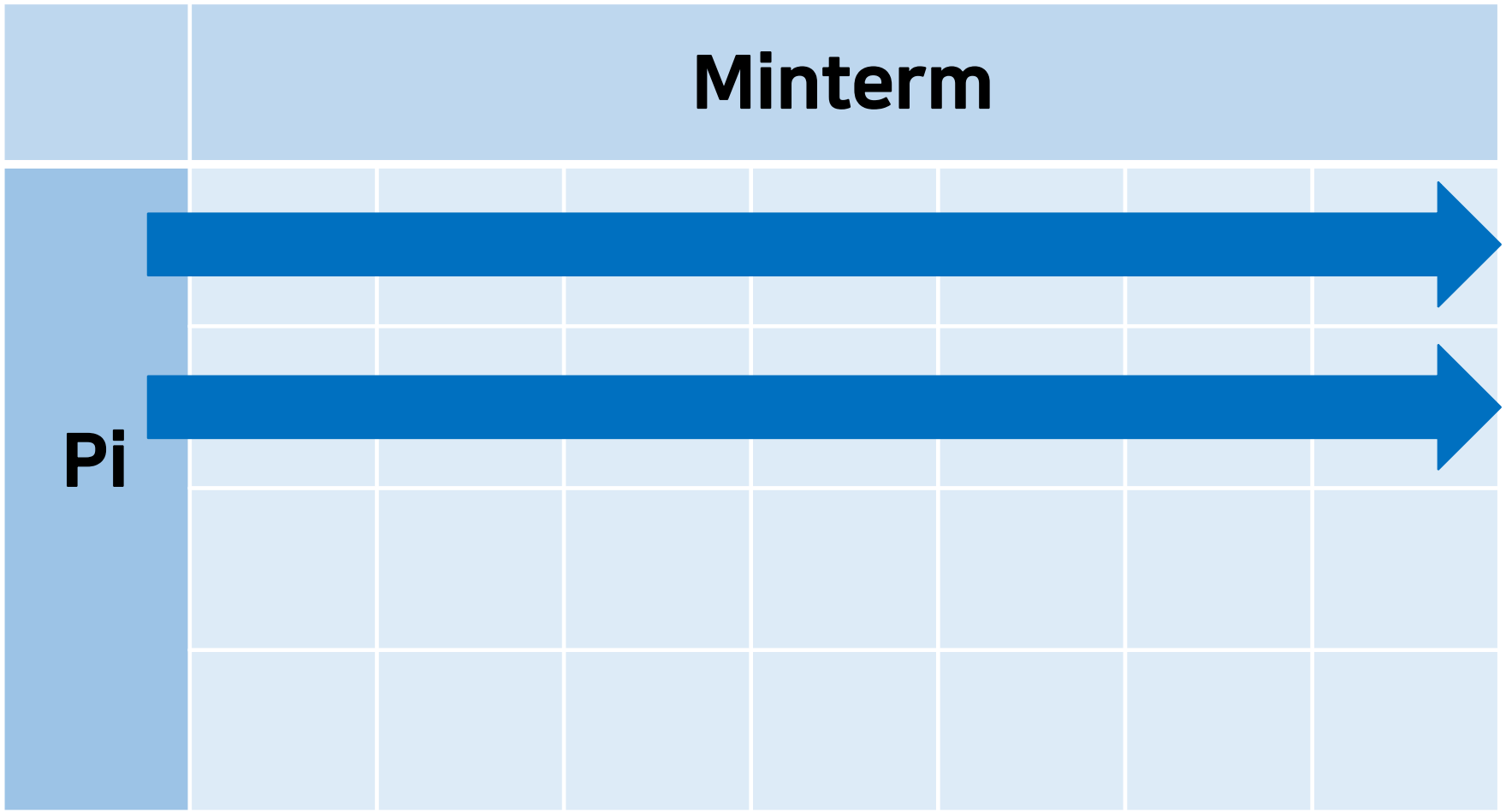
Row & Column Dominance

<rowdic>



PI & EPI

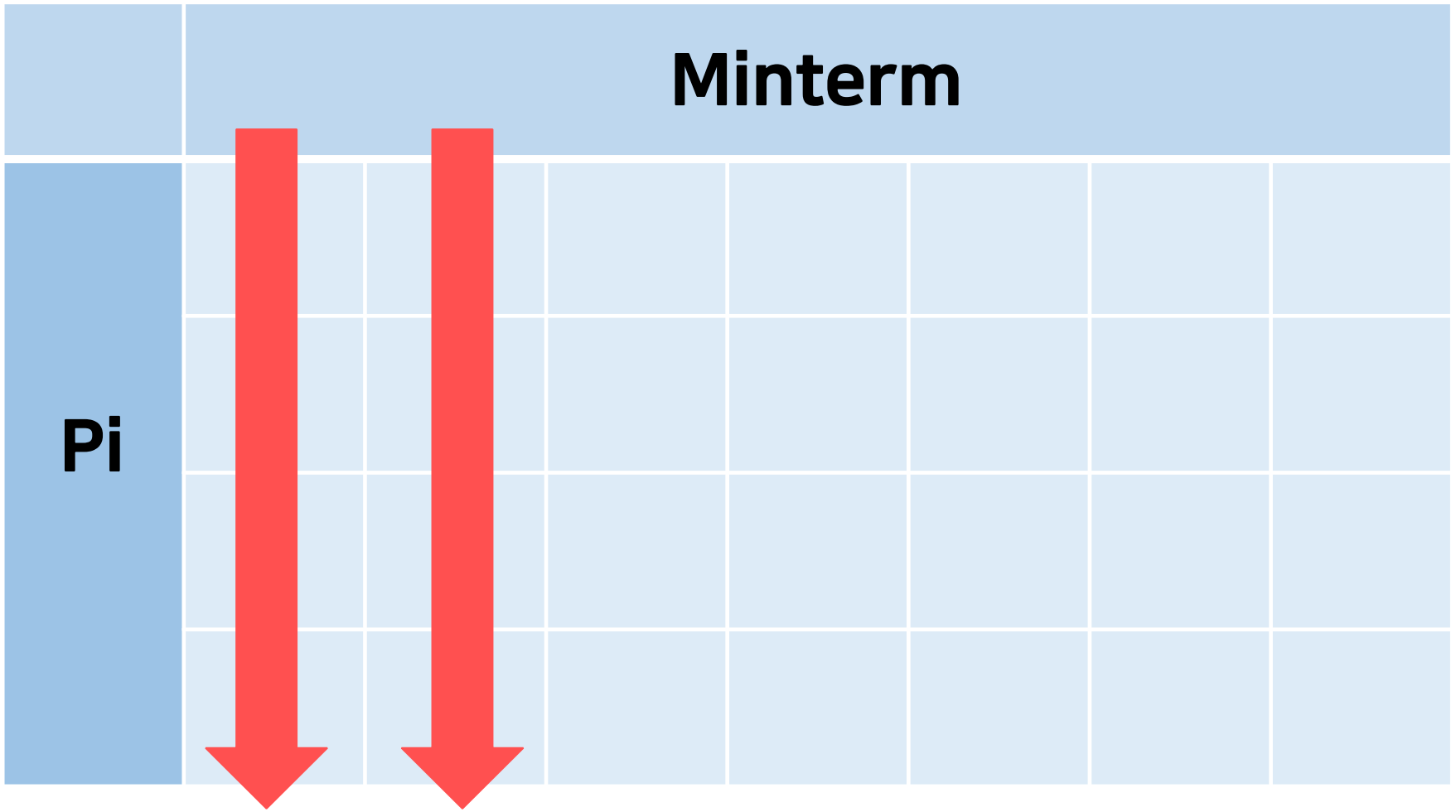
rowdic (Dictionary)

