Lab12 Solution

YAO ZHAO

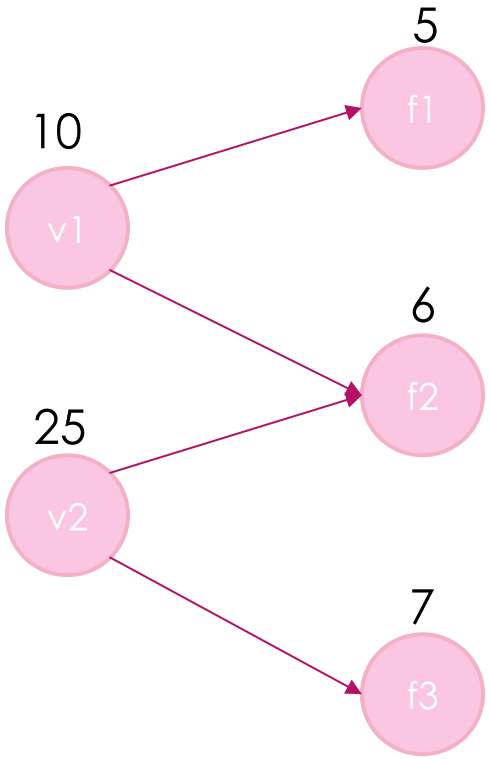
Lab12.A Cyclops

- As a superhero, you possess a remarkable eye with a range of optional functions labeled from 1 to n. Each function i incurs a cost of a_i yuan when integrated into your eye. Alongside your abilities, there exist m supervillains that you must confront. Each supervillain, denoted as the j-th villain, requires a specific subset S_j of eye functions to be installed in order to be defeated. The defeat of each supervillain yields a reward of b_j yuan.
- Your objective is not to defeat all the supervillains but rather to strategize in a way that maximizes your profits.

Sample Input

2 3
10 2 1 2
25 2 2 3
5 6 7

n m S_j s_j b_j a_i



	reward	cost	profits
defeat v1:	10	5+6 = 11	-1
defeat v2:	25	6+7 = 13	12
defeat v1+v2:	10+25	5+6+7 = 18	17

Sample Output

1 2
1 2 3
17

Key point

- The key point is how to construct a graph, convert to a familiar question to you.

Sample Input

2 3 n

10 2 1 2 m

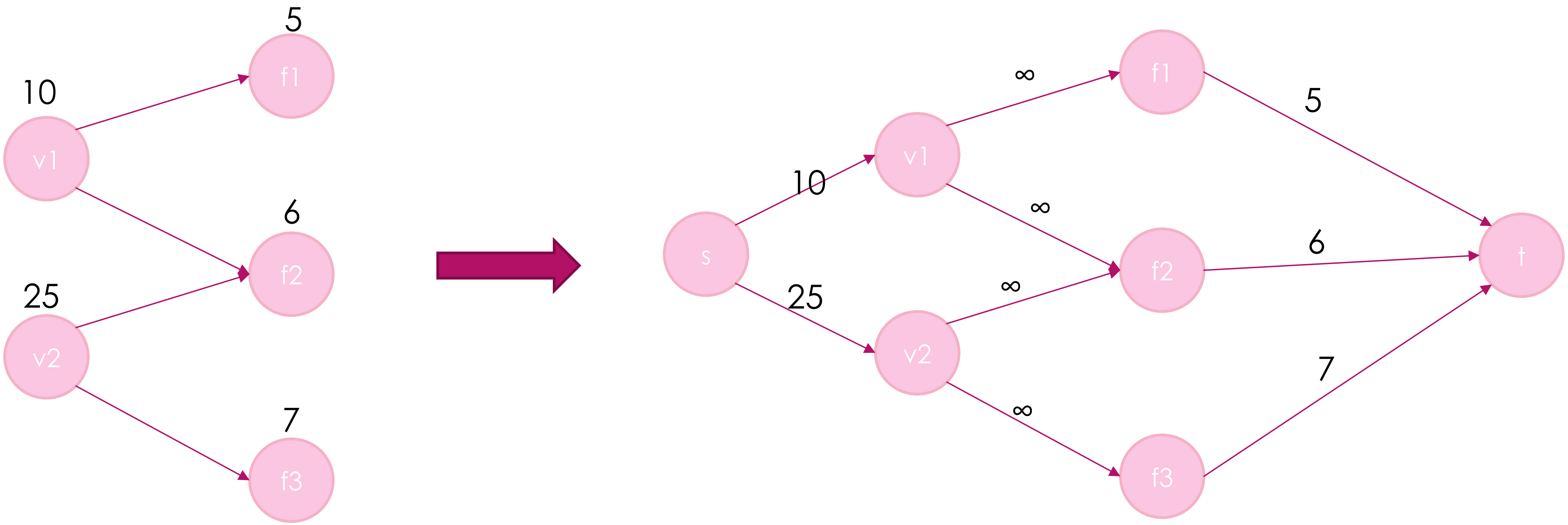
25 2 2 3 S_j

5 6 7 s_j

b_j

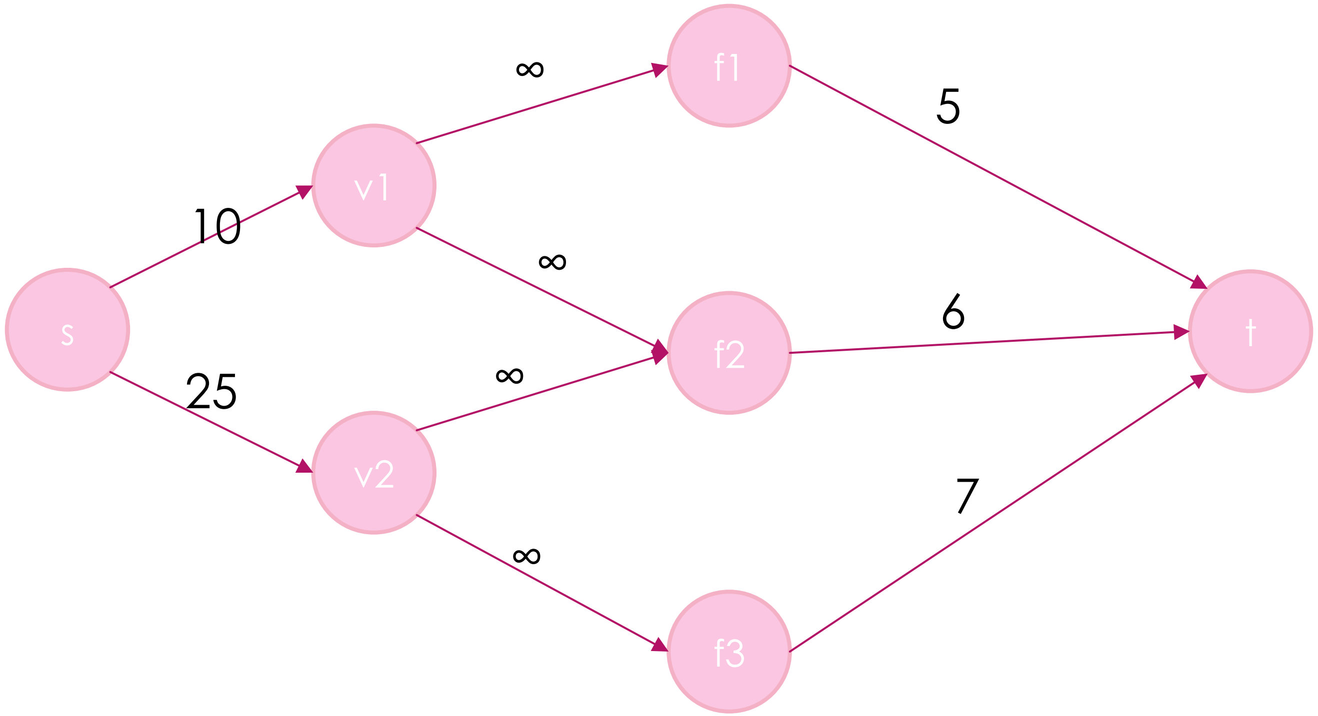
a_i

- Create the source point s and the sink point t.
- Create m vertices to represent m supervillains, add m edges from source s to each vertex with weight b_j;
- Create n vertices to represent n functions, add n edges from each vertex to the sink point t with weight a_j;
- Add edges with weight ∞ from the supervillain vertices to the function vertices according the map between b_j to S_j



Analysis

- The problem converts to the minimum cut problem of the following network G'.Assumes that the minimum cut vertices set is (A', B'), A'- {s} is the optimal project set. Assume the minimum cut is c， while C=$\sum b_j$, the maximum profit is C-c。



C is the sum of all supervillains' rewards.

C = 10+25=35

c is the minimum cut in G'

c = 18(Max flow in G' = 18)

Maximum profit = C-c = 35 – 18 = 17

c represent the cost of all defeated supervillains + the rewards of the abandoned supervillains.

Lab12.B Cube

- LYC has an amazing $h \times a \times b$ cube and plans to cut it into two amazing parts. The cut surface could be considered as an integer-valued function (x, y) whose domain is $([1, a] \times [1, b]) \cup (\mathbb{Z} \times \mathbb{Z})$ and should satisfy conditions below:
  - $1 \le f(x, y) \le h$
  - $\forall (x, y), (x', y')\ such\ that\ |x - x'| + |y - y'| = 1, |f(x, y) - (x', y')| \le D, where\ D\ is\ a\ non-negative\ integer\ given.$
- There is another function $v(x, y, z)$ $(1 \le x \le a, 1 \le y \le b, 1 \le z \le h)$ given. Your objective is to help LYC determine $\min \sum_{i,j} v(i, j, f(i, j))$.

