$n = 14$

1 2 3 4 5 6 7 8 9 10 11 12 13 14

1	2	3	4	5	6	7	8	9	100	110	120
10	20	30	40						101	111	
11	21	31	41						102	112	
12	22	32	42						103		
13	23	33							.	.	
14	24	34							.	.	
15	25	35							.	.	
16	26	36							.	.	
17	27	37							.	.	
18	28	38	49	59	67	79	89	91	109	119	
19	29	39									

100 →

Num => 20
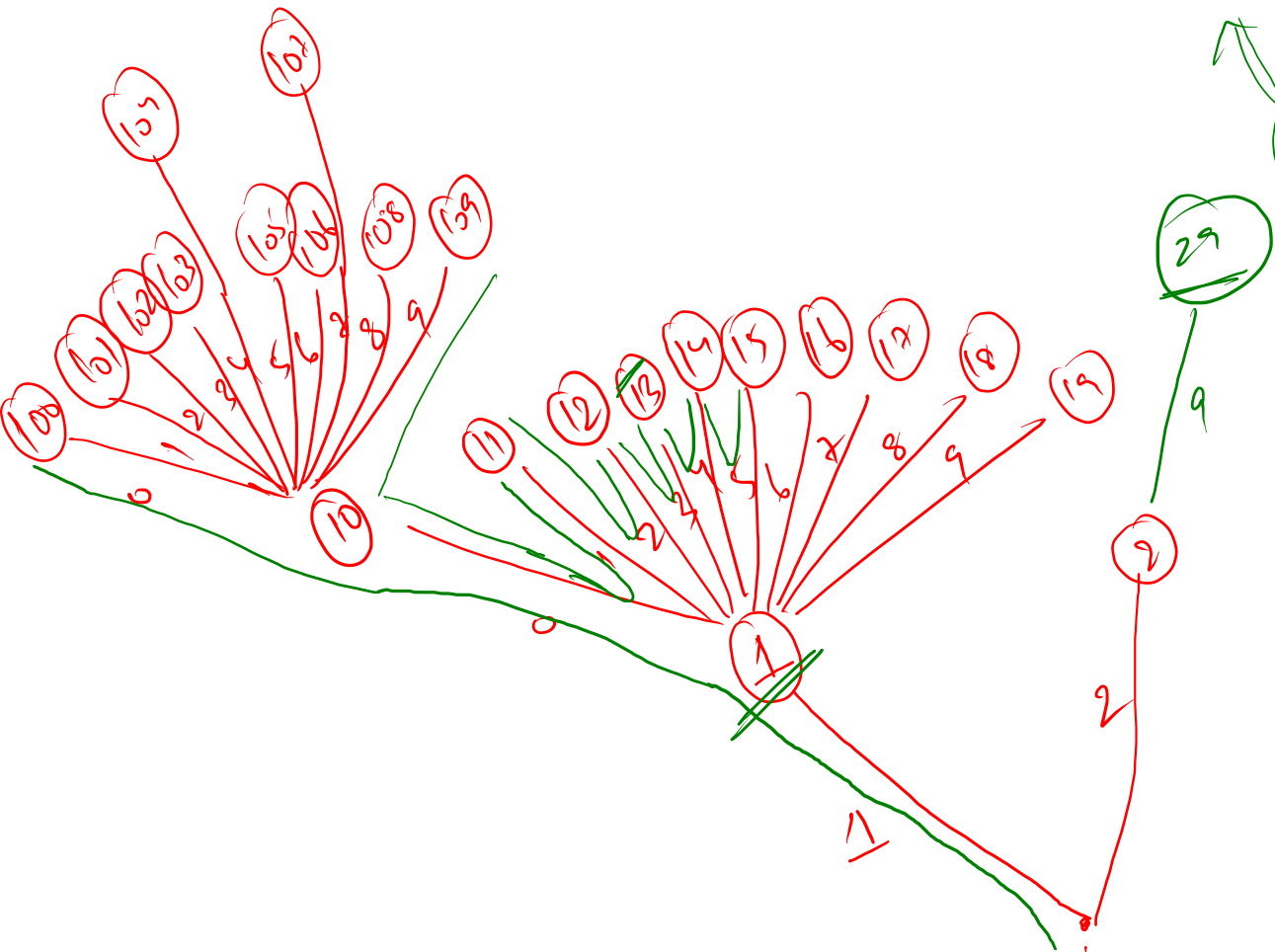
ans = 1

$(ans * 10) + digit;$

Try

ans = 2

$(2 \times 10) + 9$

29

1
10
11
12
13
14
15

$n = 14$

$1, 10, 11, 12, 13, 14, 2$

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();

    for(int i = 1 ; i <= 9 ; i++){
        solve(i,n);
    }
}

public static void solve(int num , int n){
    if(num > n){
        return;
    }

