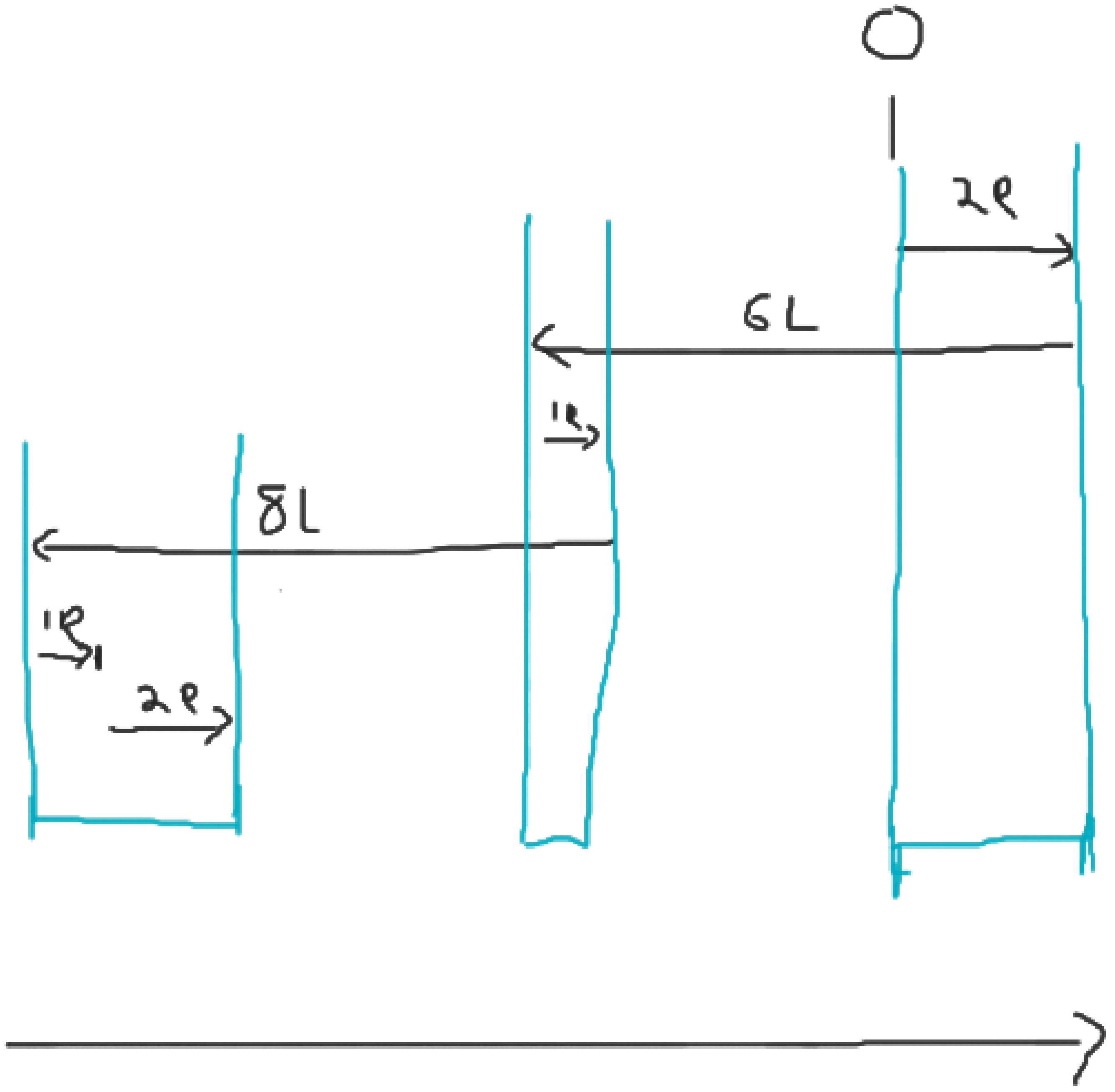
Self-Reflection Form: Until 9:30

6. Painting the Fence



(0, 2)
(2, -4)
(-4, -3)
(-3, -11)
(-11, -10)
(-10, -8)

Runtime $O(N \log N)$

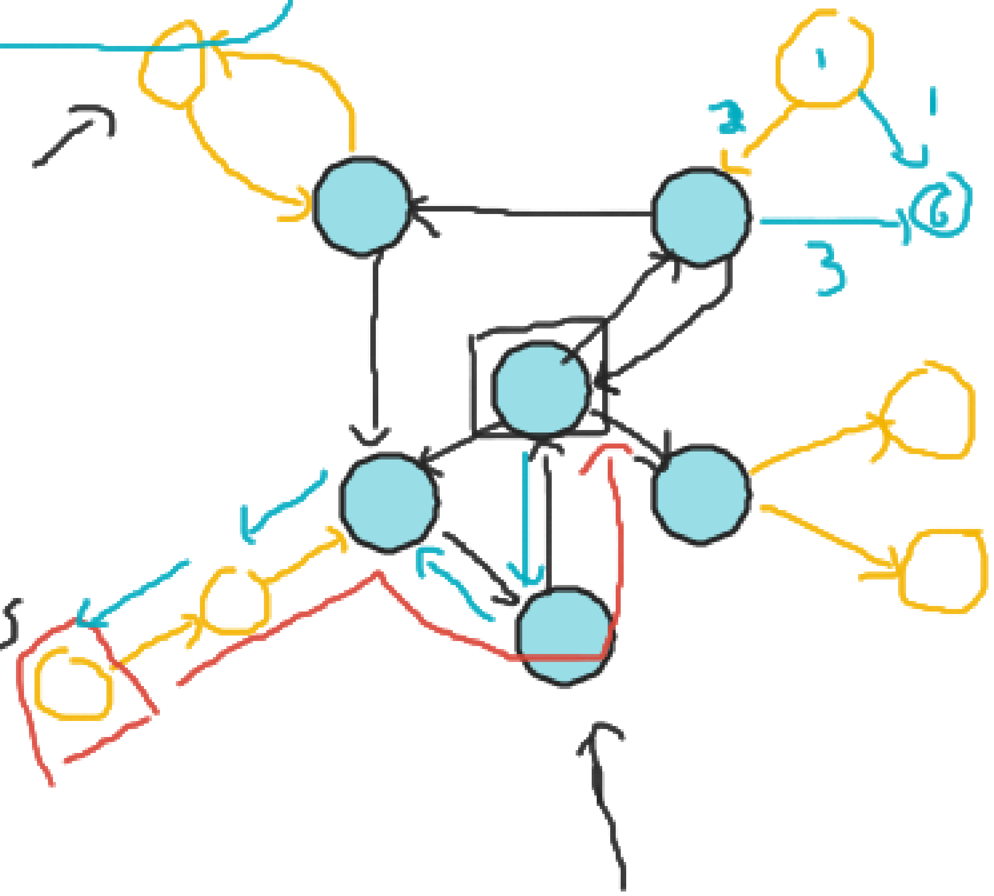
(-11, -10)
(-11, -3)
(-10, -8)
(-4, -4)
(-4, 2)
(0, 2)