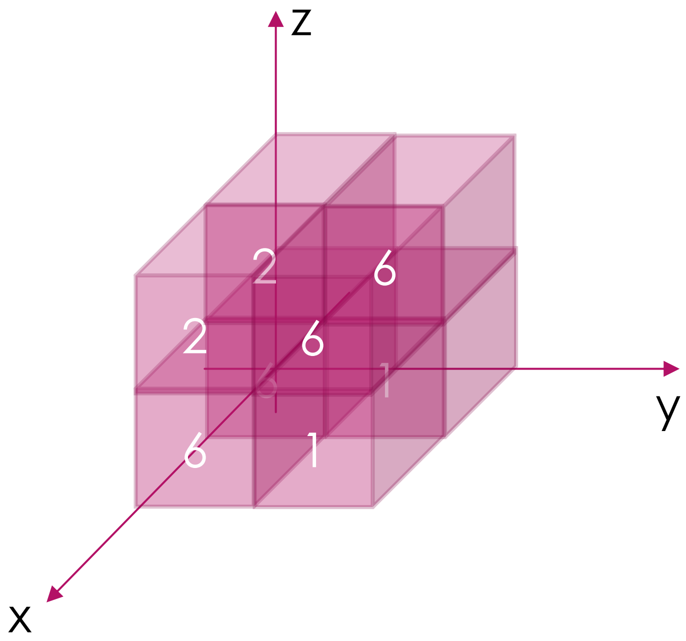
Sample Input

2 2 2
1
6 1
6 1
2 6
2 6

h b a D

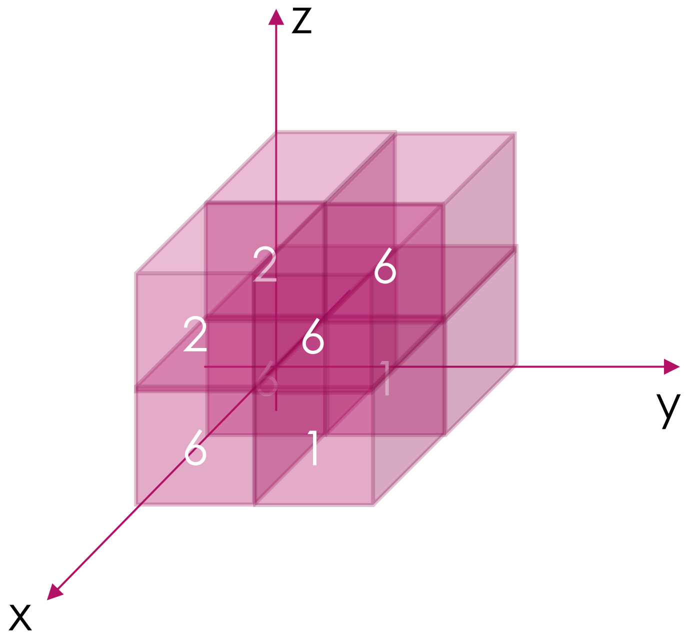
h

$a \times b \ matrix$



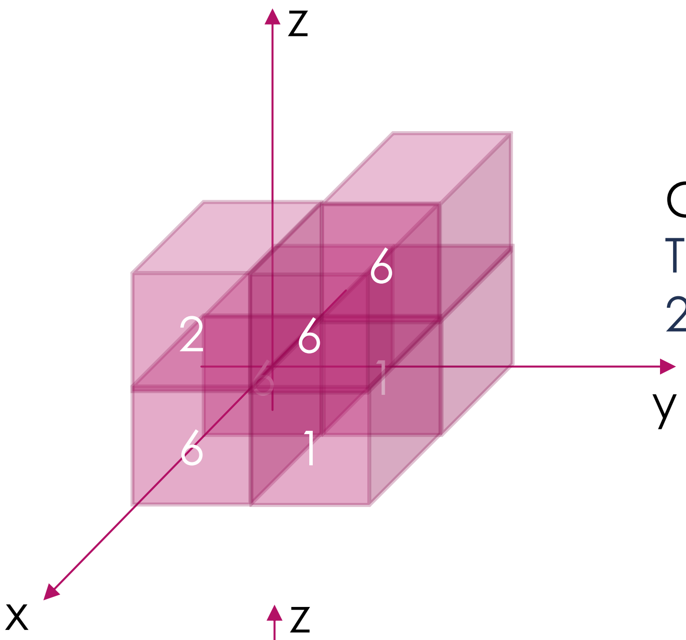
Sample Output

6

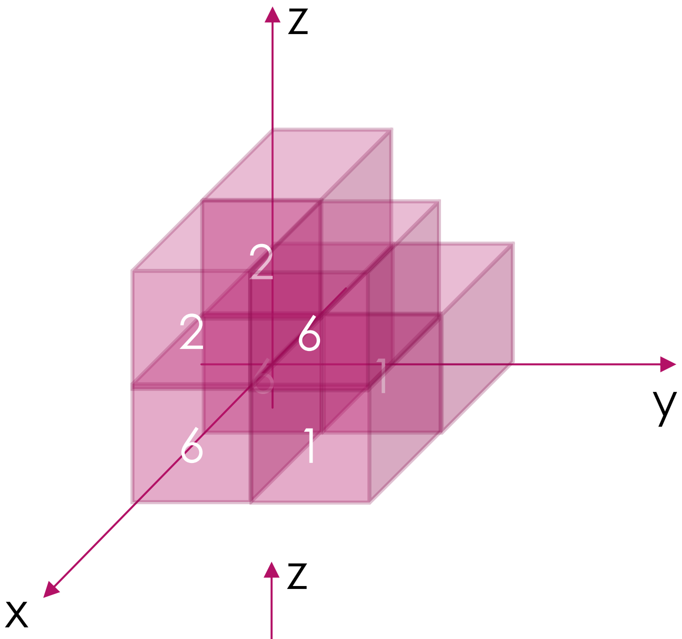


z

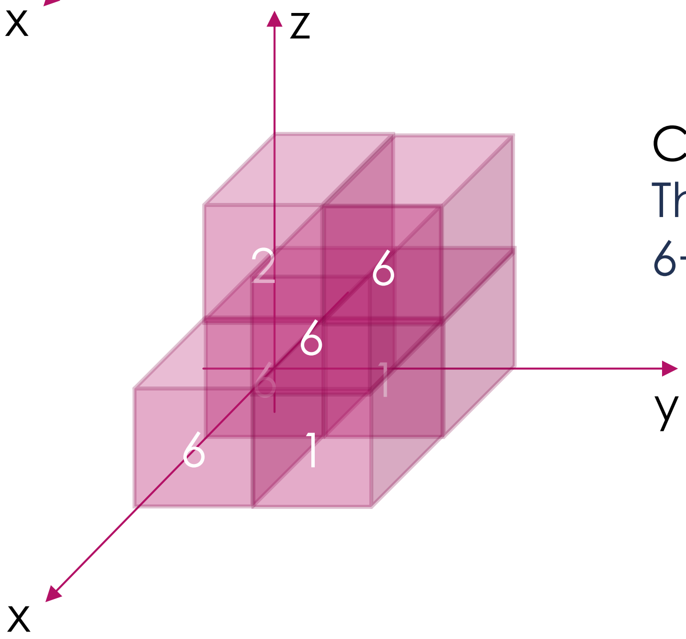
No cut. The cut surface: 2+2+6+6=16



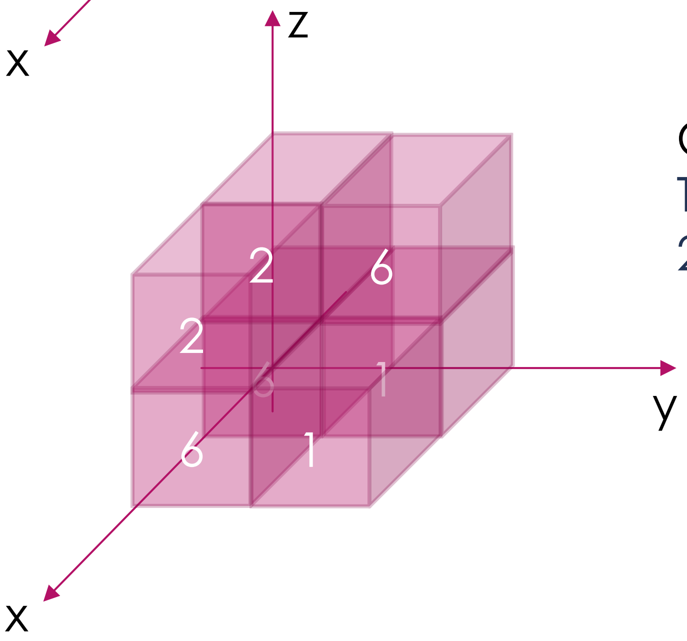
Cut (1, 1, 2)
The cut surface:
2+6+6+6=20



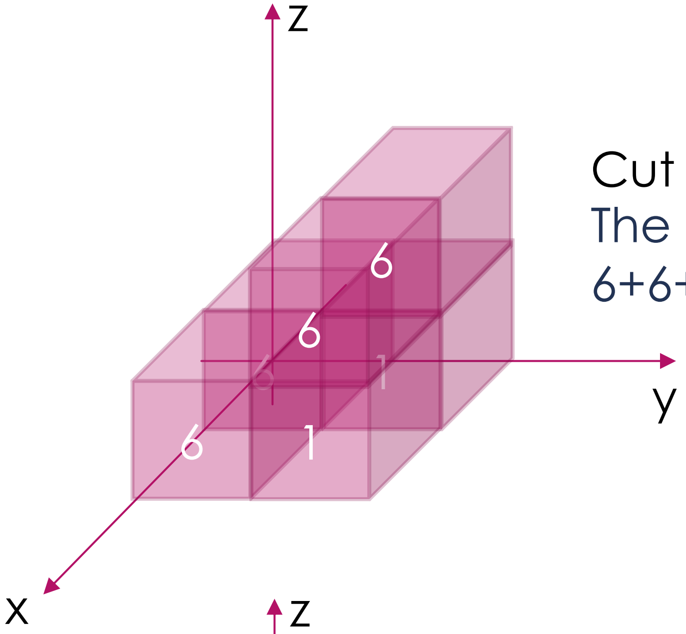
Cut (1, 2, 2)
The cut surface:
2+2+6+1=11



