

A problem exhibits ***optimal substructure*** if an optimal solution to the problem contains within it optimal solutions to subproblems.

*Who* exhibits . . . ?

*How to prove “YES”?*

*How to use “Cut-and-Paste”?*

*How to show “NO”?*

**Relative to Subproblems**

# Rod Cutting



## Optimal Substructure of Rod-Cutting (Problem 15.3-5)

*Limit*  $l_i$  : # of pieces of length  $i$ ,  $1 \leq i \leq n$

## Optimal Substructure of Rod-Cutting (Problem 15.3-5)

*Limit*  $l_i$  : # of pieces of length  $i$ ,  $1 \leq i \leq n$

$$n = 4$$

length $i$	1	2	3	4
price $p_i$	1	1	1	1

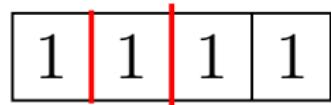
length $i$	1	2	3	4
limit $l_i$	2	1	1	1

## Optimal Substructure of Rod-Cutting (Problem 15.3-5)

*Limit*  $l_i$  : # of pieces of length  $i$ ,  $1 \leq i \leq n$

$$n = 4$$

length $i$	1	2	3	4
price $p_i$	1	1	1	1



$$R(4) = 3$$

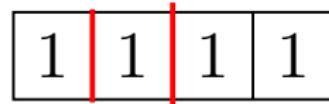
length $i$	1	2	3	4
limit $l_i$	2	1	1	1

## Optimal Substructure of Rod-Cutting (Problem 15.3-5)

*Limit*  $l_i$  : # of pieces of length  $i$ ,  $1 \leq i \leq n$

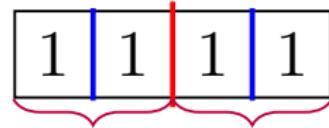
$$n = 4$$

length $i$	1	2	3	4
price $p_i$	1	1	1	1

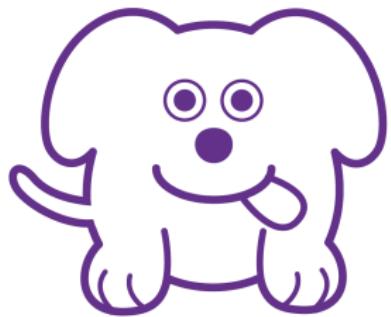


$$R(4) = 3$$

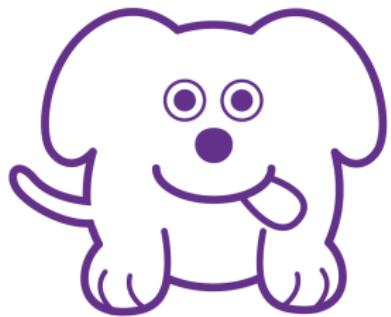
length $i$	1	2	3	4
limit $l_i$	2	1	1	1



$$R(2) = 2 \quad R(2) = 2$$

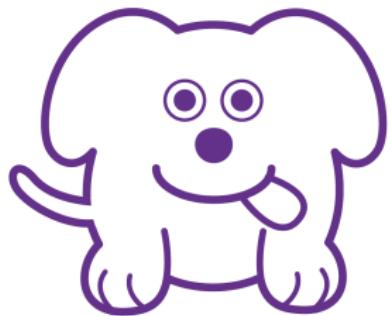


Well done



# Well done

*“Show that the optimal-substructure property described in Section 15.1 no longer holds.”*



# Well done

*“Show that the optimal-substructure property **described in Section 15.1** no longer holds.”*

$R(i, L)$  : max revenue obtainable by cutting up a rod of length  $i$

*with the length limit array  $L$*

$R(i, L)$  : max revenue obtainable by cutting up a rod of length  $i$

with the length limit array  $L$

Where is the leftmost cut?