{‘2201’: [5, 13], ‘0022’: [0, 1, 2, 3], ‘0202’: [0,1,4,5]··· }

PI (String)

Minterm (List)

<coldic>



PI & EPI

coldic (Dictionary)

{‘0’: [‘0022’, ‘0202’, ‘0220], ‘1’: [‘0022’, ‘0202’]··· }

Minterm (Int)

PI (String-List)

Column Dominance

Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                print(x, " dominate ", y)
                if((x not in remove_list) & (y not in remove_list)):
                    remove_list.append(x)
            else:
                print(x, " dominate ", y)
                if (x not in remove_list):
                    remove_list.append(x)
```

	Minterm				
PI		V			
		V	V		

Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                print(x, " dominate ", y)
                if((x not in remove_list) & (y not in remove_list)):
                    remove_list.append(x)
            else:
                print(x, " dominate ", y)
                if (x not in remove_list):
                    remove_list.append(x)
```

	Minterm				
PI		V			
		V	V		

Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                print(x, " dominate ", y)
                if((x not in remove_list) & (y not in remove_list)):
                    remove_list.append(x)
            else:
                print(x, " dominate ", y)
                if (x not in remove_list):
                    remove_list.append(x)
```

	Minterm				
PI		V			
		V	V		

Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                print(x, " dominate ", y)
                if((x not in remove_list) & (y not in remove_list)):
                    remove_list.append(x)
            else:
                print(x, " dominate ", y)
                if (x not in remove_list):
                    remove_list.append(x)
```

	Minterm				
PI		V			
		V	V		

Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                print(x, " dominate ", y)
                if((x not in remove_list) & (y not in remove_list)):
                    remove_list.append(x)
            else:
                print(x, " dominate ", y)
                if (x not in remove_list):
                    remove_list.append(x)
```

	Minterm				
PI		V			
		V	V		

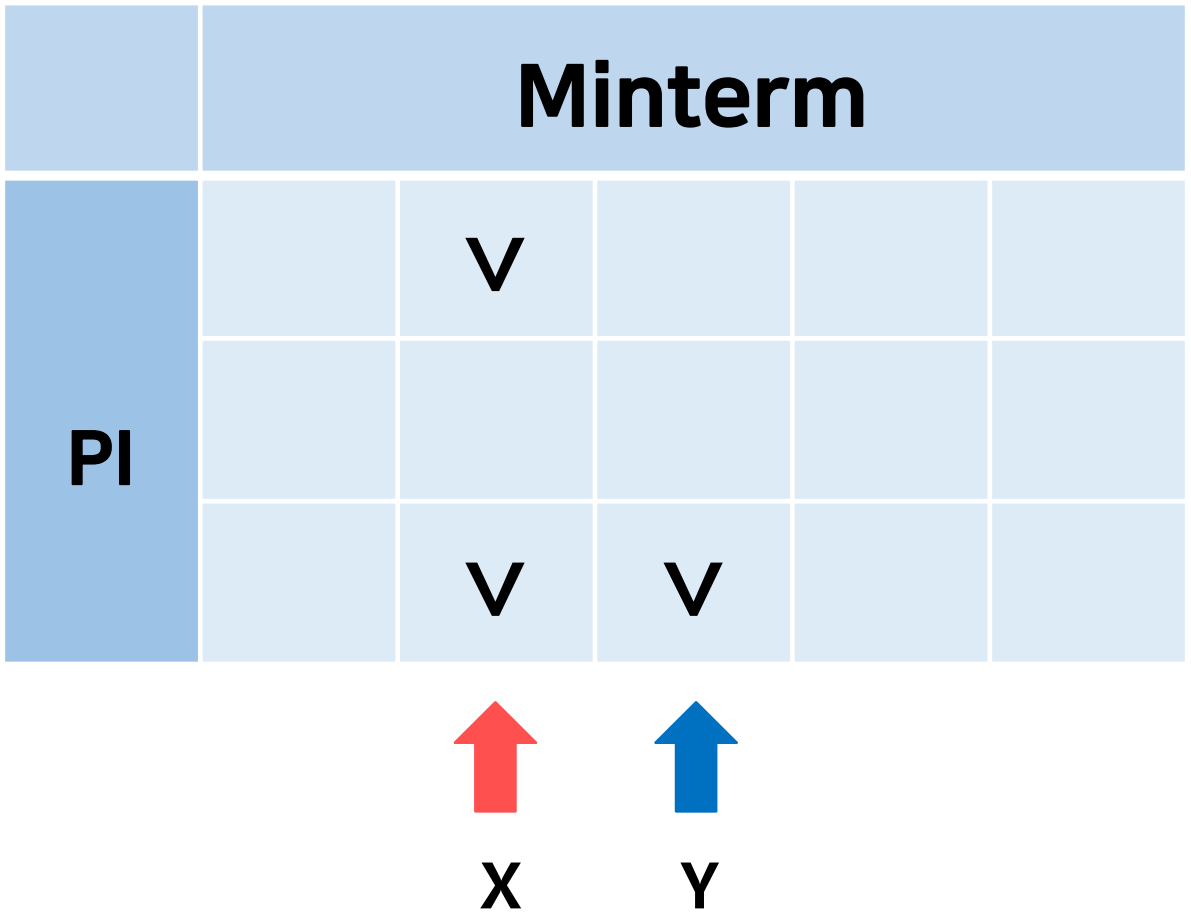
Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                print(x, " dominate ", y)
                if((x not in remove_list) & (y not in remove_list)):
                    remove_list.append(x)
            else:
                print(x, " dominate ", y)
                if (x not in remove_list):
                    remove_list.append(x)
```