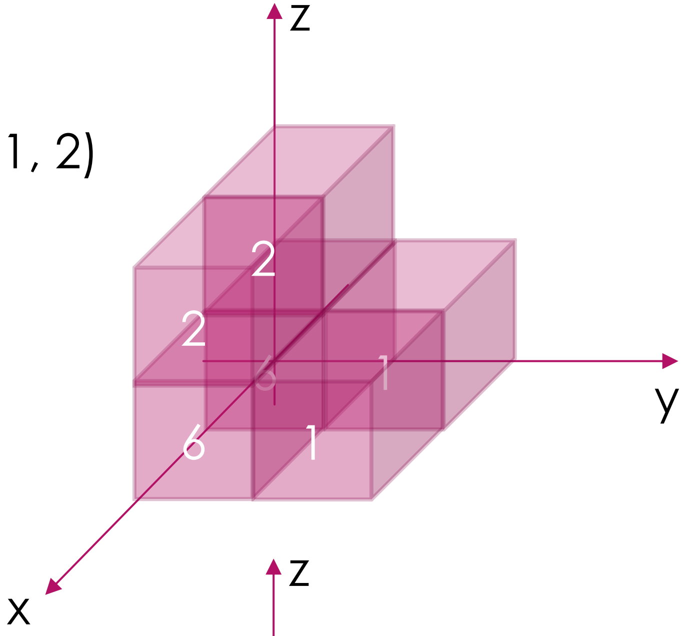
Cut (2, 1, 2)
The cut surface:
6+2+6+6=20



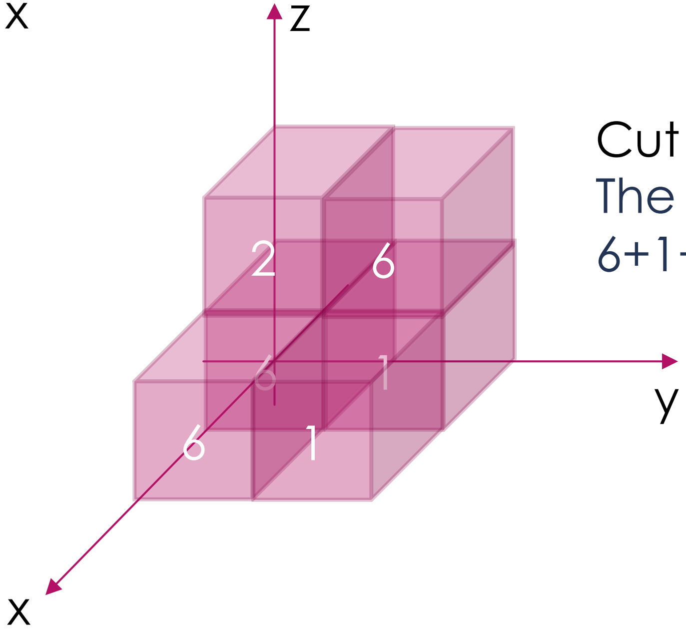
Cut (2, 2, 2)
The cut surface:
2+2+6+1=11



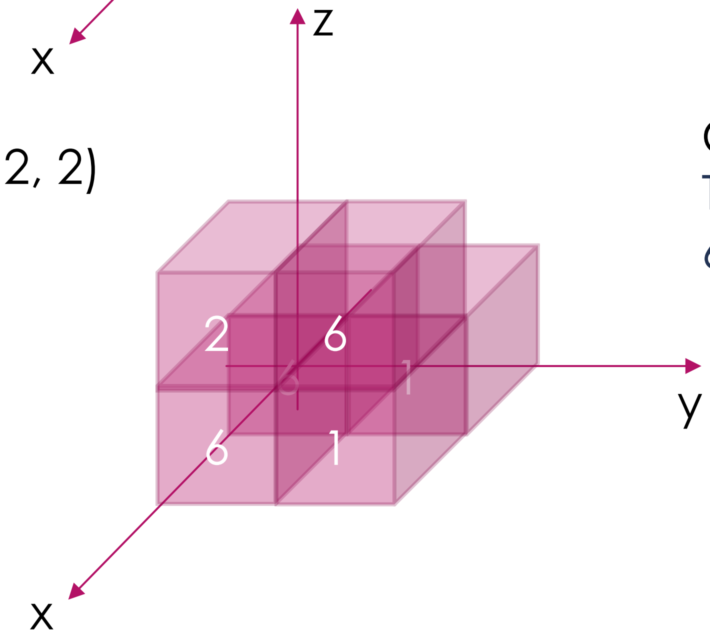
Cut (1, 1, 2) and (2, 1, 2)
The cut surface:
6+6+6+6=24



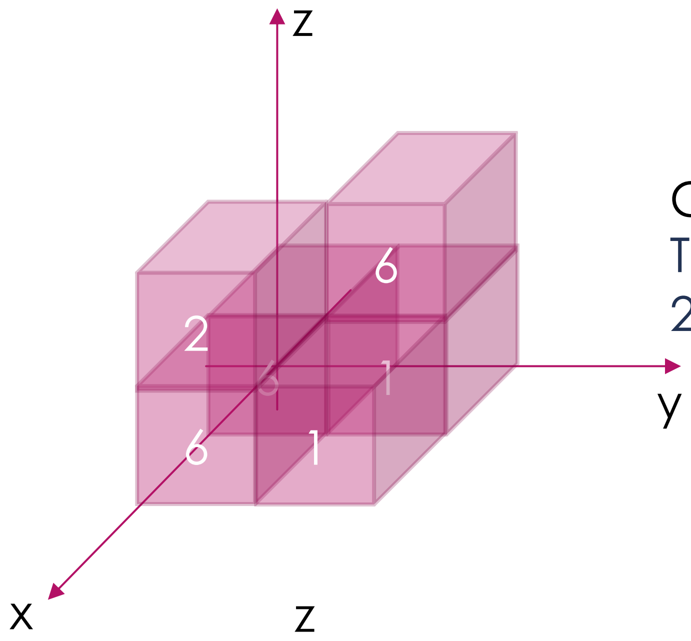
Cut (1, 2, 2) and (2, 2, 2)
The cut surface:
2+2+1+1=6



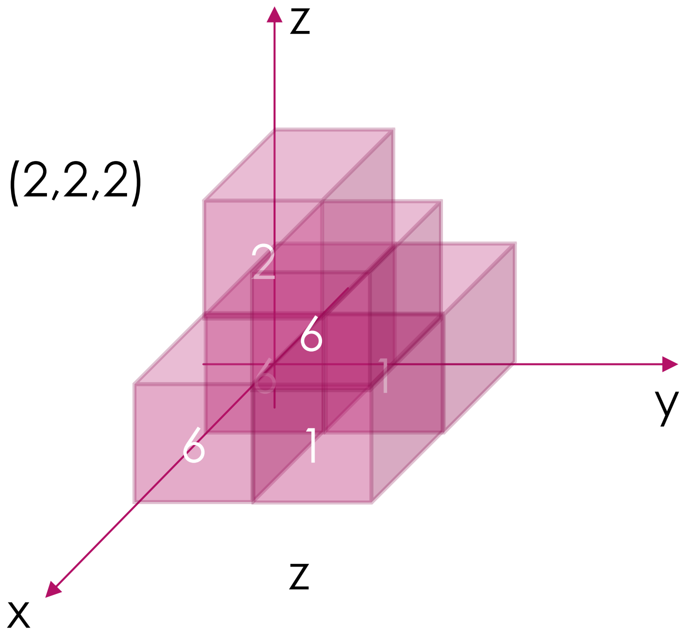
Cut (2, 1, 2)and(2, 2, 2)
The cut surface:
6+1+2+6=15



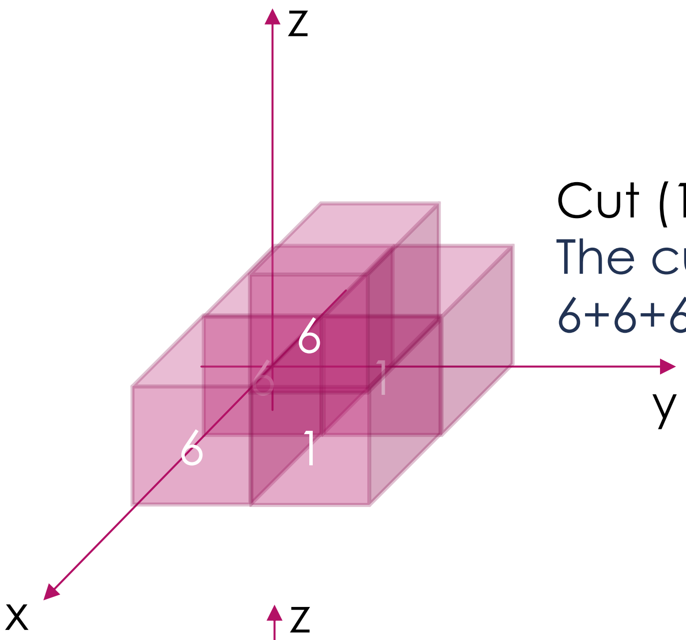
Cut (1, 1, 2) and (1, 2, 2)
The cut surface:
6+1+2+6=15



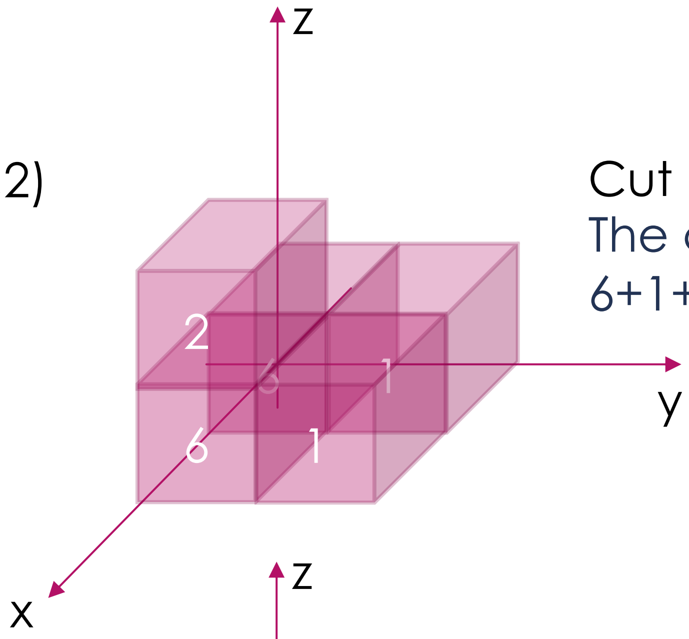
Cut (1, 1, 2) and (2,2,2)
The cut surface:
2+6+6+1=15