coats ans 1

3. Vacation Planning

AlphaStar Academy

$a_i \to \dots \to h_j \to \dots \to b_i$



Naive

$O(Q M \log N)$

Floyd-Warshall

$O(N^3 + QK)$

Dijkstras

$O(KM \log N + QK)$

1. Dijkstras from hubs

$O(KM \log N)$

2 Reverse then (1)

$O(KM \log N)$

$1 \longrightarrow 6$

h
2
3

dist[1][2] + dist[2][1]

4. A Coin Game

DP[i][j] = max score at position i picking up j coins

$< = j$

$O(N)$

prefix

if $i + j > n$

$DP[i][j] =$ sum of all coins after i, Σ

$i = 1$, $j = 2$, 3, 7, 2

else

$$DP[i][j] = \max\left(\Sigma - DP[i+j][K],\ DP[i][j-1]\right)$$

$\min(j \cdot 2, N)$

s.t. $K \in [1 \ldots j \cdot 2]$

$O(j)$ $O(N)$

sig = 0

for i: N-1 → 0

sig += d[i+1]

for j: 0 → N

max(sigma - dp[i+j][min(j*2,N)], dp[i][j-1])

Break 1:	10:30 AM - 10:45 AM
Lunch:	12:15 PM - 1:30 PM
Break 2:	3:00 PM - 3:15 PM
End:	4:45 PM

$$f(i) = f(i-1) + f(i-2)$$

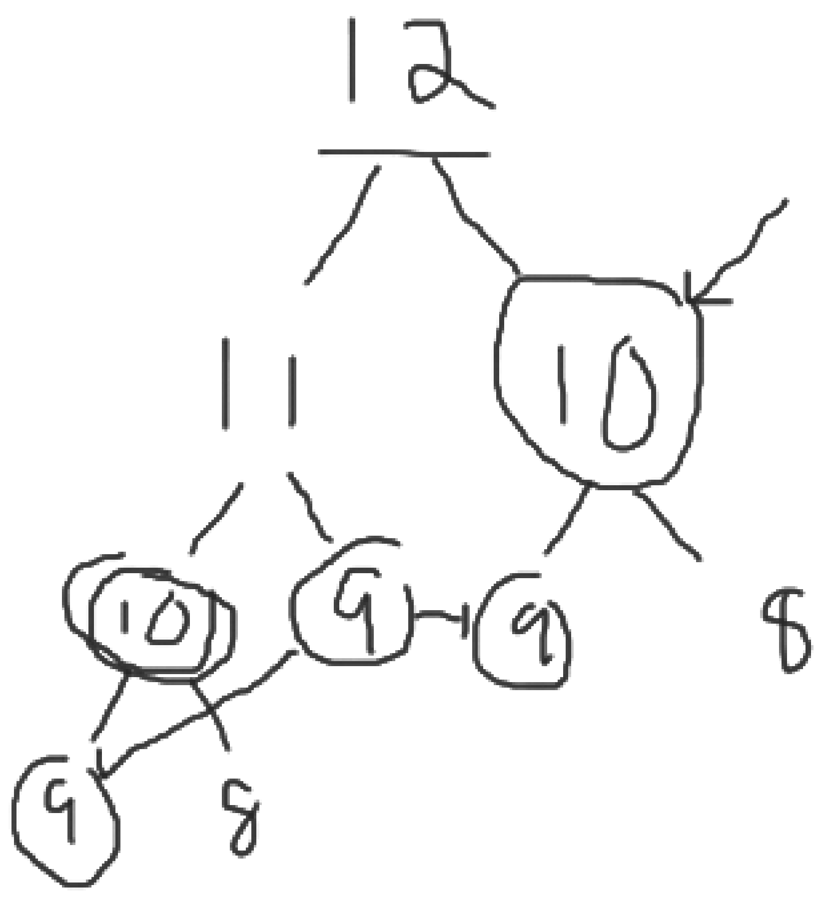
dp init -1

$f(\phi) = 1$

$f(2) = 1$

```
def f(i)
    if i <= 2
        return 1
    else
        dp[i] = return f(i-1) + f(i-2)
    return dp[i]
```

```
dp[i] != -1
    return dp[i]
```



O(N)