	Minterm				
PI		V			
		V	V		

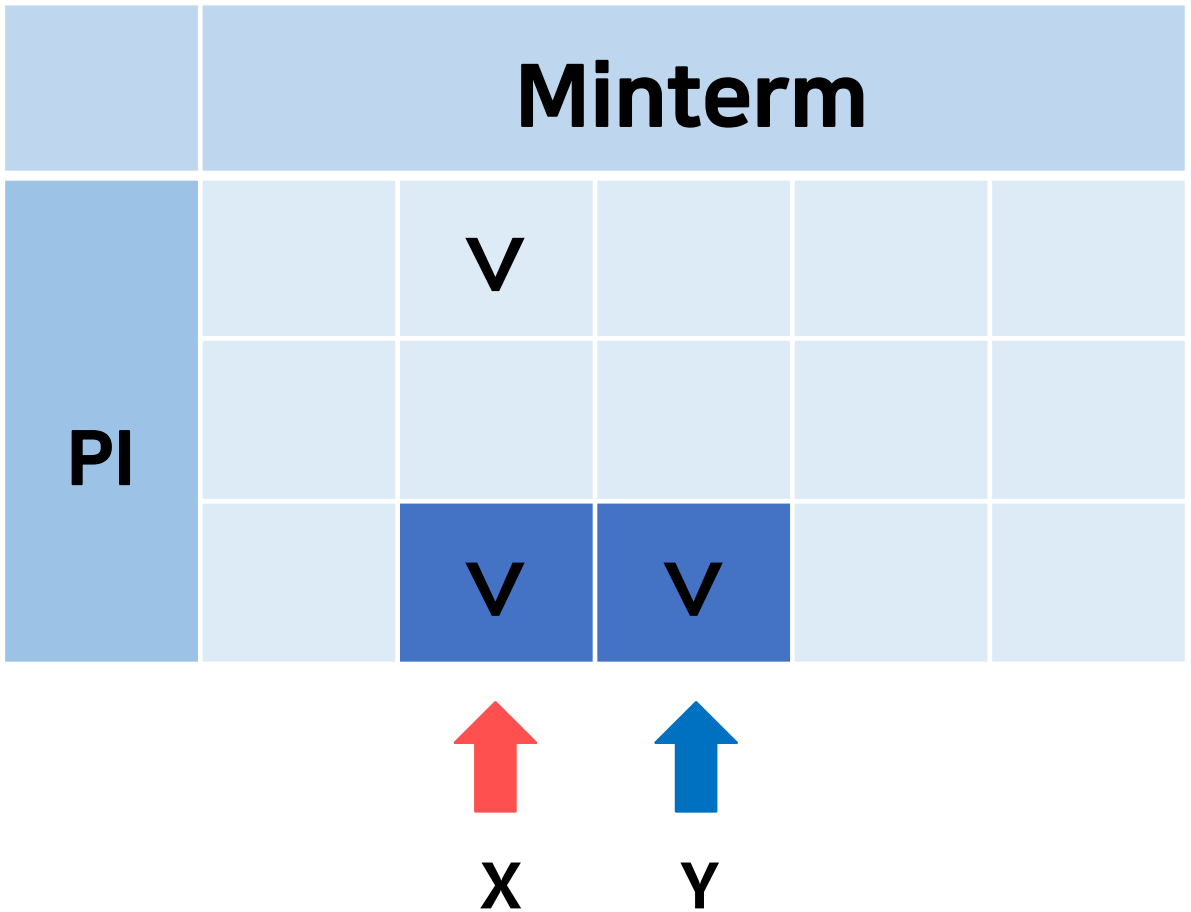
Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if(x!=y):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
                if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(x)
                else:
                    print(x, " dominate ", y)
                    if (x not in remove_list):
                        remove_list.append(x)
```



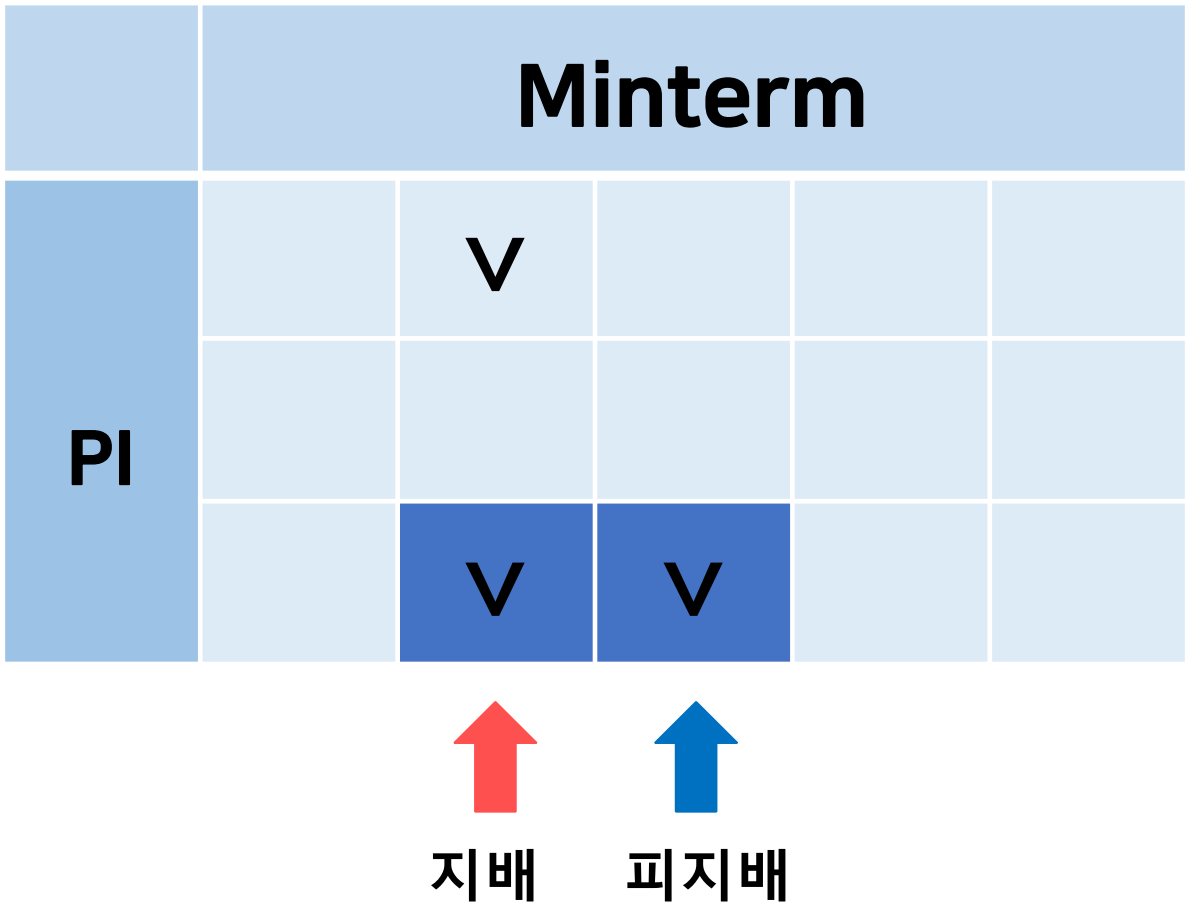
Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if(x!=y):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
                if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(x)
                else:
                    print(x, " dominate ", y)
                    if (x not in remove_list):
                        remove_list.append(x)
```