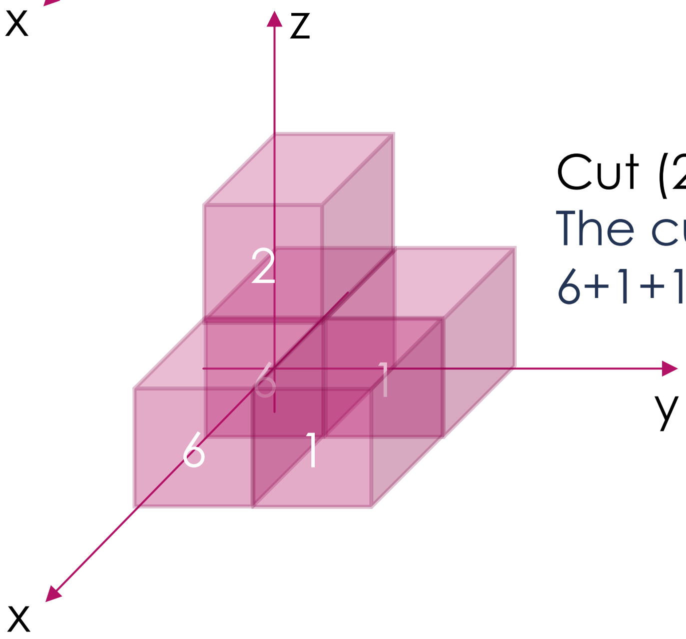
Cut (1, 2, 2) and (2,1,2)
The cut surface:
2+6+6+1=15



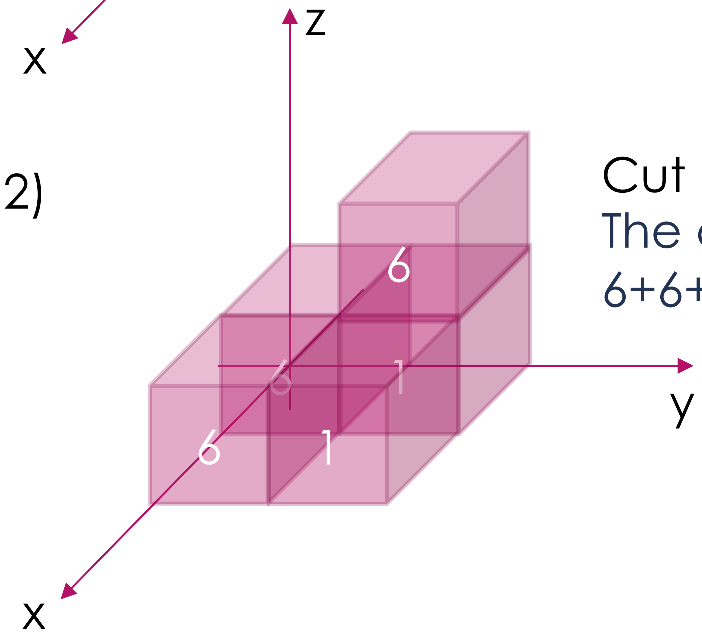
Cut (1,1,2),(1,2,2)(2,1,2)
The cut surface:
6+6+6+1=19



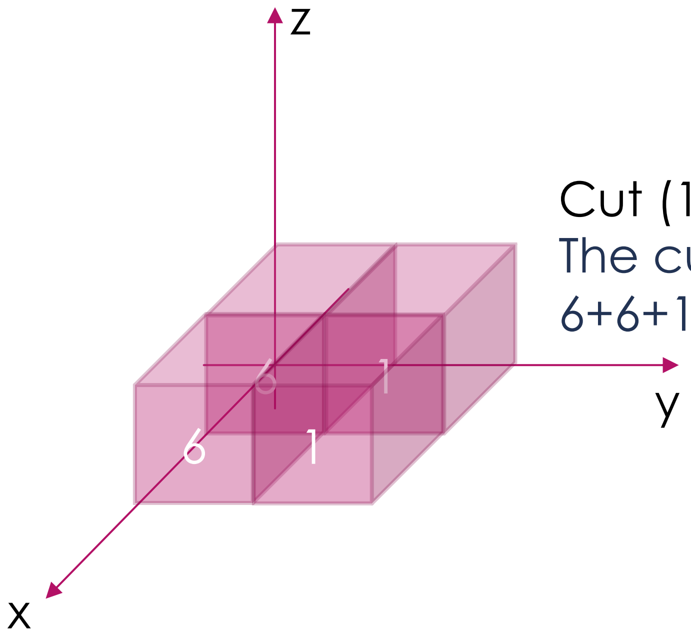
Cut (1,1,2),(1,2,2),(2,2,2)
The cut surface:
6+1+1+2=10



Cut (2,1,2)(1,2,2)(2,2,2)
The cut surface:
6+1+1+2=10



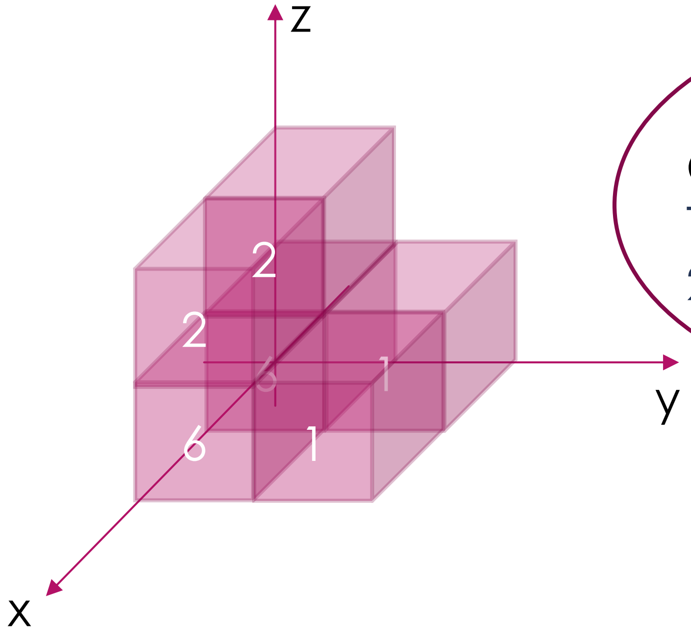
Cut (1,1,2)(2,1,2)(2, 2, 2)
The cut surface:
6+6+1+6=19



Cut (1,1,2),(1,2,2)(2,1,2)(2,2,2)

The cut surface:

6+6+1+1=14



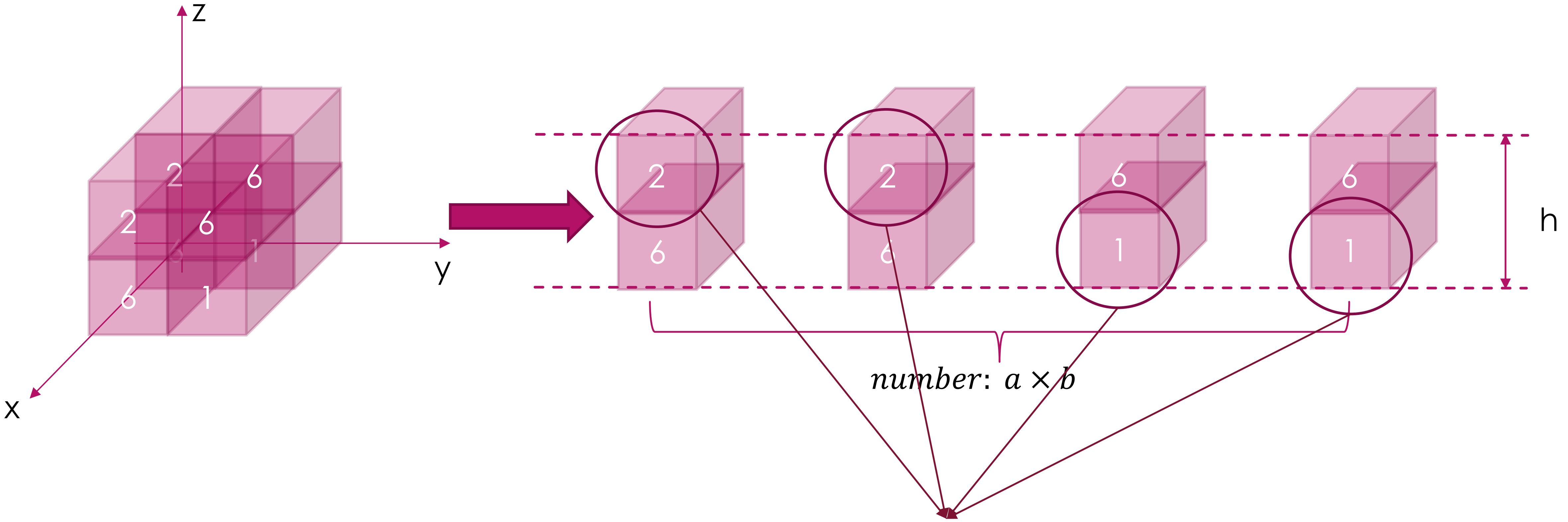
Cut (1, 2, 2) and (2, 2, 2)
The cut surface:
2+2+1+1=6