$R(i, L)$  : max revenue obtainable by cutting up a rod of length  $i$

with the length limit array  $L$

Where is the leftmost cut?

$$R(i, L) = \max_{\substack{1 \leq j \leq i \\ L_j \geq 1}} \left( p_j + R(i - j, L[j \mapsto L_j - 1]) \right)$$

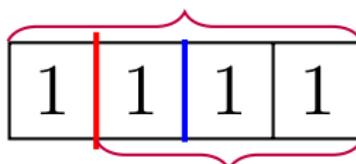
$R(i, L)$  : max revenue obtainable by cutting up a rod of length  $i$

with the length limit array  $L$

Where is the leftmost cut?

$$R(i, L) = \max_{\substack{1 \leq j \leq i \\ L_j \geq 1}} \left( p_j + R(i - j, L[j \mapsto L_j - 1]) \right)$$

$$R(4, [2, 1, 1, 1]) = 3$$



$$R(3, [1, 1, 1, 1]) = 2$$

## Currency Exchange (Problem 15.3-6)

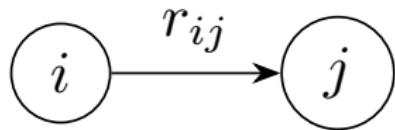
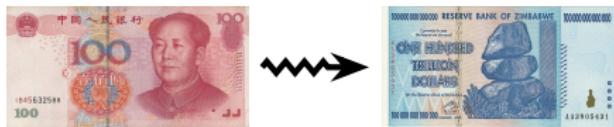


$1, 2, \dots, n$  currencies

## Currency Exchange (Problem 15.3-6)



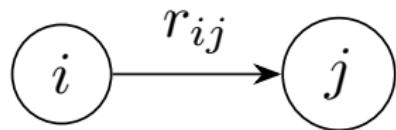
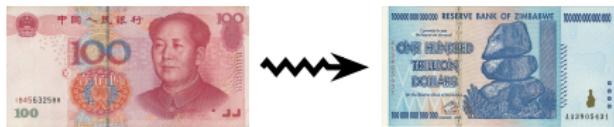
$1, 2, \dots, n$  currencies



## Currency Exchange (Problem 15.3-6)



$1, 2, \dots, n$  currencies

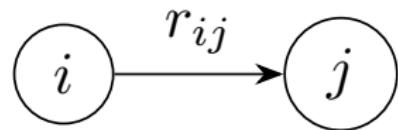
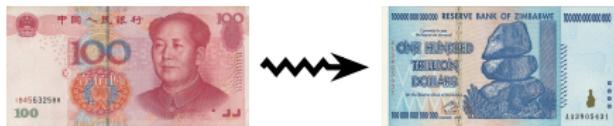


$c_k$  : Commission charged for  $k$  trades

## Currency Exchange (Problem 15.3-6)

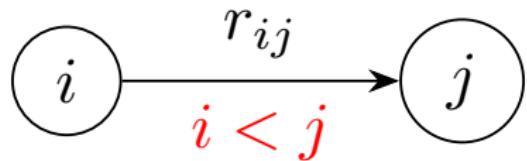


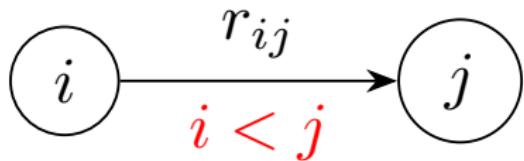
$1, 2, \dots, n$  currencies



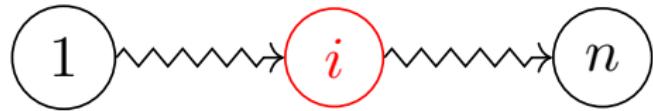
$c_k$  : Commission charged for  $k$  trades

$$c_k = 0$$





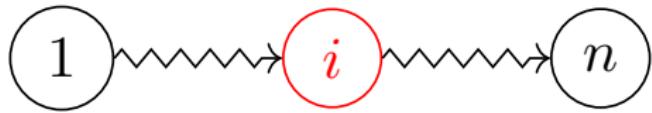
An *optimal* sequence of trades from 1 to  $n$  through  $i$ :







An *optimal* sequence of trades from 1 to  $n$  through  $i$ :