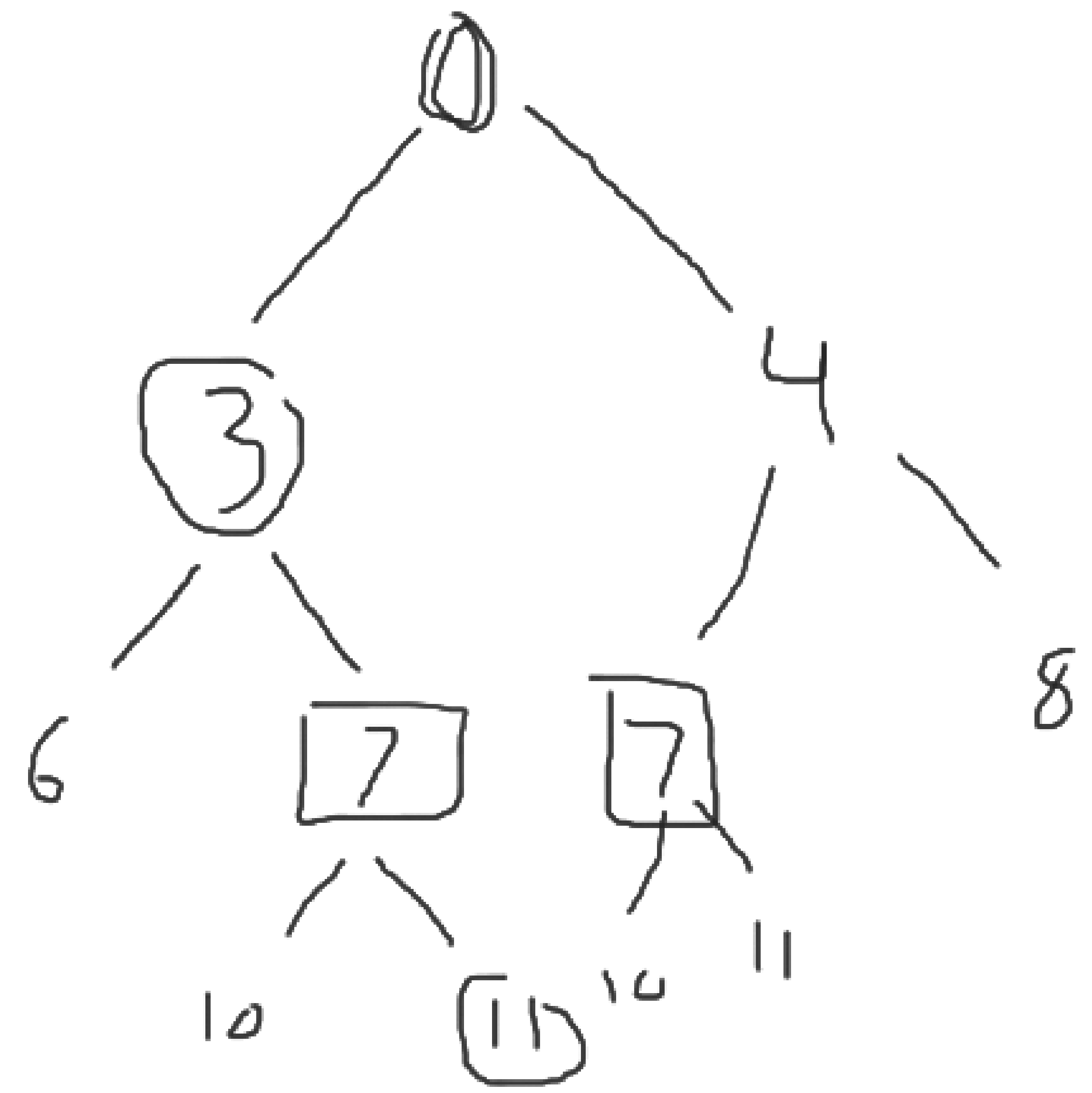
knapsack

11 lbs

weights

3 lbs ∞

4 lbs ∞



dp[0] = true   ans

for i : 0 → 11

  if dp[i] :