This is the minimum sum

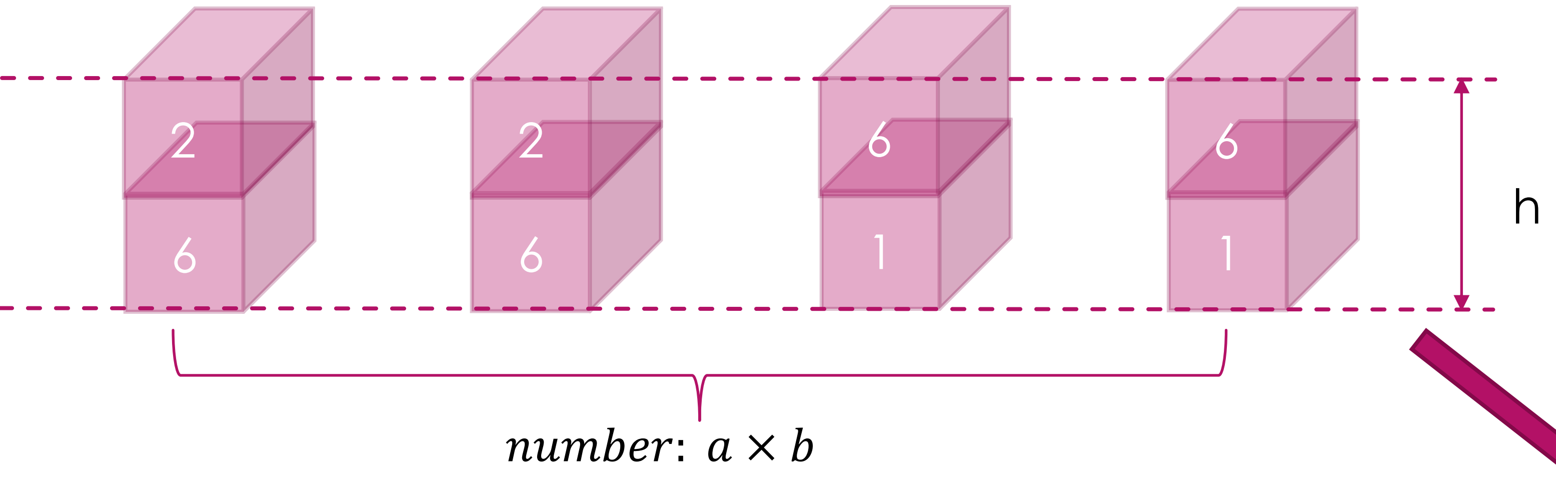
Sample Output

6

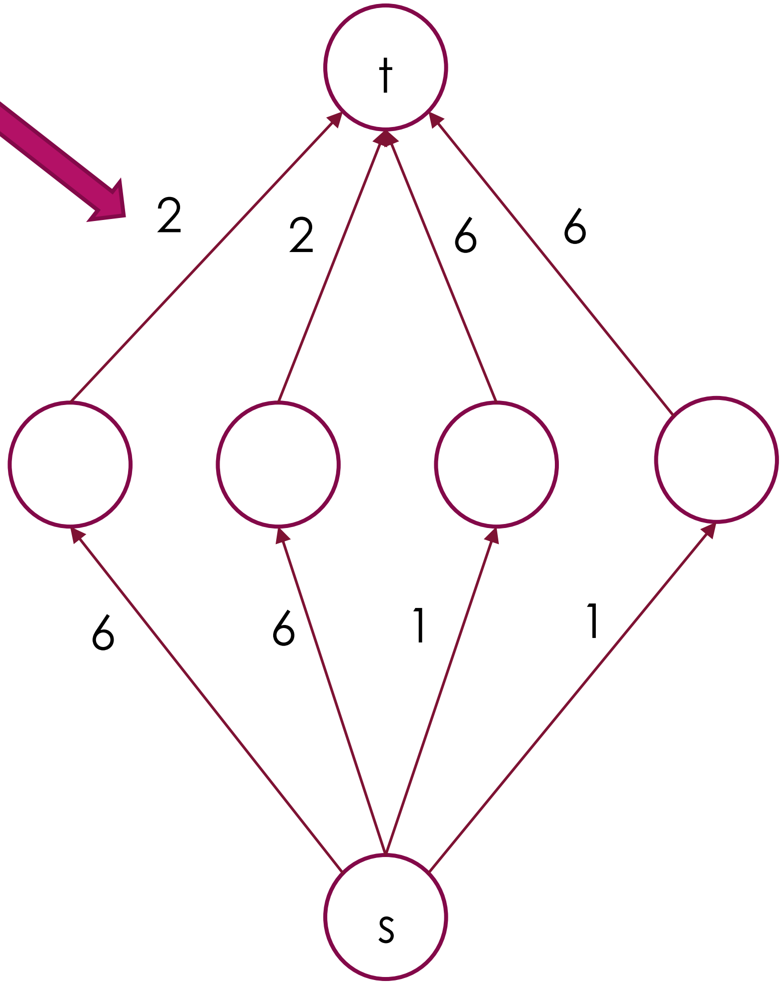
Suppose $|f(x,y)-(x',y')| \leq D$ is not considered