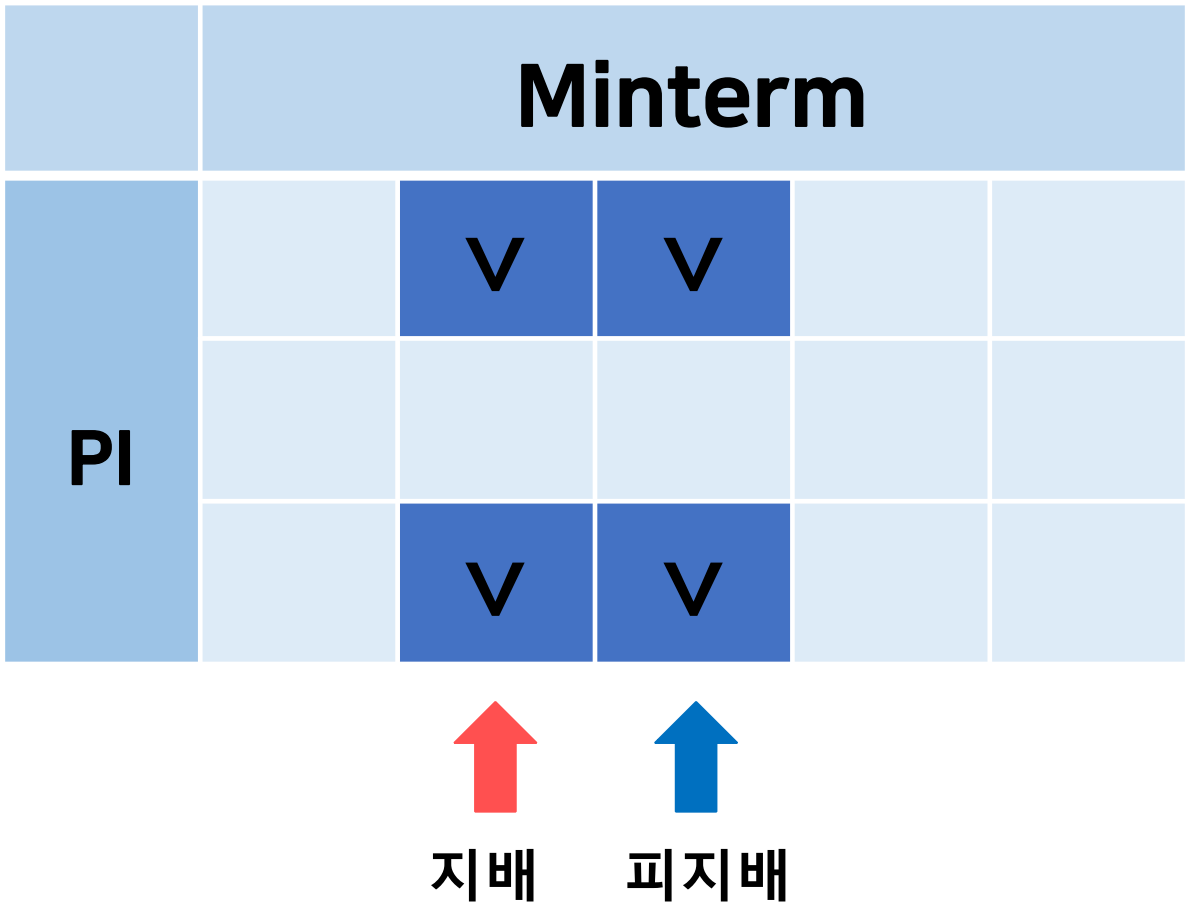
Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if(x!=y):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
                if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(x)
                else:
                    print(x, " dominate ", y)
                    if (x not in remove_list):
                        remove_list.append(x)
```



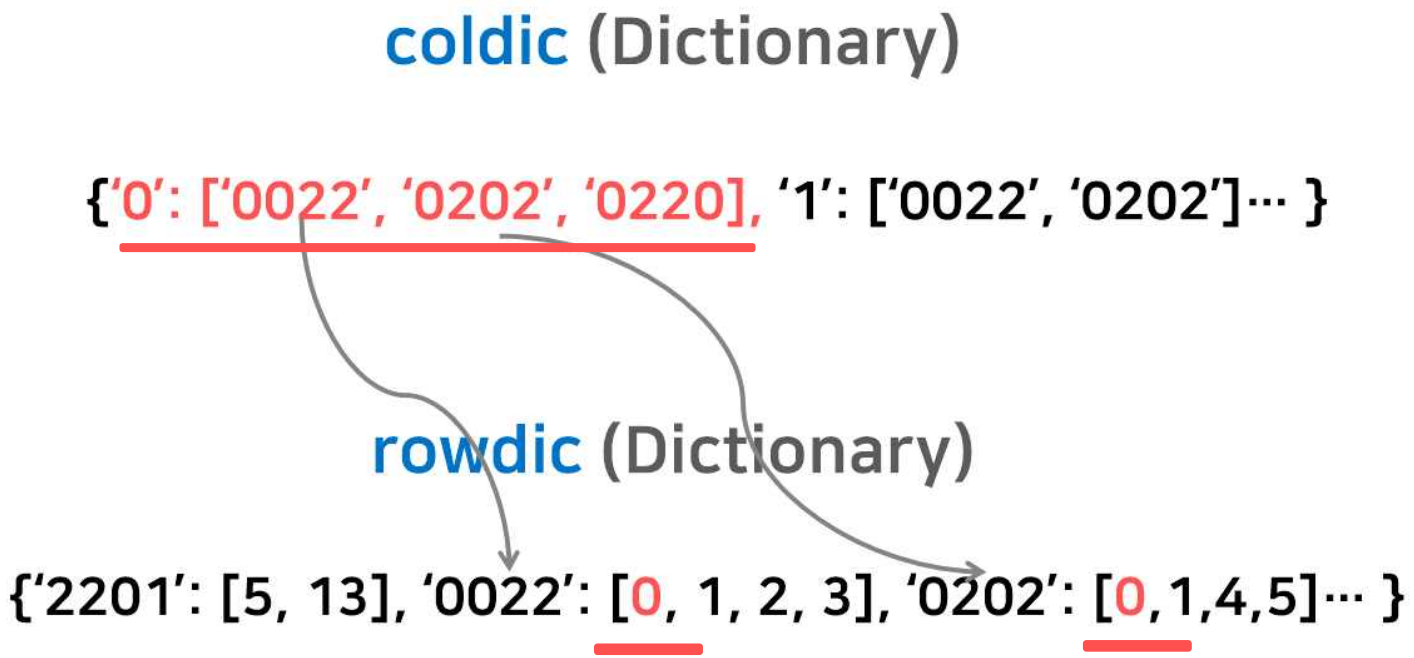
Column Dominance

```
remove_list=[]
for x in coldic:
    for y in coldic:
        if(x!=y):
            if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(y)))):
                if list(sorted(set(coldic.get(x)) & set(coldic.get(y))))==list(sorted(set(coldic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(x)
                else:
                    print(x, " dominate ", y)
                    if (x not in remove_list):
                        remove_list.append(x)
```



Column Dominance

```
coldic_get=[]
for x in remove_list:
    for r in coldic.get(x):
        coldic_get.append(r)
    for y in coldic_get:
        if(x in rowdic.get(y)):
            rowdic.get(y).remove(x)
    coldic_get=[]
    del coldic[x]
print("col_dominance: ", remove_list)
```



Column Dominance

```
coldic_get=[]
for x in remove_list:
    for r in coldic.get(x):
        coldic_get.append(r)
    for y in coldic_get:
        if(x in rowdic.get(y)):
            rowdic.get(y).remove(x)
    coldic_get=[]
    del coldic[x]
print("col_dominance: ", remove_list)
```

coldic (Dictionary)

{'0': ['0022', '0202', '0220], '1': ['0022', '0202']··· }

rowdic (Dictionary)

{'2201': [5, 13], '0022': [1, 2, 3], '0202': [1,4,5]··· }

Column Dominance