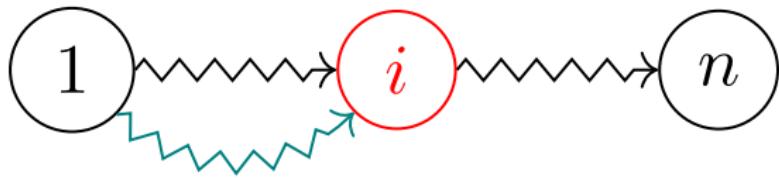




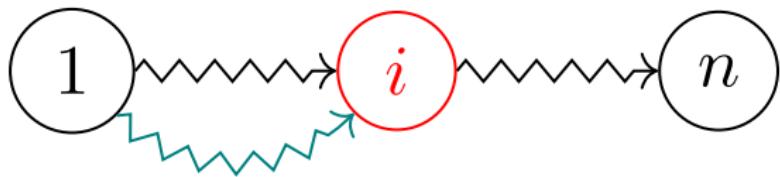
An *optimal* sequence of trades from 1 to  $n$  through  $i$ :



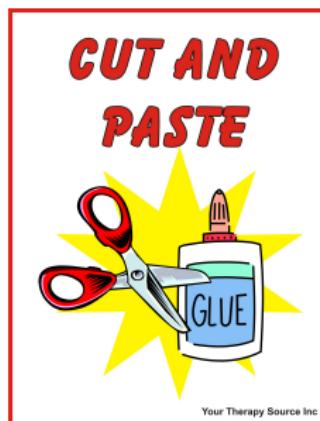
## *By Contradiction.*

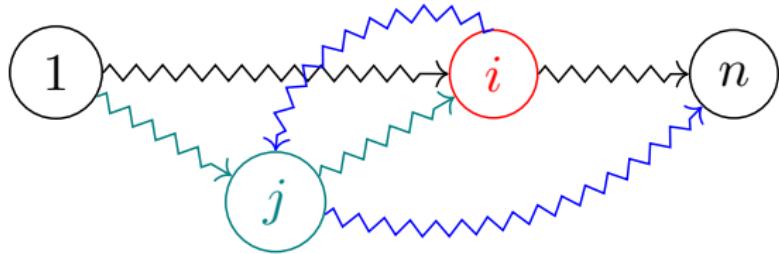


CASE I :  $s_{1 \sim i} \cap s_{i \sim n} = \emptyset$



CASE I :  $s_{1 \sim i} \cap s_{i \sim n} = \emptyset$





CASE II :  $j \in s_{1 \rightsquigarrow i} \cap s_{i \rightsquigarrow n}$

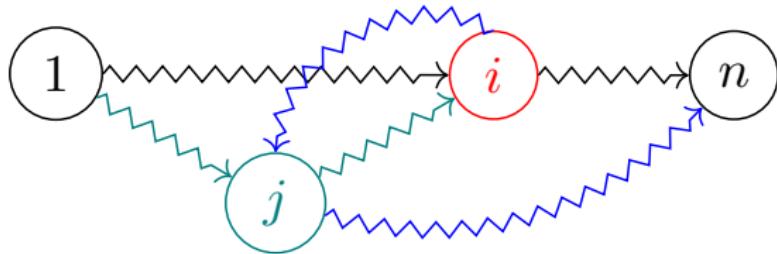
$$\begin{aligned}
 & 1 \rightsquigarrow j \rightsquigarrow n \\
 \geq & 1 \rightsquigarrow j \rightsquigarrow i \rightsquigarrow j \rightsquigarrow n \\
 = & 1 \rightsquigarrow j \rightsquigarrow i \rightsquigarrow n \\
 > & 1 \rightsquigarrow i \rightsquigarrow n
 \end{aligned}$$

## Longest Path Problem

To find a *simple* path of maximum length from  $s$  to  $t$  in a graph.

## Longest Path Problem

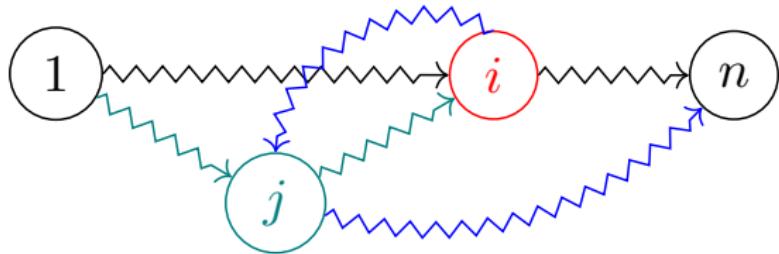
To find a *simple* path of maximum length from  $s$  to  $t$  in a graph.