get the minimum value in each vertical axis stack cube



Transform to get the minimum cut in the left graph



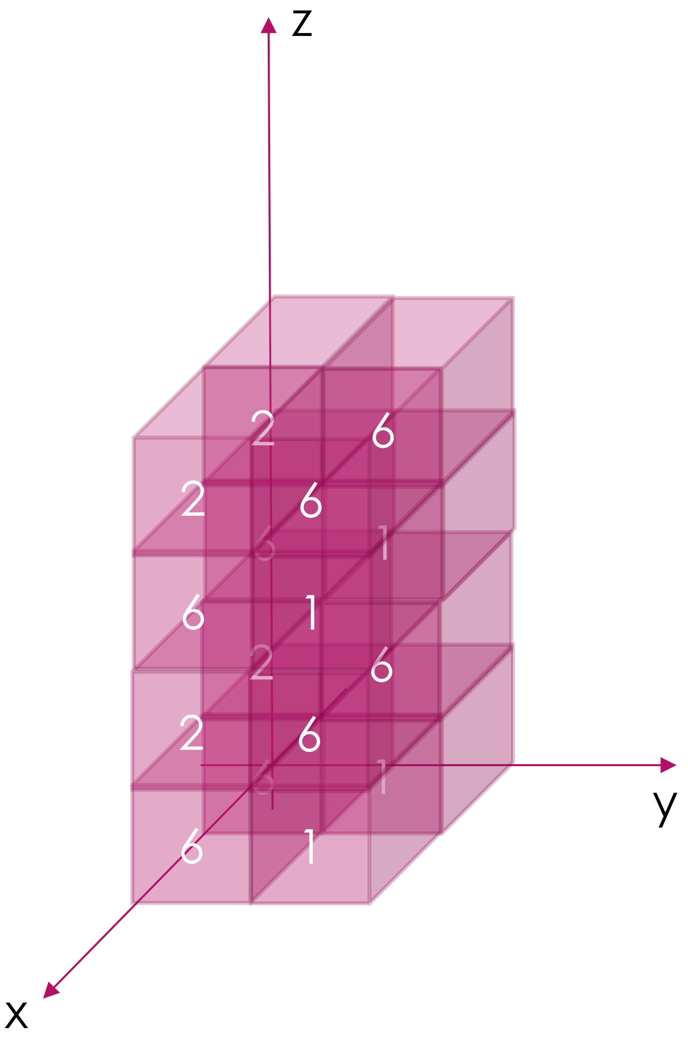
Sample Input

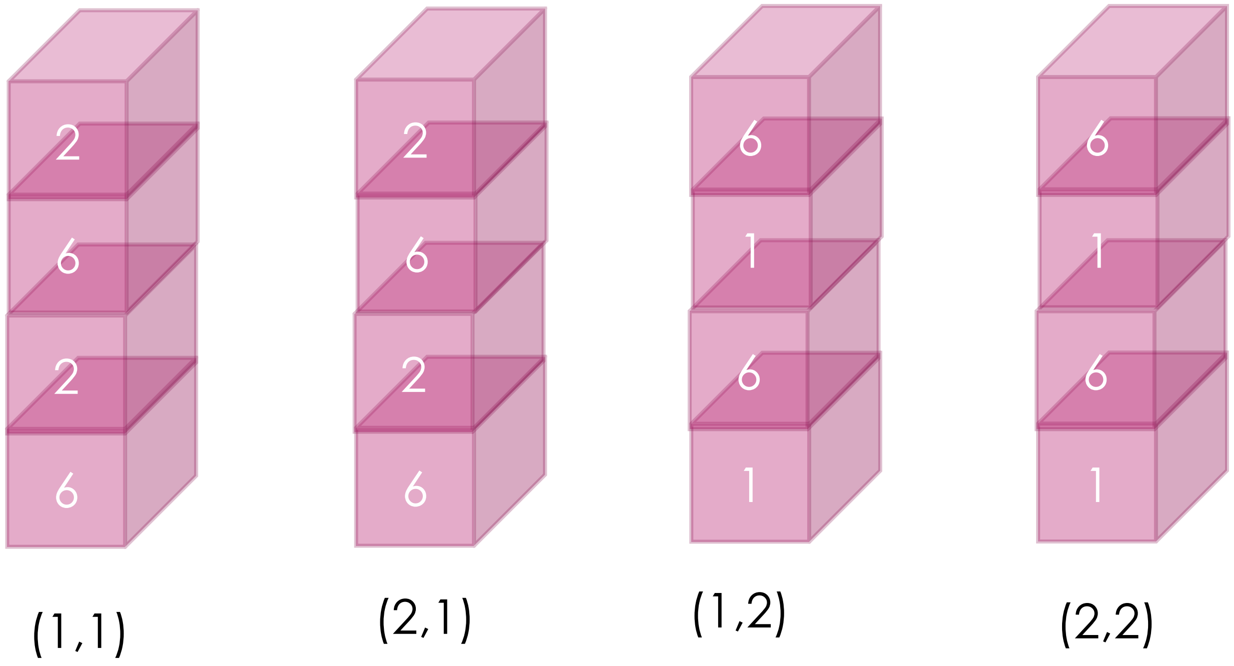
2 2 4
1
6 1
6 1
2 6
2 6
6 1
6 1
2 6
2 6

a, b, h, D

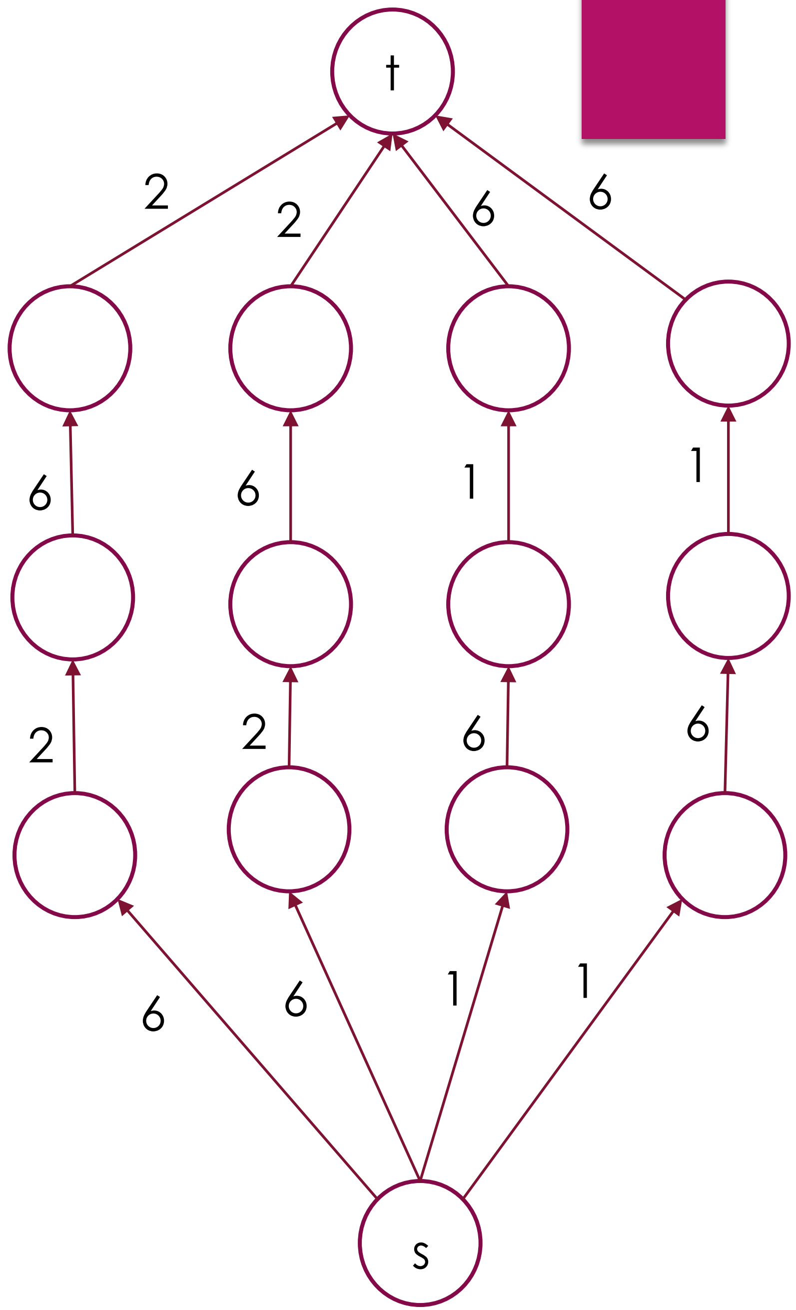
h

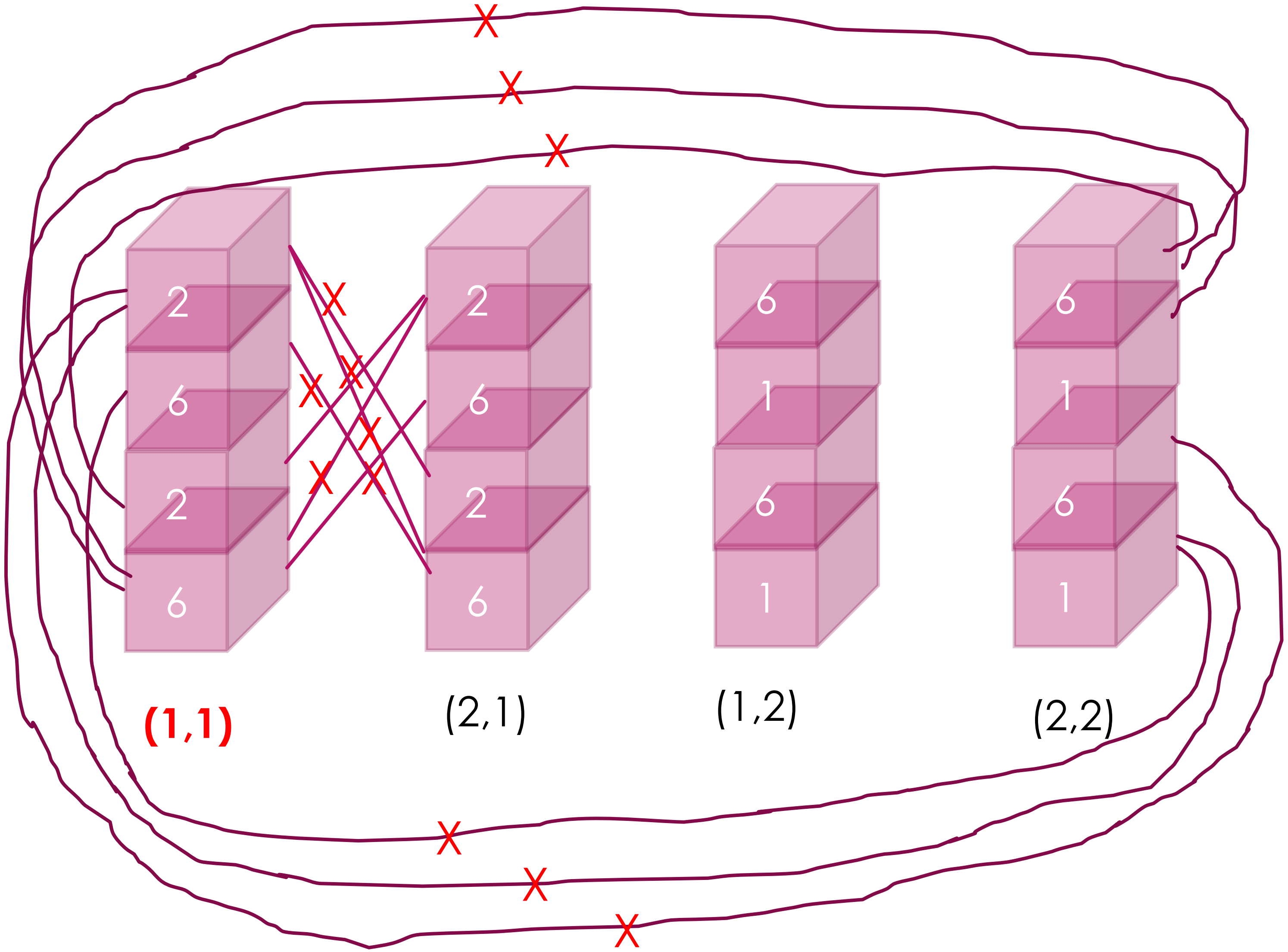
$a \times b$ $matrix$

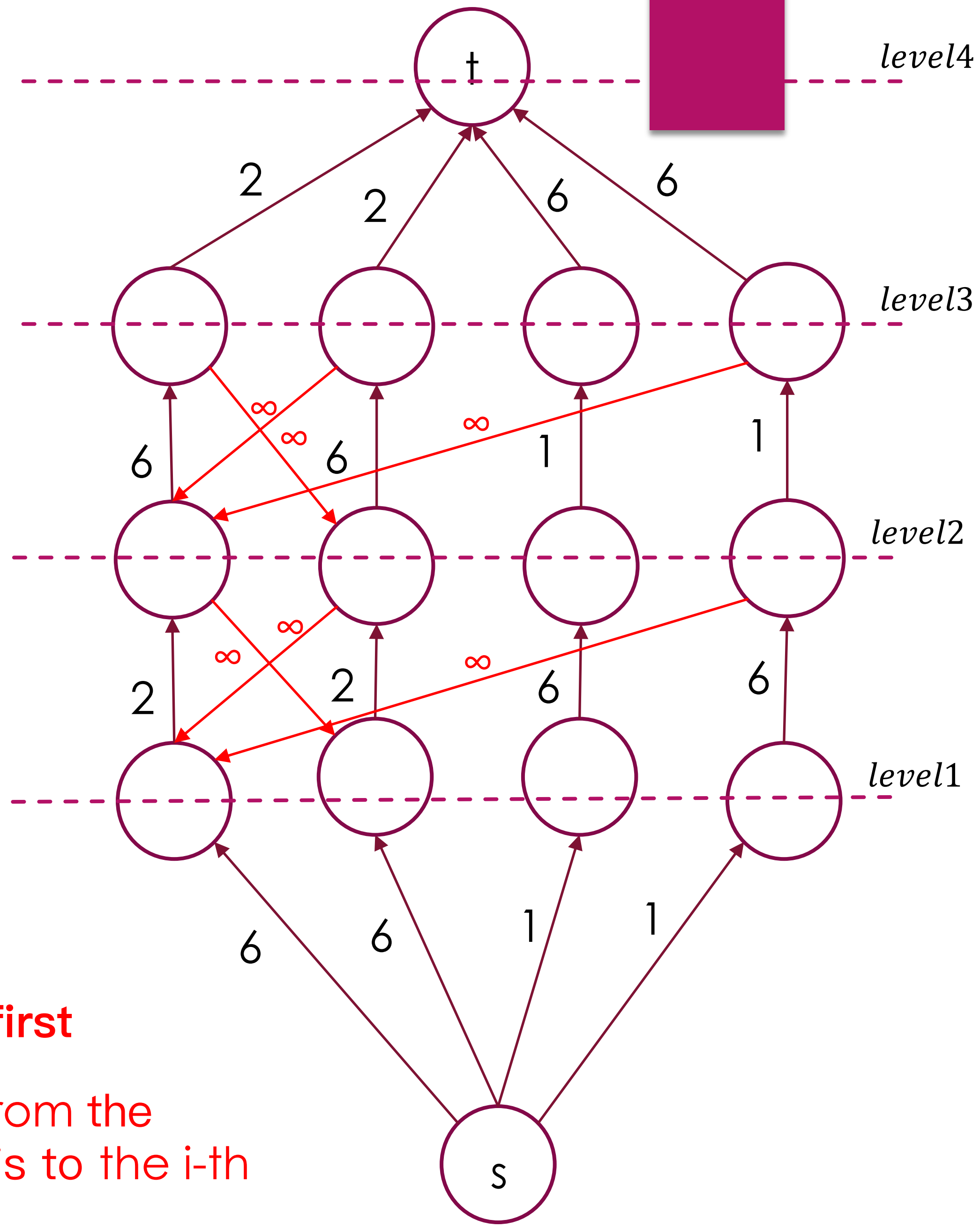




no constraint: $|\boldsymbol{f}(\boldsymbol{x},\boldsymbol{y}) - (\boldsymbol{x}',\boldsymbol{y}')| \le \boldsymbol{D}$