```
coldic_get=[]
for x in remove_list:
    for r in coldic.get(x):
        coldic_get.append(r)
    for y in coldic_get:
        if(x in rowdic.get(y)):
            rowdic.get(y).remove(x)
    coldic_get=[]
    del coldic[x]
print("col_dominance: ", remove_list)
```

coldic (Dictionary)

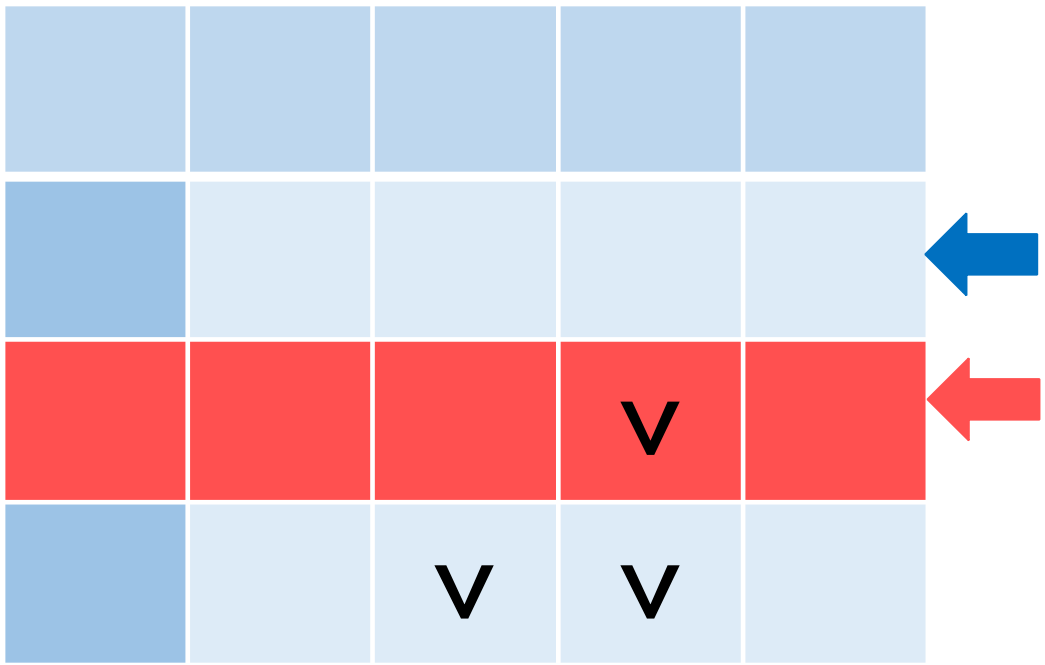
{'1': ['0022', '0202']··· }

rowdic (Dictionary)

{'2201': [5, 13], '0022': [1, 2, 3], '0202': [1,4,5]··· }

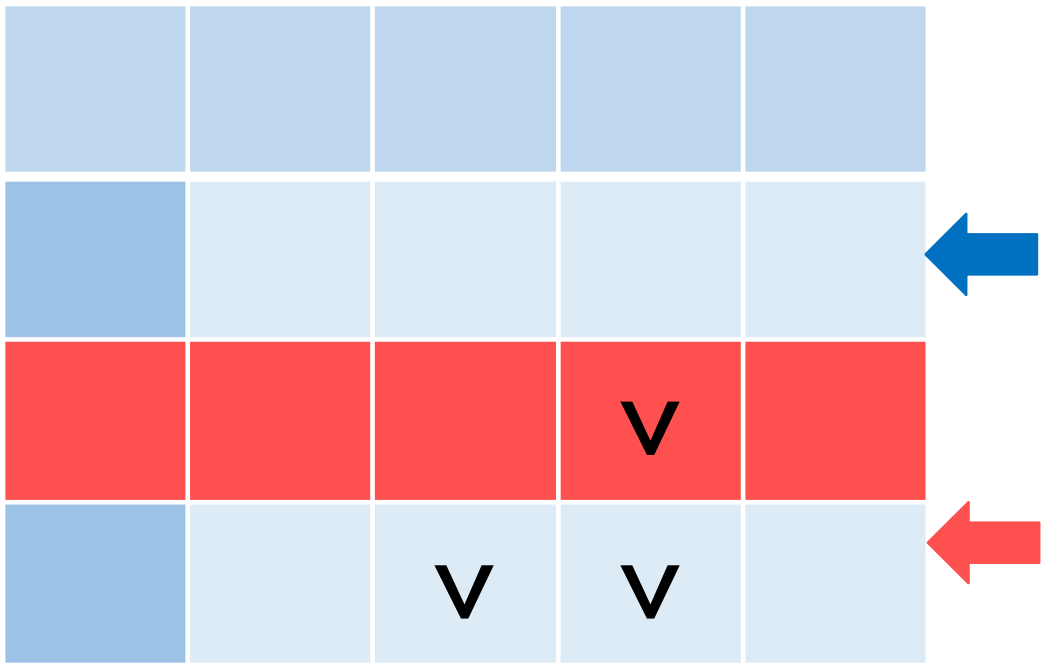
Row Dominance

```
remove_list=[]
for x in rowdic:
    for y in rowdic:
        if(x!=y)
            if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(y)))):
                if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(y)
                else:
                    if( y not in remove_list):
                        remove_list.append(y)
                        print(x, " dominate ", y)
```



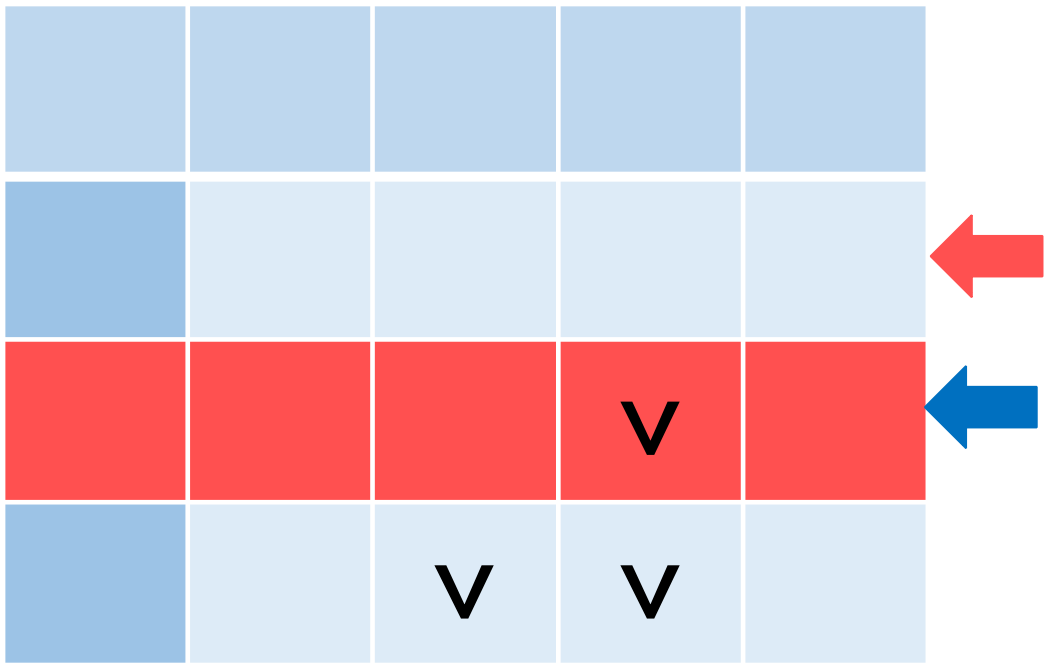
Row Dominance

```
remove_list=[]
for x in rowdic:
    for y in rowdic:
        if(x!=y)
            if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(y)))):
                if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(y)
                else:
                    if( y not in remove_list):
                        remove_list.append(y)
                        print(x, " dominate ", y)
```



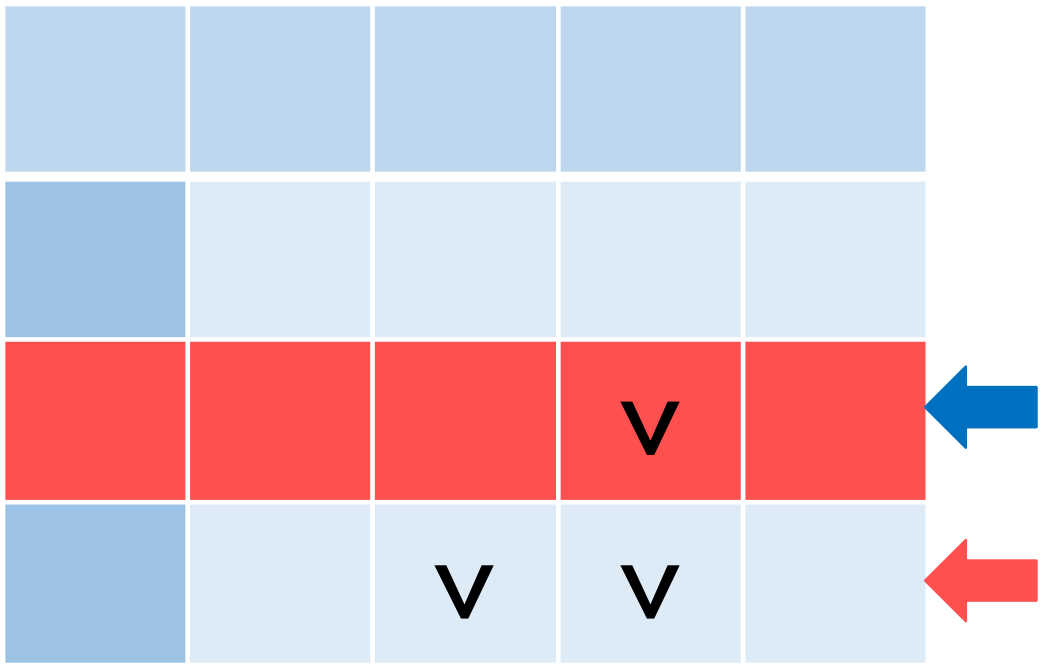
Row Dominance