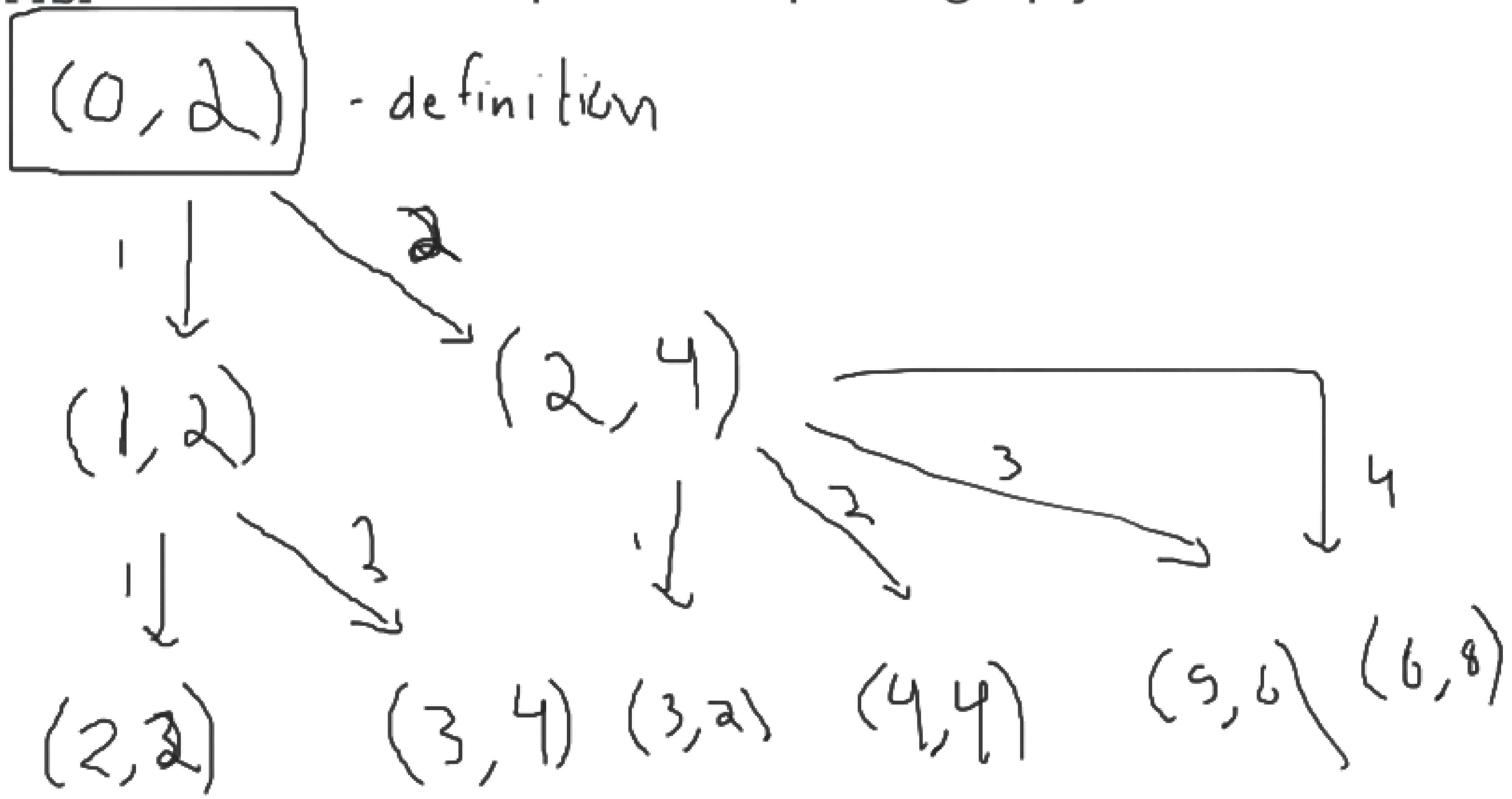
    dp[i+3] = true

    dp[i+4] = true

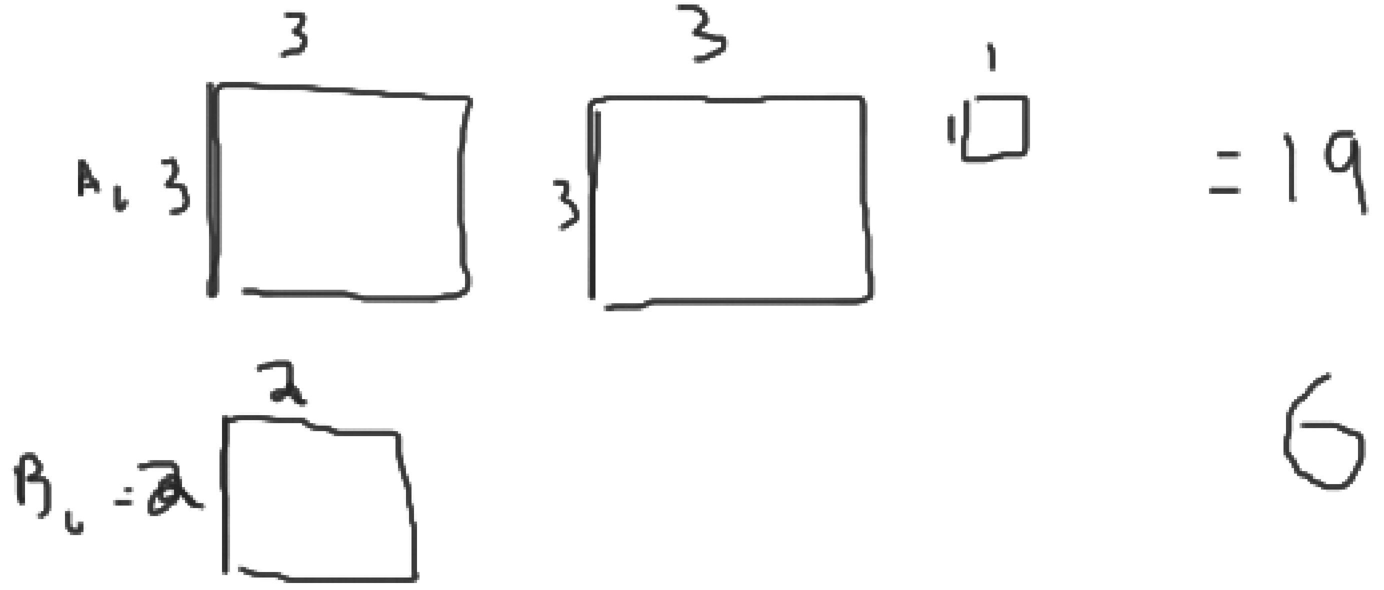
dp[i] = can we do weight i / true or false

dp(i) → dp[i+3], dp(i+4)

DP[i][j] = max score at position i picking up ≤ j coins



Tile Exchanging



DP STATE

DP[i][j] = Area i, Tiles 1...j → min cost

BASE CASE

$$DP[0][0] = 0$$

TRANSITION

$$DP[i][j] = \min_{k} \Big[\underbrace{dp[i-k^2][j-1]}_{\text{with } j-1 \text{ tiles}} \Big] + \underbrace{(A_j - K)^2}_{\text{cost of } j\text{-th tile}}$$