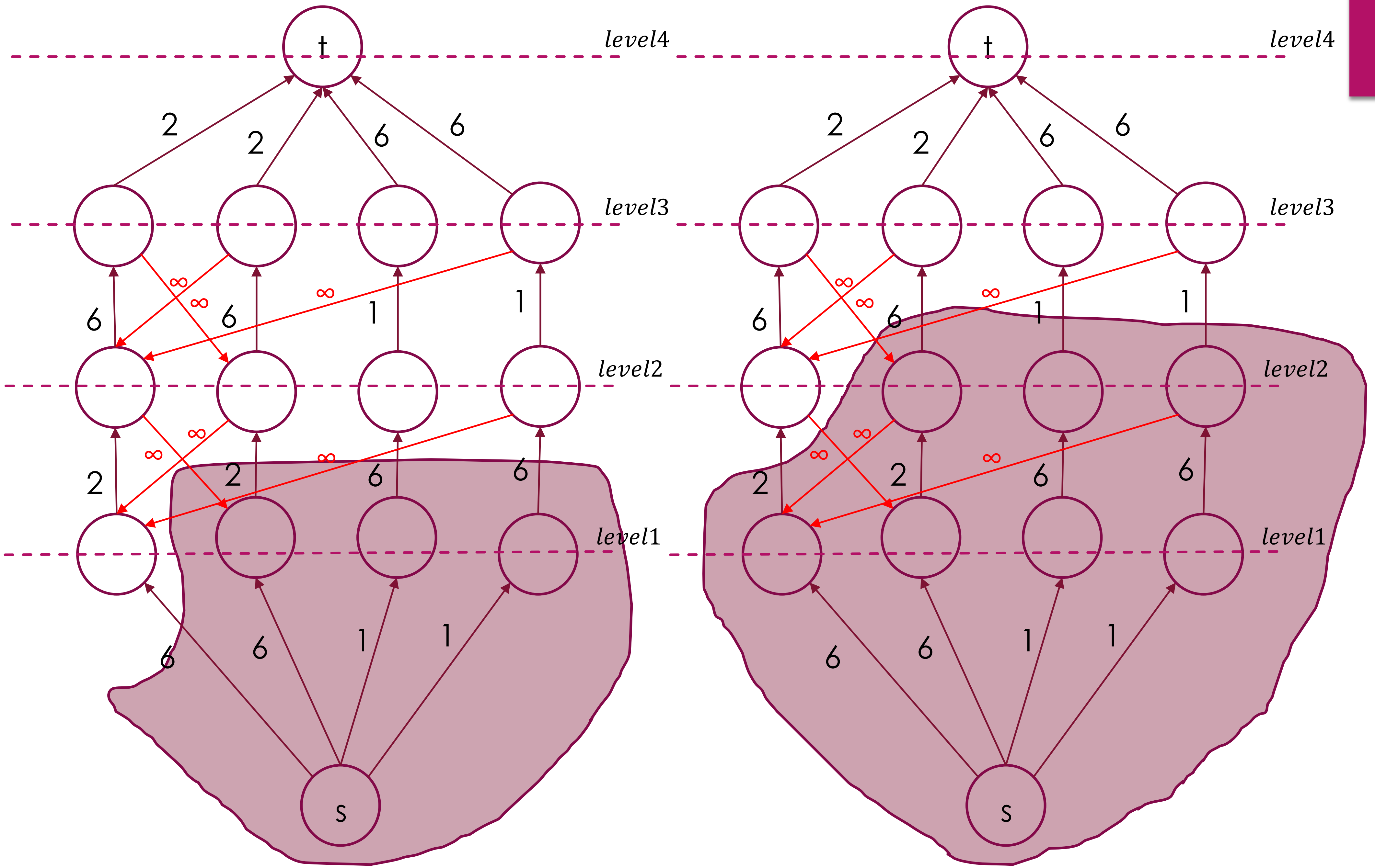




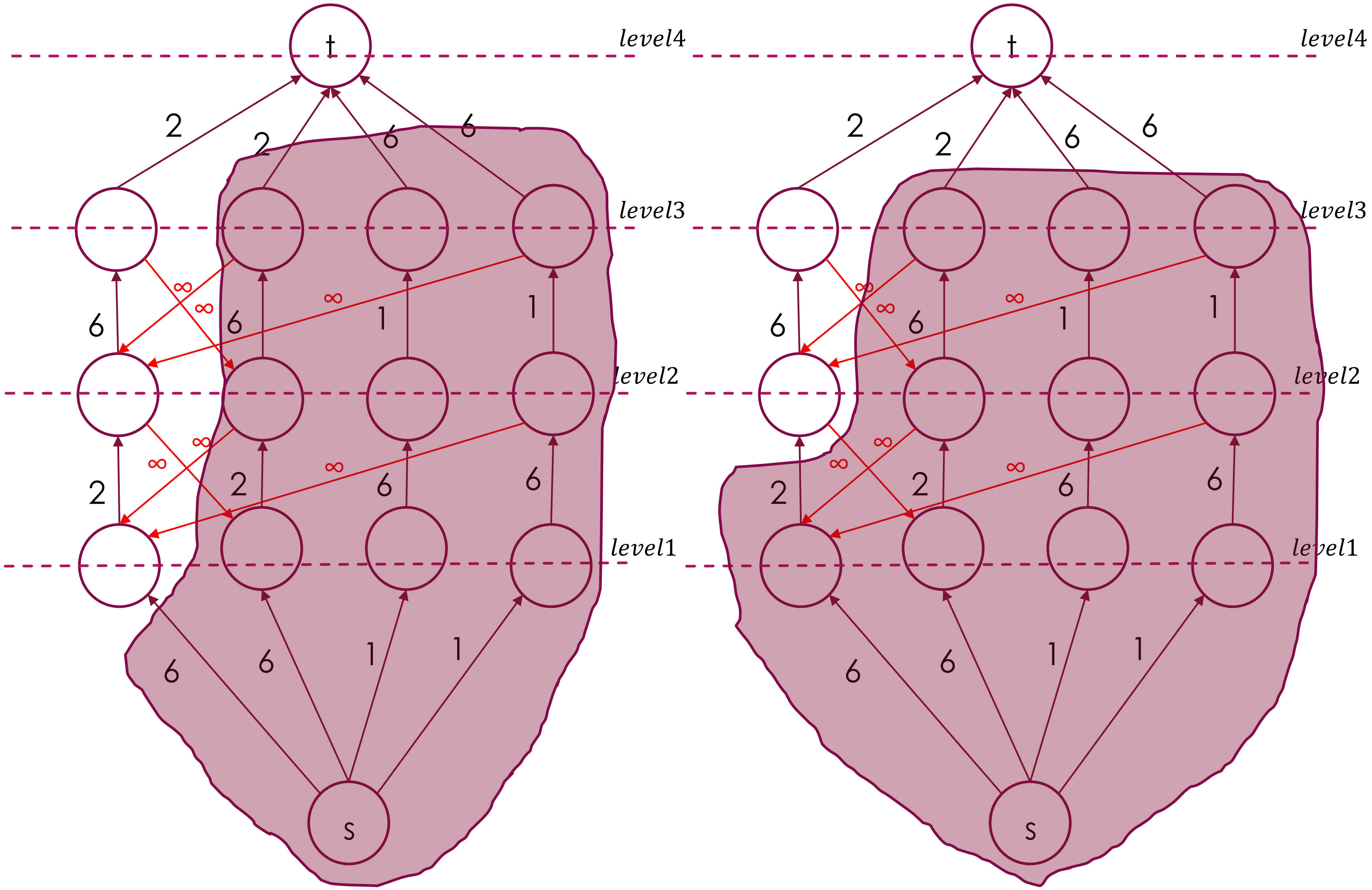


add constraint **Only consider the vertical axis (1,1) first**

Try to add edges with infinite weight from the (i+D)th node in the neighbor vertical axis to the i-th node in one vertical axis.