```
remove_list=[]
for x in rowdic:
    for y in rowdic:
        if(x!=y)
            if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(y)))):
                if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(y)
                else:
                    if( y not in remove_list):
                        remove_list.append(y)
                        print(x, " dominate ", y)
```



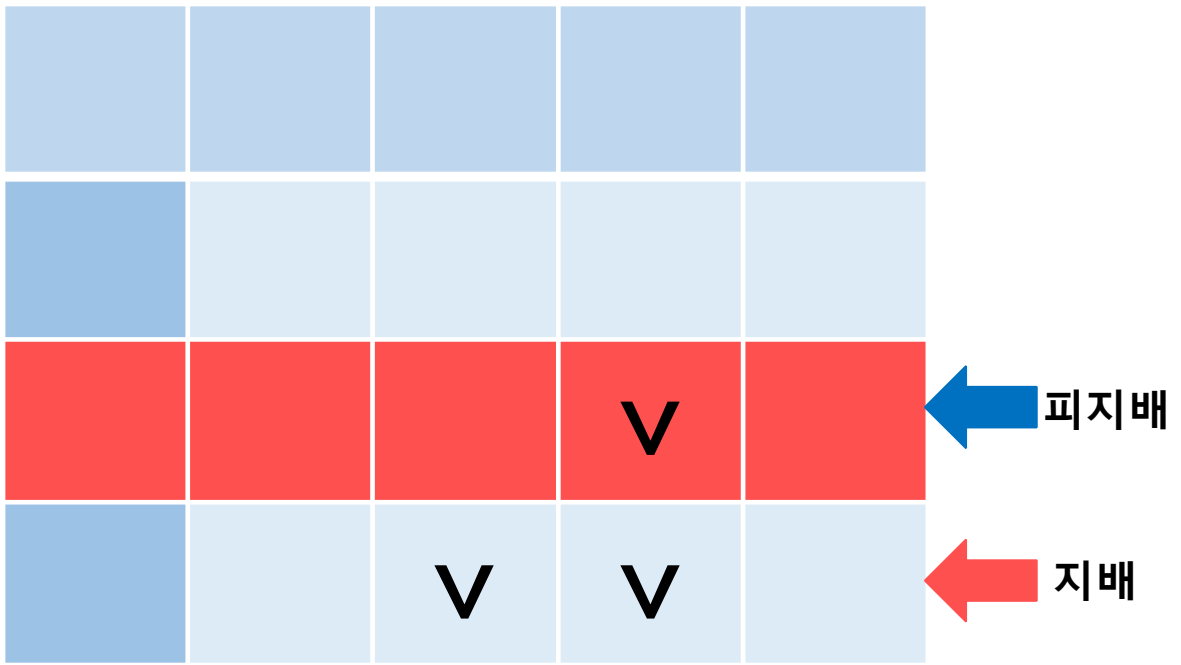
Row Dominance

```
remove_list=[]
for x in rowdic:
    for y in rowdic:
        if(x!=y)
            if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(y)))):
                if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(y)
                else:
                    if( y not in remove_list):
                        remove_list.append(y)
                        print(x, " dominate ", y)
```



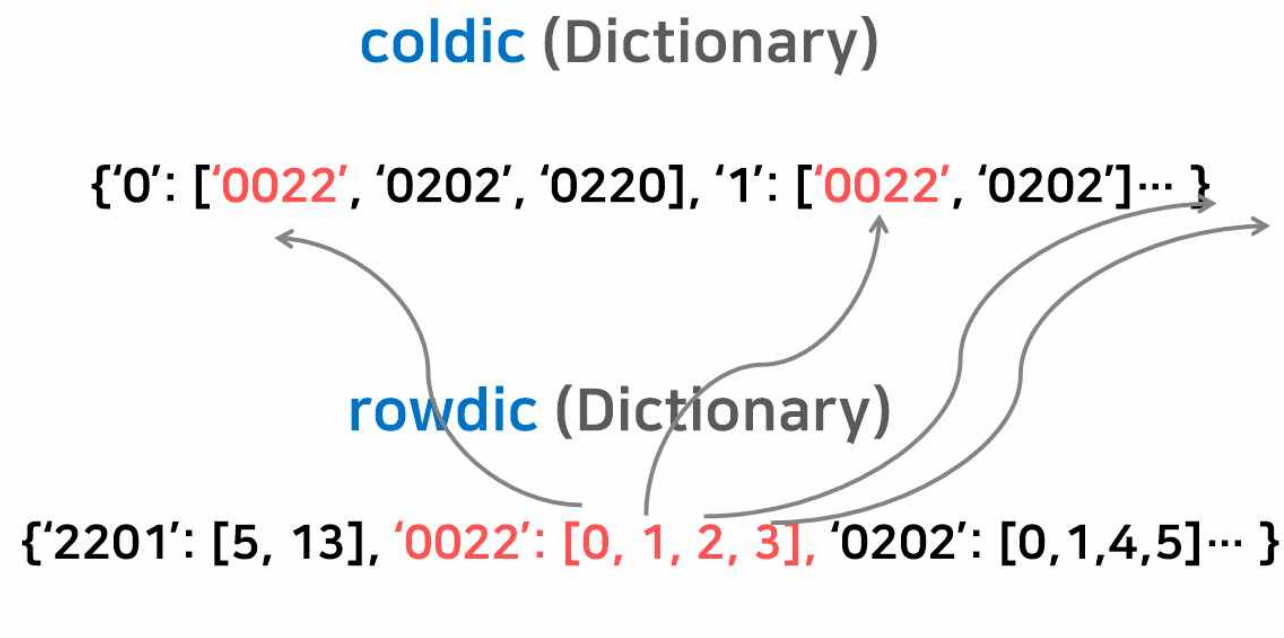
Row Dominance

```
remove_list=[]
for x in rowdic:
    for y in rowdic:
        if(x!=y)
            if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(y)))):
                if list(sorted(set(rowdic.get(x))&set(rowdic.get(y))))==list(sorted(set(rowdic.get(x)))):
                    print(x, " dominate ", y)
                    if((x not in remove_list) & (y not in remove_list)):
                        remove_list.append(y)
                else:
                    if( y not in remove_list):
                        remove_list.append(y)
                        print(x, " dominate ", y)
```



Row Dominance