$$\begin{aligned}1 &\rightsquigarrow j \rightsquigarrow n \\&\geq 1 \rightsquigarrow j \rightsquigarrow i \rightsquigarrow j \rightsquigarrow n \\&= 1 \rightsquigarrow j \rightsquigarrow i \rightsquigarrow n \\&> 1 \rightsquigarrow i \rightsquigarrow n\end{aligned}$$

## Longest Path Problem

To find a *simple* path of maximum length from  $s$  to  $t$  in a graph.



$$\begin{aligned} & 1 \rightsquigarrow j \rightsquigarrow n \\ \geq & 1 \rightsquigarrow j \rightsquigarrow i \rightsquigarrow j \rightsquigarrow n \\ = & 1 \rightsquigarrow j \rightsquigarrow i \rightsquigarrow n \\ > & 1 \rightsquigarrow i \rightsquigarrow n \end{aligned}$$

*Does the longest path problem really  
have no optimal substructure?*

*Does the longest path problem really  
have no optimal substructure?*



**WAIT WAIT...  
DON'T TELL ME!®**

FROM NPR® & WBEZ® CHICAGO

$L(s, t, A)$  : the longest path from  $s$  to  $t$

through **exactly** the intermediate vertices in  $I$

$L(s, t, A)$  : the longest path from  $s$  to  $t$

through **exactly** the intermediate vertices in  $I$

$$L(s, t) =$$

$L(s, t, A)$  : the longest path from  $s$  to  $t$

through **exactly** the intermediate vertices in  $I$

$$L(s, t) = \max_{I \subseteq V} L(s, t, I)$$

$L(s, t, A)$  : the longest path from  $s$  to  $t$

through **exactly** the intermediate vertices in  $I$

$$L(s, t) = \max_{I \subseteq V} L(s, t, I)$$

What is the next vertex from  $s$ ?

$L(s, t, A)$  : the longest path from  $s$  to  $t$

through **exactly** the intermediate vertices in  $I$

$$L(s, t) = \max_{I \subseteq V} L(s, t, I)$$

What is the next vertex from  $s$ ?

$$L(s, t, I) = 1 + \max_{\substack{(s, x) \in E \\ x \in I}} l(x, t, I \setminus \{x\})$$

$L(s, t, A)$  : the longest path from  $s$  to  $t$

through **exactly** the intermediate vertices in  $I$

$$L(s, t) = \max_{I \subseteq V} L(s, t, I)$$

What is the next vertex from  $s$ ?

$$L(s, t, I) = 1 + \max_{\substack{(s, x) \in E \\ x \in I}} l(x, t, I \setminus \{x\})$$

$$L(t, t, I) = 0, \quad L(s, t, \emptyset) = \begin{cases} 1, & \text{if } (s, t) \in E \\ 0, & \text{otherwise} \end{cases}$$

$$L(s, t, I) = 1 + \max_{\substack{(s, x) \in E \\ x \in I}} l(x, t, I \setminus \{x\})$$

$$L(s, t, I) = 1 + \max_{\substack{(s, x) \in E \\ x \in I}} l(x, t, I \setminus \{x\})$$

$$O(n2^n)$$

$$L(s, t, I) = 1 + \max_{\substack{(s, x) \in E \\ x \in I}} l(x, t, I \setminus \{x\})$$

$$O(n2^n)$$

*Cycle  $\implies$  No Order on Vertices*

$$L(s, t, I) = 1 + \max_{\substack{(s, x) \in E \\ x \in I}} l(x, t, I \setminus \{x\})$$

$$O(n2^n)$$

*Cycle  $\implies$  No Order on Vertices*

*DP does **not** necessarily lead to efficient (polynomial) algorithms.*

$$L(s, t, I) = 1 + \max_{\substack{(s, x) \in E \\ x \in I}} l(x, t, I \setminus \{x\})$$

$$O(n2^n)$$

*Cycle  $\implies$  No Order on Vertices*

*DP does not necessarily lead to efficient (polynomial) algorithms.*

*The (decision version of the) longest path problem is NP-hard!*

# The Change-Making Problem

Coins values:  $x_1, x_2, \dots, x_n$

Amount:  $v$

*Is it possible to make change for  $v$ ?*



# The Change-Making Problem

*Without repetition: 0/1*

# The Change-Making Problem

*Without repetition: 0/1*

$C[i, w]$  : Make change for  $w$  using only values of  $x_1 \dots x_i$ ?