$B_j = K$

$$|A_i - B_i|, \; |A_i - B_i| = (A_i - B_i)^2$$

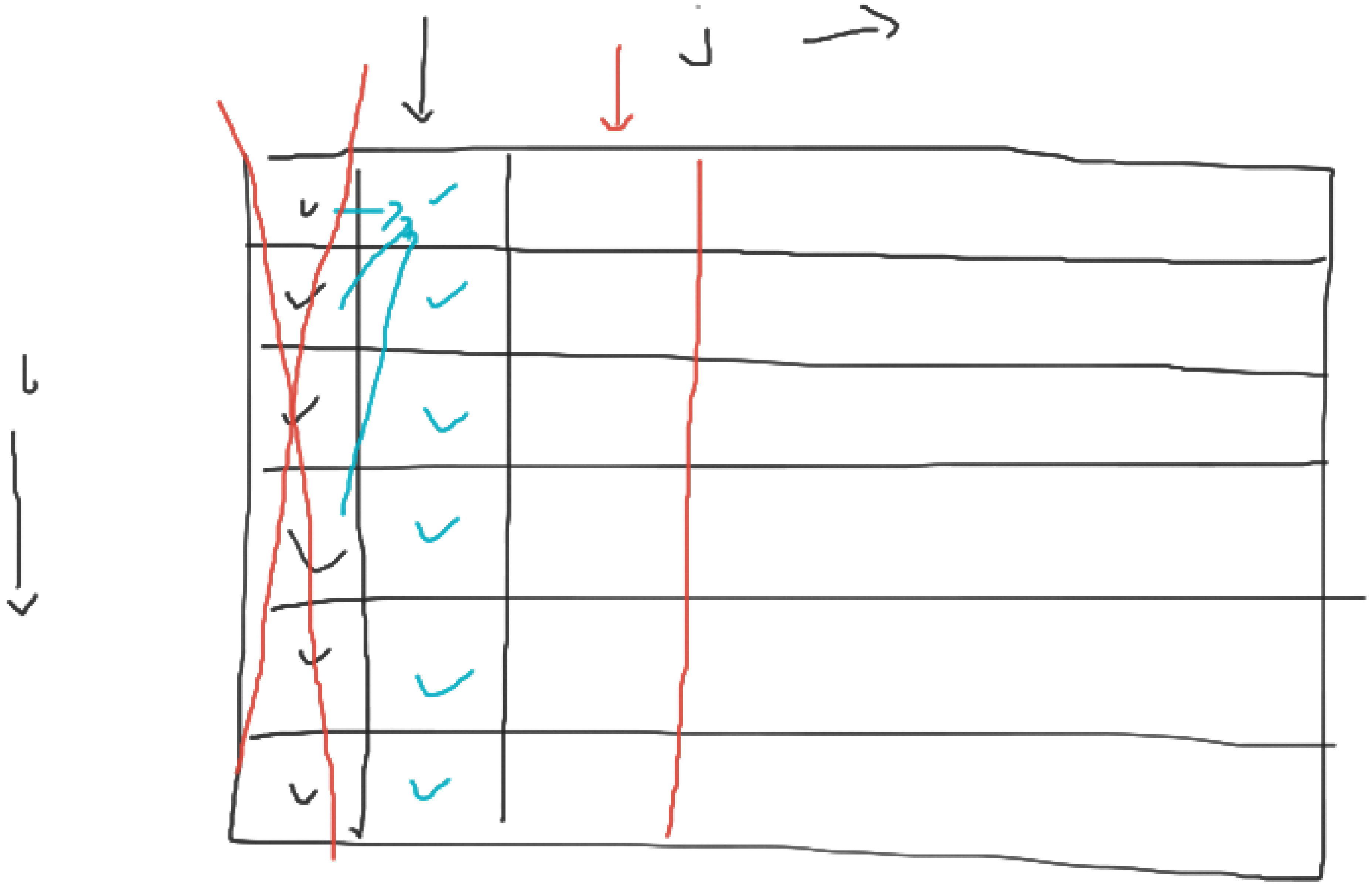
RUNTIME

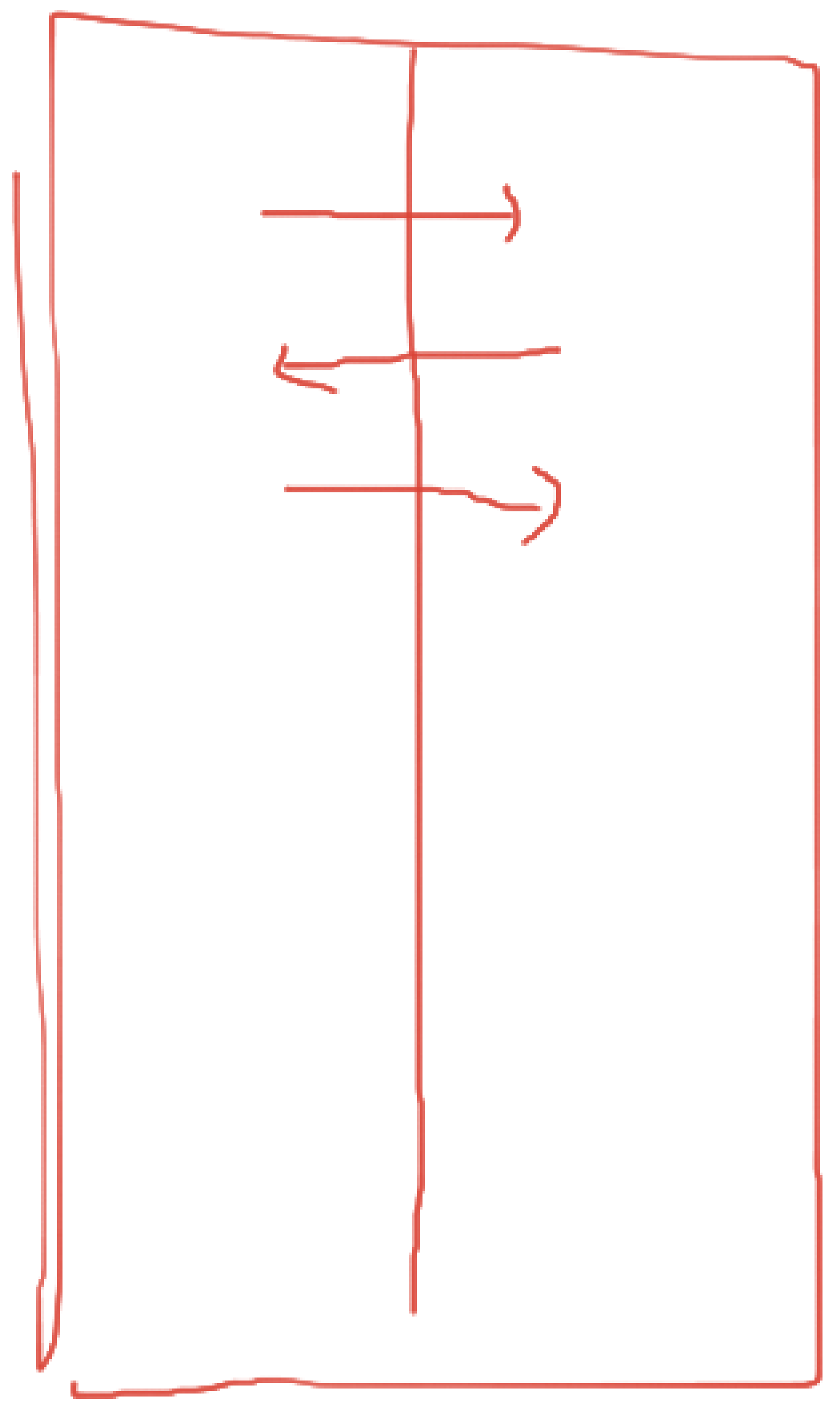
$$O(\underbrace{\overset{i}{M}\,\overset{j}{N}}_{\text{dp state}}\sqrt{M}) = 10^4 \cdot 10^1 \cdot 10^2 = 10^7$$