```
rowdic_get=[]
for x in remove_list:
    for r in rowdic.get(x):
        rowdic_get.append(r)
    for y in rowdic_get:
        if(x in coldic.get(y)):
            coldic.get(y).remove(x)
    rowdic_get=[]
    del rowdic[x]
```



Row Dominance raw

```
rowdic_get=[]
for x in remove_list:
    for r in rowdic.get(x):
        rowdic_get.append(r)
    for y in rowdic_get:
        if(x in coldic.get(y)):
            coldic.get(y).remove(x)
    rowdic_get=[]
    del rowdic[x]
```

coldic (Dictionary)

{'0': ['0202', '0220], '1': ['0202']··· }

rowdic (Dictionary)

{'2201': [5, 13], '0022': [0, 1, 2, 3], '0202': [0,1,4,5]··· }

Row Dominance

```
rowdic_get=[]
for x in remove_list:
    for r in rowdic.get(x):
        rowdic_get.append(r)
    for y in rowdic_get:
        if(x in coldic.get(y)):
            coldic.get(y).remove(x)
    rowdic_get=[]
    del rowdic[x]
```

coldic (Dictionary)

{'0': ['0202', '0220], '1': ['0202']··· }

rowdic (Dictionary)

{'2201': [5, 13], '0202': [0,1,4,5]··· }

QM Method

1. Find all PIs to construct a PI table
2. Find EPIs to simplify the table

   If (no NEPI remained) -> then quit
3. Apply column dominance raw
4. Apply row dominance raw

   If (any simplification made)

Run!

minterm=[4,11,0,1,2,3,4,5,6,10,11,13,14]

	0	1	2	3	4	5	6	10	11	13	14
0022	v	v	v	v							
0202	v	v			v	v					
0220	v		v		v		v				
2012			v	v				v	v		
2201						v				v	
2210			v				v	v			v

Run!

```
<pi>
 {'0022': [0, 1, 2, 3], '0202': [0, 1, 4, 5], '0220': [0, 2, 4, 6], '2012': [2, 3, 10, 11], '2101': [5, 13], '2210': [2, 6, 10, 14]}


<rowdic>
 {'0022': [0, 1, 2, 3], '0202': [0, 1, 4, 5], '0220': [0, 2, 4, 6], '2012': [2, 3, 10, 11], '2101': [5, 13], '2210': [2, 6, 10, 14]}


<coldic>
 {0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 2: ['0022', '0220', '2012', '2210'], 3: ['0022', '2012'], 4: ['0202', '0220'], 5: ['2101', '0202'],
 6: ['0220', '2210'], 10: ['2012', '2210'], 11: ['2012'], 13: ['2101'], 14: ['2210']}
```

EPI

epi_list=['2012', '2101', '2210']

	0	1	2	3	4	5	6	10	11	13	14
0022	v	v	v	v							
0202	v	v			v	v					
0220	v		v		v		v				
2012			v	v				v	v		
2101						v				v	
2210			v				v	v			v

EPI

NEPI (o)

rowdic={'0022': [0, 1], '0202': [0, 1, 4], '0220': [0, 4]}

coldic={0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 4: ['0202', '0220']}

	0	1	4
0022	v	v	
0202	v	v	v
0220	v		v

첫 번째 순환

```
loop_count:  1
<epi_list>
['2012', '2101', '2210']


<rowdic>
 {'0022': [0, 1], '0202': [0, 1, 4], '0220': [0, 4]}


<coldic>
 {0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 4: ['0202', '0220']}


0  dominate  1
0  dominate  4
col_remove:  [0]


<rowdic>
 {'0022': [1], '0202': [1, 4], '0220': [4]}


<coldic>
 {1: ['0022', '0202'], 4: ['0202', '0220']}


0202  dominate  0022
0202  dominate  0220
row_remove:  ['0022', '0220']


<rowdic>
 {'0202': [1, 4]}


<coldic>
 {1: ['0202'], 4: ['0202']}
```

Column Dominance

0 dominance 1
0 dominance 4
col_remove : [0]

	0	1	4
0022	v	v	
0202	v	v	v
0220	v		v

첫 번째 순환

```
loop_count:  1
<epi_list>
['2012', '2101', '2210']


<rowdic>
 {'0022': [0, 1], '0202': [0, 1, 4], '0220': [0, 4]}


<coldic>
 {0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 4: ['0202', '0220']}


0  dominate  1
0  dominate  4
col_remove:  [0]


<rowdic>
 {'0022': [1], '0202': [1, 4], '0220': [4]}


<coldic>
 {1: ['0022', '0202'], 4: ['0202', '0220']}


0202  dominate  0022
0202  dominate  0220
row_remove:  ['0022', '0220']


<rowdic>
 {'0202': [1, 4]}


<coldic>
 {1: ['0202'], 4: ['0202']}
```

Column Dominance

rowdic={'0022': [1], '0202': [1, 4], '0220': [4]}

coldic={1: ['0022', '0202'], 4: ['0202', '0220']}

	1	4
0022	v	
0202	v	v
0220		v

첫 번째 순환

```
loop_count:  1
<epi_list>
['2012', '2101', '2210']


<rowdic>
 {'0022': [0, 1], '0202': [0, 1, 4], '0220': [0, 4]}


<coldic>
 {0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 4: ['0202', '0220']}


0  dominate  1
0  dominate  4
col_remove:  [0]


<rowdic>
 {'0022': [1], '0202': [1, 4], '0220': [4]}


<coldic>
 {1: ['0022', '0202'], 4: ['0202', '0220']}


0202  dominate  0022
0202  dominate  0220
row_remove:  ['0022', '0220']


<rowdic>
 {'0202': [1, 4]}


<coldic>
 {1: ['0202'], 4: ['0202']}
```

RowDominance

0202 dominance 0022
0202 dominance 0220
col_remove : [’0022’, ‘0220’]

	1	4
0022	v	
0202	v	v
0220		v

첫 번째 순환

```
loop_count:  1
<epi_list>
['2012', '2101', '2210']


<rowdic>
 {'0022': [0, 1], '0202': [0, 1, 4], '0220': [0, 4]}


<coldic>
 {0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 4: ['0202', '0220']}


0  dominate  1
0  dominate  4
col_remove:  [0]


<rowdic>
 {'0022': [1], '0202': [1, 4], '0220': [4]}


<coldic>
 {1: ['0022', '0202'], 4: ['0202', '0220']}


0202  dominate  0022
0202  dominate  0220
row_remove:  ['0022', '0220']