$C[n, v]$

# The Change-Making Problem

*Without repetition: 0/1*

$C[i, w]$  : Make change for  $w$  using only values of  $x_1 \dots x_i$ ?

$C[n, v]$

*Using value  $x_i$  or not?*

$$C[i, w] = \underbrace{C[i - 1, w]}_{\notin} \vee \left( \underbrace{C[i - 1, w - x_i]}_{\in} \wedge w \geq x_i \right)$$

# The Change-Making Problem

*Without repetition: 0/1*

$C[i, w]$  : Make change for  $w$  using only values of  $x_1 \dots x_i$ ?

$C[n, v]$

*Using value  $x_i$  or not?*

$$C[i, w] = \underbrace{C[i - 1, w]}_{\notin} \vee \left( \underbrace{C[i - 1, w - x_i]}_{\in} \wedge w \geq x_i \right)$$

$$C[i, 0] = \text{true}, \forall i = 0 \dots n$$

$$C[0, w] = \text{false}, \text{ if } w > 0$$

$$C[0, 0] = \text{true}$$

## The 0/1 Knapsack Problem (Problem 16.2-2)

Values :  $v_i$

Weights :  $w_i$

Capacity :  $W$

*Taking as valuable a load as possible*

## The 0/1 Knapsack Problem (Problem 16.2-2)

Values :  $v_i$

Weights :  $w_i$

Capacity :  $W$

*Taking as valuable a load as possible*

$C[i, w]$  : Knapsack with capacity  $w$  using only items of values  $v_1 \dots v_i$

*Using value  $v_i$  or not?*

$$C[i, w] = \max \left( \underbrace{C[i - 1, w]}_{\notin}, \underbrace{\left( w \geq x_i \implies C[i - 1, w - w_i] + v_i \right)}_{\in} \right)$$

# The Change-Making Problem

*Unbounded repetition:  $\infty$*

# The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[i, w]$  : Make change for  $w$  using only values of  $x_1 \dots x_i$ ?

$C[n, v]$

# The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[i, w]$  : Make change for  $w$  using only values of  $x_1 \dots x_i$ ?

$C[n, v]$

*Using value  $x_i$  or not?*

# The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[i, w]$  : Make change for  $w$  using only values of  $x_1 \dots x_i$ ?

$C[n, v]$

*Using value  $x_i$  or not?*

$$C[i, w] = \underbrace{C[i - 1, w]}_{\notin} \vee \underbrace{(C[\textcolor{red}{i}, w - x_i] \wedge w \geq x_i)}_{\in}$$

# The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[i, w]$  : Make change for  $w$  using only values of  $x_1 \dots x_i$ ?

$C[n, v]$

*Using value  $x_i$  or not?*

$$C[i, w] = \underbrace{C[i - 1, w]}_{\notin} \vee \underbrace{(C[\textcolor{red}{i}, w - x_i] \wedge w \geq x_i)}_{\in}$$