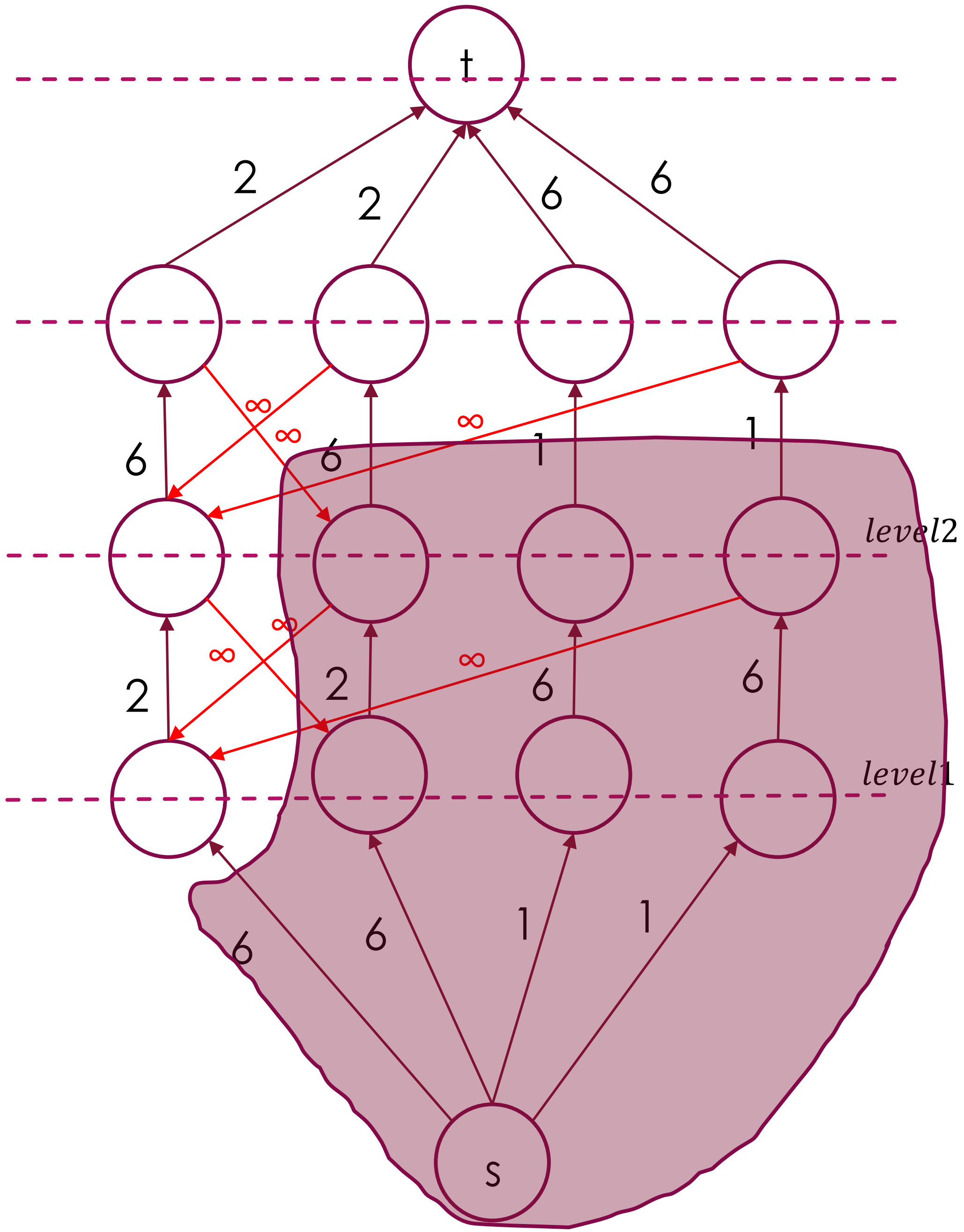


D≤1



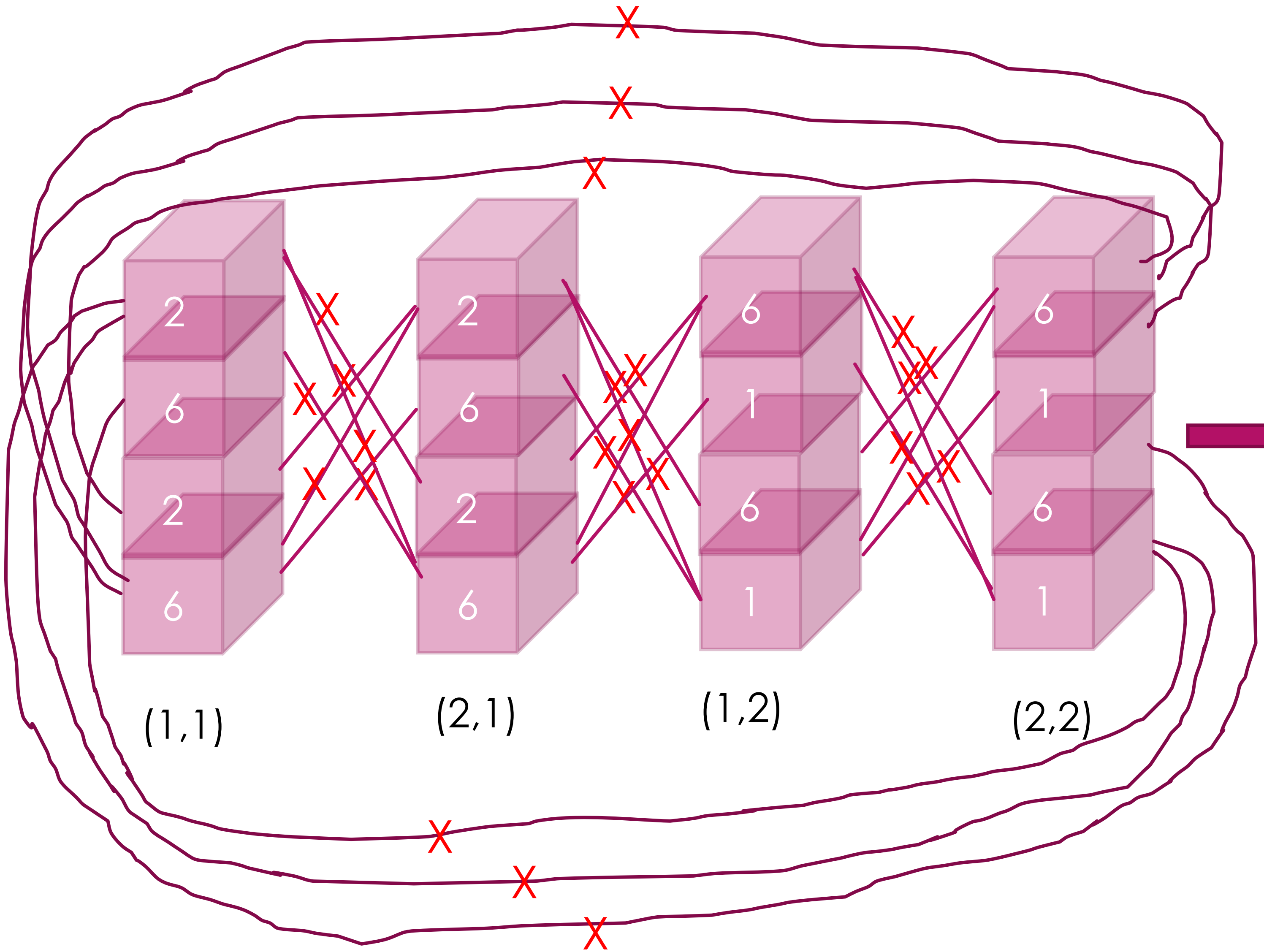
t
level4
level3
level2
level1
s
t
level4
level3
level2
level1
s

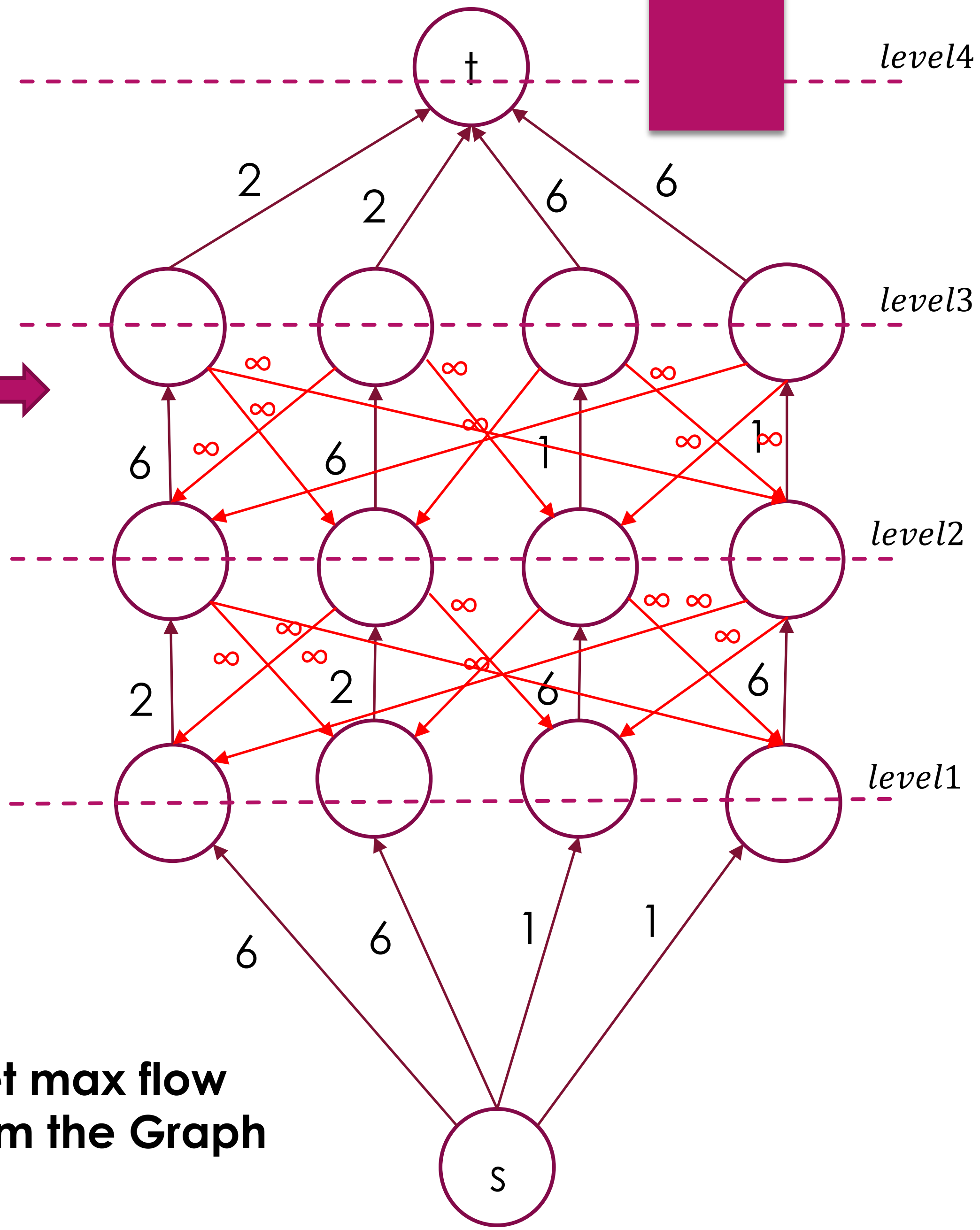
D>1



D>1

The cut which contains an outgoing edge with infinite weight cannot **be the minimum cut.**





add constraint

Try to add edges with infinite weight from the (i+D)th node in the neighbor vertical axis to the i-th node in one vertical axis.

Get max flow from the Graph