$N \le 10$

$M \le 10{,}000$

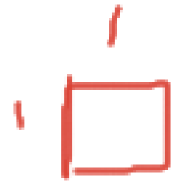
$$1 \xrightarrow{1} 2 \xrightarrow{1} 3 \qquad 1 \xrightarrow{4} 3$$


j
i



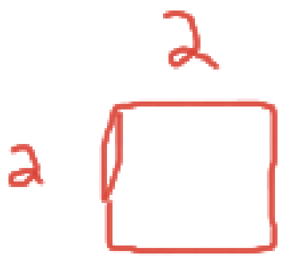
$\underline{i = 6}$, $\underline{j = 3}$

Case 1



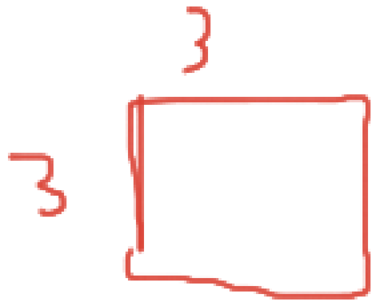
$dp[5][2]$

Case 2



$dp[2][2] + (2-1)^2$

Case 3



5 3

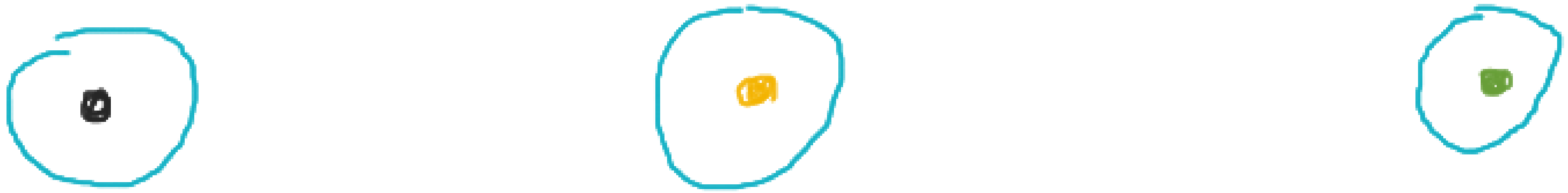
1 2

1 3

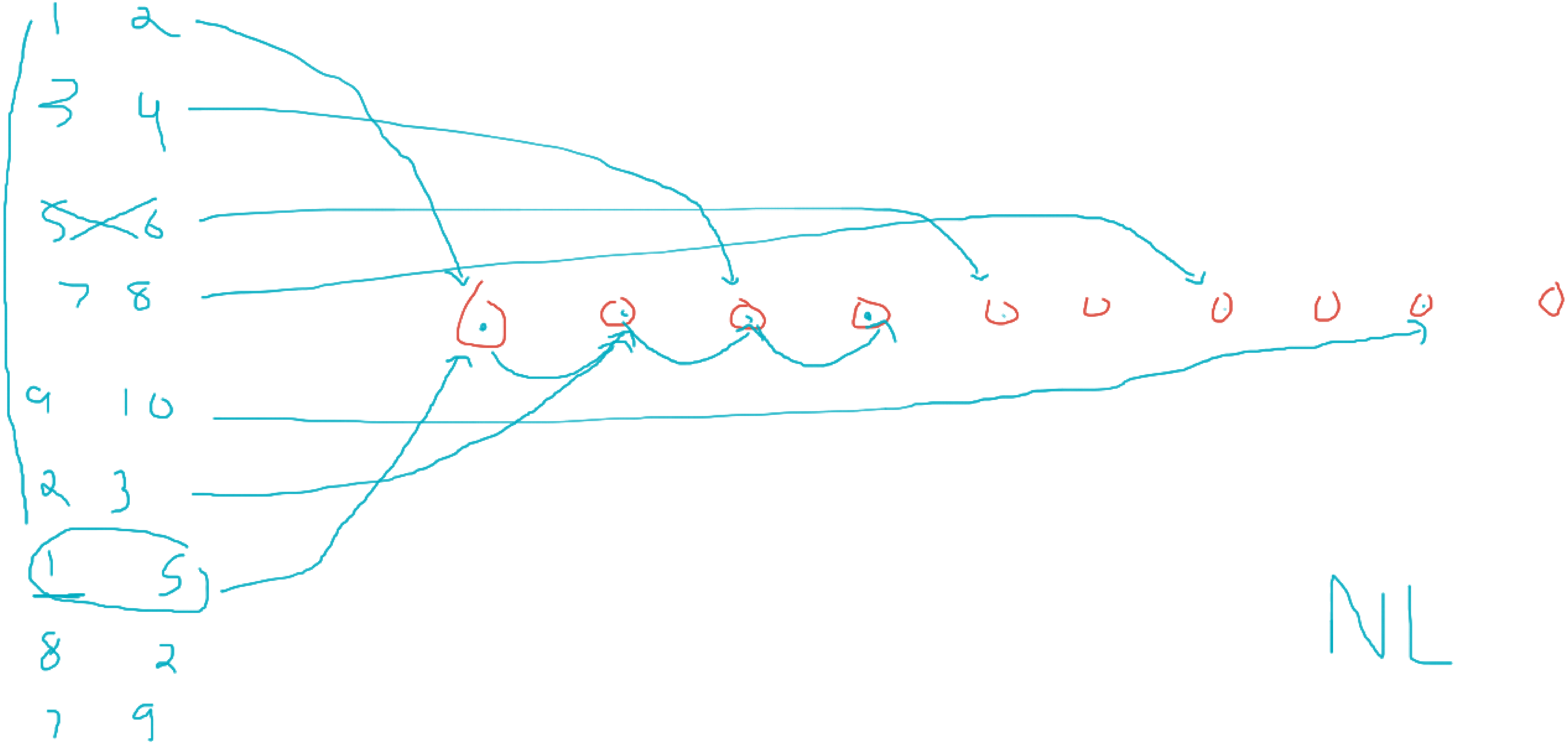
1 2

1 3

1 2



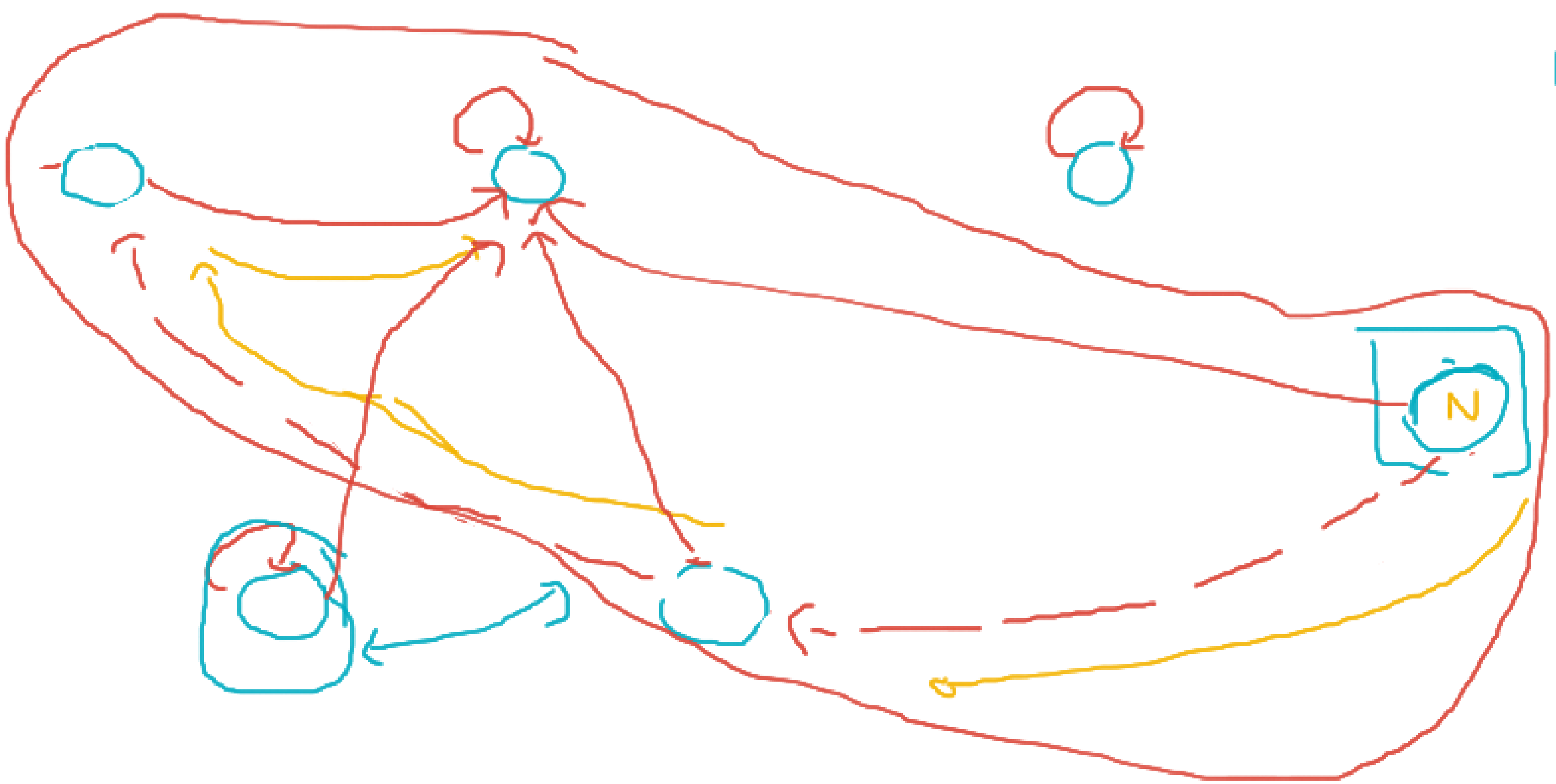
1. If the drawer Ai is empty, he stores the item i in that drawer
2. If the drawer Bi is empty, he stores the item i in that drawer
3. Try to move the item from Ai to its other drawer; if that one's filled too, try moving that item to its other drawer, and so on untilyou either succeed or get back to a previously seen drawer. In case of success, store the item i in the drawer Ai. In case of failure, continue to next rule.
4. Try moving the item from Bi to its other drawer; if that one's filled too, try moving that item to its other drawer, and so on until you either succeed or get back to a previously seen drawer. In case of success, store the item i in the drawer Bi. In case of failure, continue to next rule.
5. Give up and throw away the item i. For given pairs of drawers for each item, determine which items will be stored and which will be thrown away.