$$C[i, 0] = \text{true}, \forall i = 0 \dots n$$

$$C[0, w] = \text{false}, \text{ if } w > 0$$

$$C[0, 0] = \text{true}$$

# The Change-Making Problem

*Unbounded repetition:  $\infty$*

## The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[w]$  : Possible to make change for  $w$ ?

$C[v]$

# The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[w]$  : Possible to make change for  $w$ ?

$C[v]$

**What is the first coin to use?**

## The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[w]$  : Possible to make change for  $w$ ?

$C[v]$

**What is the first coin to use?**

$$C[w] = \bigvee_{i: x_i \leq w} C[w - x_i]$$

## The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[w]$  : Possible to make change for  $w$ ?

$C[v]$

**What is the first coin to use?**

$$C[w] = \bigvee_{i: x_i \leq w} C[w - x_i]$$

$$C[0] = \text{true}$$

## The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[w]$  : Possible to make change for  $w$ ?

$C[v]$

**What is the first coin to use?**

$$C[w] = \bigvee_{i: x_i \leq w} C[w - x_i]$$

$$C[0] = \text{true}$$

$$O(nv)$$

## The Change-Making Problem

*Unbounded repetition:  $\infty$*

$C[i, w]$  vs.  $C[w]$

$$C[i, w] = C[i - 1, w] \vee (C[i, w - x_i] \wedge w \geq x_i)$$

$$C[w] = \bigvee_{i: x_i \leq w} C[w - x_i]$$

## The Change-Making Problem (Problem 16-1 (d))

*Unbounded repetition:  $\infty$*

*Using the fewest number of coins*

## The Change-Making Problem (Problem 16-1 (d))

*Unbounded repetition:  $\infty$*

*Using the fewest number of coins*

*Using value  $x_i$  or not?*

$$C[i, w] = \min \left( \underbrace{C[i - 1, w]}_{\notin}, \underbrace{\left( w \geq x_i \implies C[i, w - x_i] + 1 \right)}_{\in} \right)$$

## The Change-Making Problem (Problem 16-1 (d))

*Unbounded repetition:  $\infty$*

*Using the fewest number of coins*

**What is the first coin to use?**

$$C[w] = \min_{i: x_i \leq w} C[w - x_i] + 1$$

# The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

## The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[i, w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins of values of  $x_1 \dots x_i$

$C[n, v, k]$

# The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[i, w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins of values of  $x_1 \dots x_i$

$$C[n, v, k]$$

*Using value  $x_i$  or not?*

# The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[i, w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins of values of  $x_1 \dots x_i$

$$C[n, v, k]$$

*Using value  $x_i$  or not?*

$$C[i, w, l] = \underbrace{C[i-1, w, l]}_{\notin} \vee \underbrace{(C[i, w - x_i, l-1] \wedge w \geq x_i)}_{\in}$$

## The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[i, w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins of values of  $x_1 \dots x_i$

$$C[n, v, k]$$

*Using value  $x_i$  or not?*

$$C[i, w, l] = \underbrace{C[i - 1, w, l]}_{\notin} \vee \underbrace{(C[i, w - x_i, l - 1] \wedge w \geq x_i)}_{\in}$$

$$C[0, 0, l] = \text{true}, \quad C[0, w, l] = \text{false}, \text{ if } w > 0$$

$$C[i, 0, l] = \text{true}, \quad C[i, w, 0] = \text{false}, \text{ if } w > 0$$

## The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

## The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins?

$C[v, k]$

## The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins?

$C[v, k]$

**What is the first coin to use?**

## The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins?

$C[v, k]$

**What is the first coin to use?**

$$C[w, l] = \bigvee_{i: x_i \leq w} C[w - x_i, l - 1]$$

## The Change-Making Problem

*Unbounded repetition with  $\leq k$  coins overall*

$C[w, l]$  : Possible to make change for  $w$  with  $\leq l$  coins?

$C[v, k]$

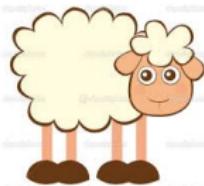
**What is the first coin to use?**

$$C[w, l] = \bigvee_{i: x_i \leq w} C[w - x_i, l - 1]$$

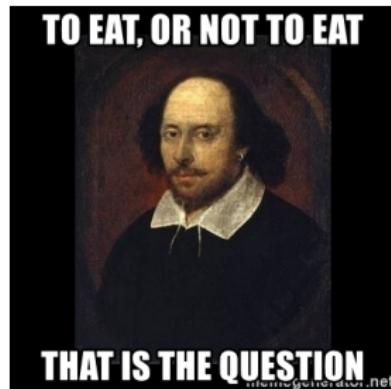
$$C[0, l] = \text{true},$$

$$C[w, 0] = \text{false, if } w > 0$$

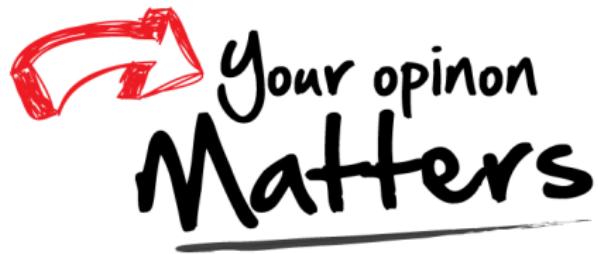
## Problem (Hungry-Lion Game)



## Problem (Hungry-Lion Game)







Office 302

Mailbox: H016

hfwei@nju.edu.cn