<rowdic>
 {'0202': [1, 4]}


<coldic>
 {1: ['0202'], 4: ['0202']}
```

RowDominance

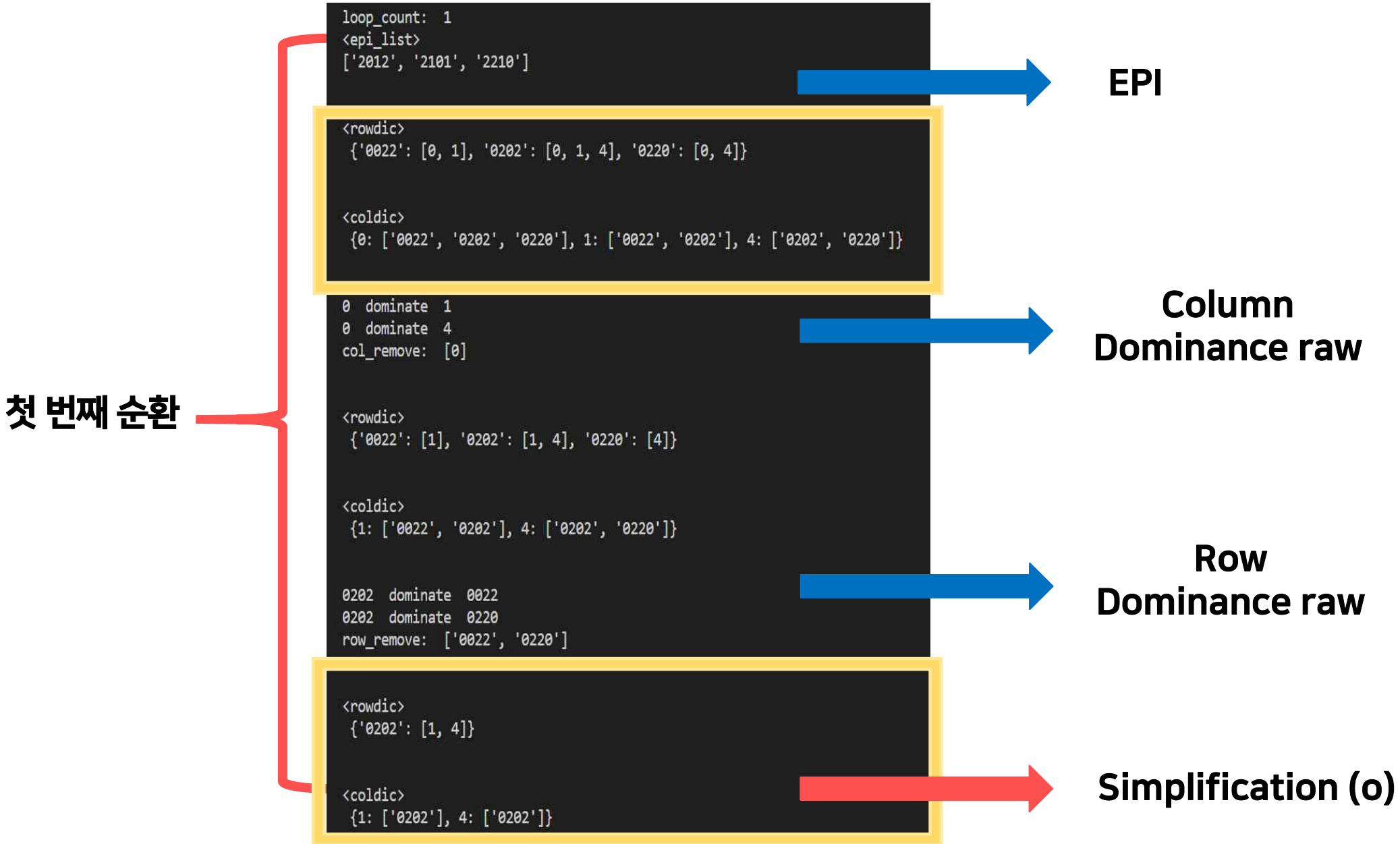
rowdic={‘0202’: [1, 4]}

coldic={1: [‘0202’], 4: [‘0202’]}

	1	4
0202	v	v

첫 번째 순환

```
loop_count:  1
<epi_list>
['2012', '2101', '2210']


<rowdic>
 {'0022': [0, 1], '0202': [0, 1, 4], '0220': [0, 4]}


<coldic>
 {0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 4: ['0202', '0220']}


0  dominate  1
0  dominate  4
col_remove:  [0]


<rowdic>
 {'0022': [1], '0202': [1, 4], '0220': [4]}


<coldic>
 {1: ['0022', '0202'], 4: ['0202', '0220']}


0202  dominate  0022
0202  dominate  0220
row_remove:  ['0022', '0220']


<rowdic>
 {'0202': [1, 4]}


<coldic>
 {1: ['0202'], 4: ['0202']}
```


첫 번째 순환
loop_count: 1
<epi_list>
['2012', '2101', '2210']
EPI
<rowdic>
{'0022': [0, 1], '0202': [0, 1, 4], '0220': [0, 4]}
<coldic>
{0: ['0022', '0202', '0220'], 1: ['0022', '0202'], 4: ['0202', '0220']}
0 dominate 1
0 dominate 4
col_remove: [0]
Column
Dominance raw
<rowdic>
{'0022': [1], '0202': [1, 4], '0220': [4]}
<coldic>
{1: ['0022', '0202'], 4: ['0202', '0220']}
0202 dominate 0022
0202 dominate 0220
row_remove: ['0022', '0220']
Row
Dominance raw
<rowdic>
{'0202': [1, 4]}
<coldic>
{1: ['0202'], 4: ['0202']}
Simplification (o)

EPI

epi_list=['0202']

	1	4
0202	v	v

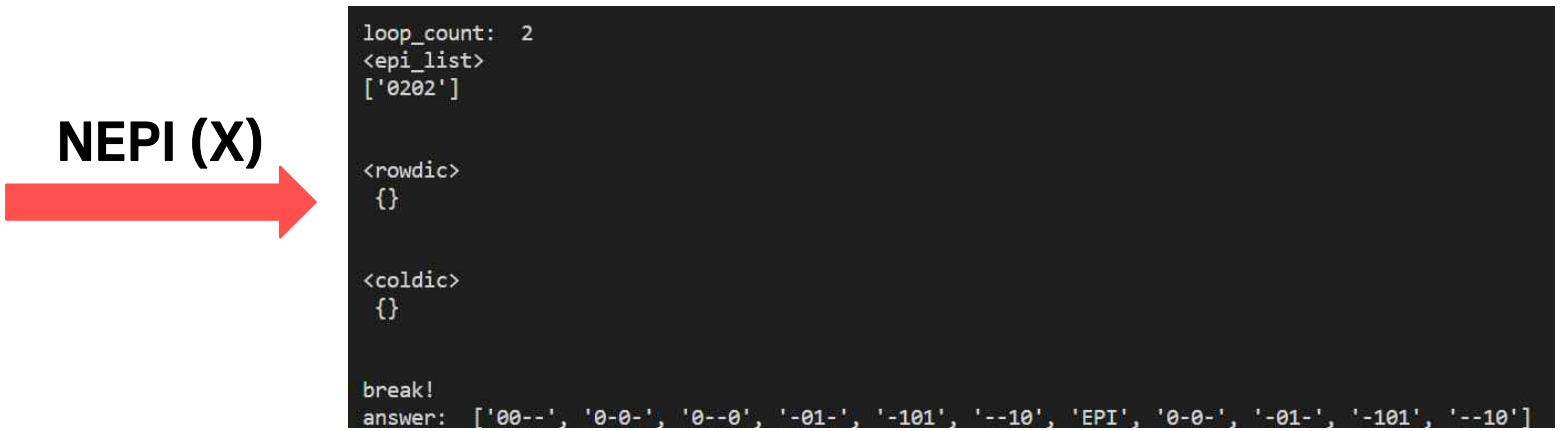
```
loop_count:  2
<epi_list>
['0202']


<rowdic>
 {}


<coldic>
 {}


break!
answer:  ['00--', '0-0-', '0--0', '-01-', '-101', '--10', 'EPI', '0-0-', '-01-', '-101', '--10']
```

EPI

rowdic={}

coldic={}

NEPI (X)



Done!


두 번째 순환
loop_count: 2
<epi_list>
['0202']
<rowdic>
{}
<coldic>
{}
break!
answer: ['00--', '0-0-', '0--0', '-01-', '-101', '--10', 'EPI', '0-0-', '-01-', '-101', '--10']
EPI
NEPI (X)
break
최종 결과