NL

$O(N)$

```
union A,B
  p[find(A)] = find(B)
```



find(N)

$O(N)$

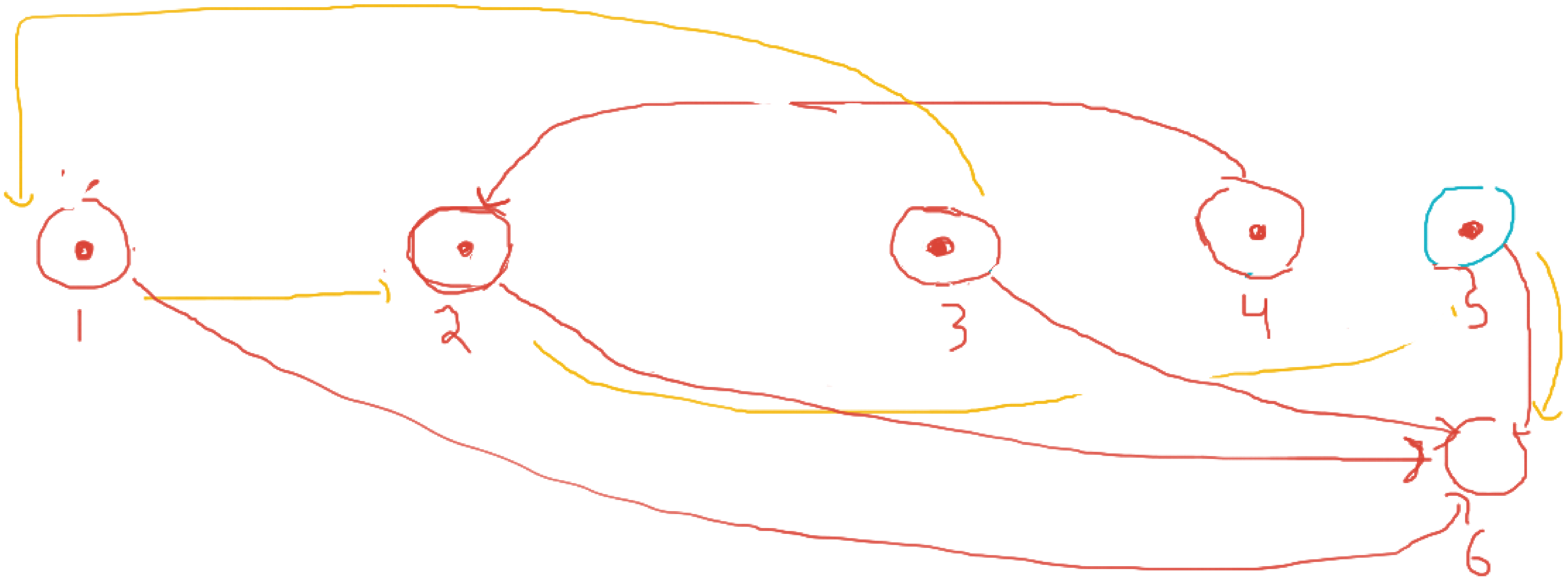
```
find A
  while p[A] != A
    A = p[A]
  return A
```

1,2

1,3

4,2

3, 5



O(NL)

$$A \begin{bmatrix} \cancel{} \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix} \quad \begin{matrix} 2 & 1 \\ 1 & \\ 3 & \\ 4 & \\ 5 & \end{matrix}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & & 0 & 1 \end{bmatrix}$$

$\boxed{1\ 3\ 4\ 5\ 2}$ → 1 3 4 5 2

$\boxed{2\ [1\ 3\ 4]\ 5}$

$[1\ 3\ 4]\ 5\ 2$

$$\begin{bmatrix} 1 & & & & \\ 0 & & 1 & & \\ 0 & & & 1 & \\ 0 & & & & 1 \\ 0 & 1 & & & \end{bmatrix}$$

shuffle

+ left

$$A$$

$$= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 1\cdot 0 + 2\cdot 1 + 3\cdot 0 \\ 1\cdot 1 + 2\cdot 0 + 3\cdot 0 \\ 1\cdot 0 + 2\cdot 0 + 3\cdot 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix}$$

$$A^N \cdot \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$\begin{array}{cccc} & 4 & 2 & 1 \\ 0b & 1 & 1 & 1 \end{array}$$

$4+2+1=7$

$$2^7 = 2^4 \cdot 2^2 \cdot 2^1$$

$$2^2 = 2^1 \cdot 2^1$$

$$2^4 = 2^2 \cdot 2^2$$

$0b101$

2^5

$$\begin{tikzcd} 2^1 \arrow[r] \arrow[r, bend left] & 2^2 \arrow[r] \arrow[r, bend left] & 2^4 \end{tikzcd}$$