    System.out.println(num);

    for(int digit = 0 ; digit <= 9 ; digit++){
        int newNum = (num*10)+digit;
        if(newNum <= n){
            solve(newNum,n);
        }
    }
}
```
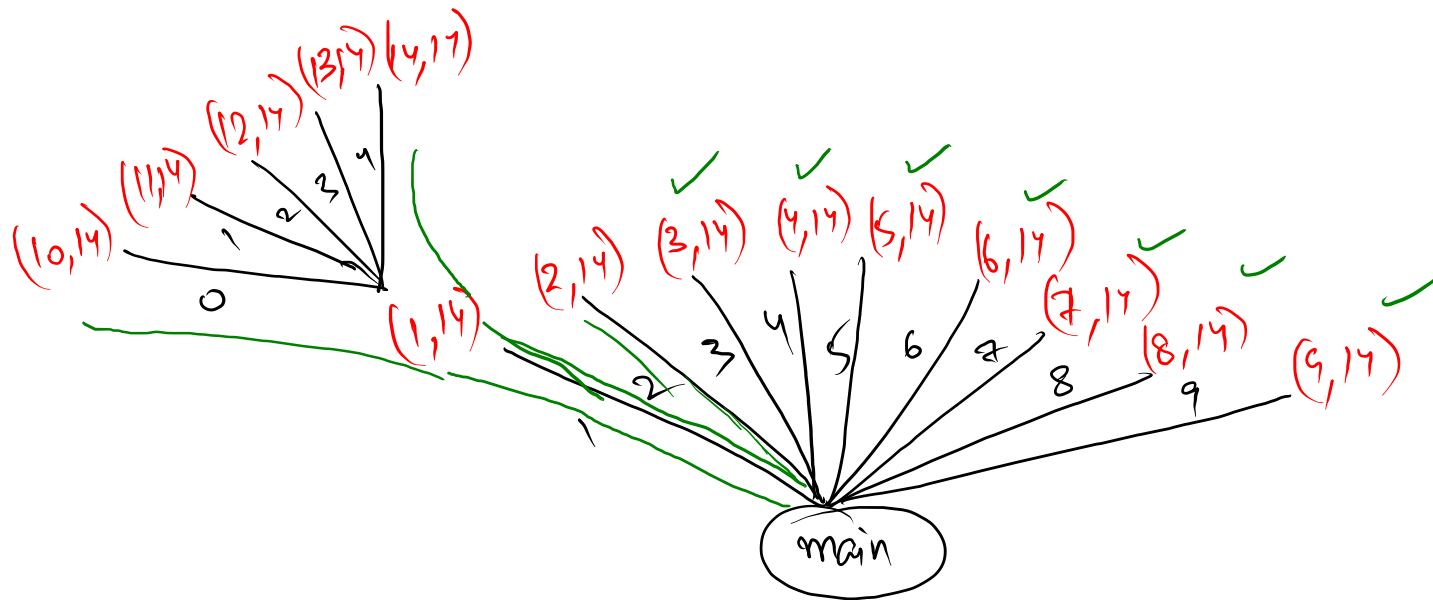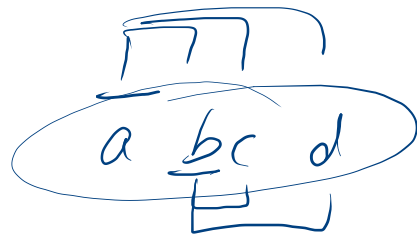
Max no. at most k times   4,3,2,1,0

5123 467   6123 54 7   7123 564

4-5   4-4   4-7   1-2   1-4   1-5   1-6   1-3

7124 563   4 12 3567   7124 563

7123 564
6123 54 7
5123 467
4123 567
3124 567
max = 7134 567

3-4   3-7

312 4 567

2-3

213 456

1-2   1-7 1-4 1-8   1-6   1-7   2-3   2-4   2-5   2-6

1234567 4

Integer. parse Int

String

123 4 5 67

String Builder → Chars

1
2
3
4

a bc d

Pep

Consei 1's → Clab

```
0 0 0  ⟶   ⎧ Pep
0 01   ⟶   ⎪ pe1
0 10   ⟶   ⎪ p1p
011    ⟶   ⎨ p2
100    ⟶   ⎪ 1ep
101    ⟶   ⎪ 1e1
110    ⟶   ⎪ 2p
111    ⟶   ⎩ 3
```

$\boxed{A B C D}$

$\boxed{0\ 0\ 0\ 0}$	$\rightarrow$	A B C D
$\boxed{0\ 0\ 0\ 1}$	$\rightarrow$	A B C 1
$\boxed{0\ 0\ 1\ 0}$	$\rightarrow$	A B 1 D
$0\ 0\ 1\ 1$	$\rightarrow$	A B 2
$\boxed{0\ 1\ 0\ 0}$	$\rightarrow$	A 1 C D
$0\ 1\ 0\ 1$	$\rightarrow$	A 1 C 1
$0\ 1\ 1\ 0$	$\rightarrow$	A 2 D
$0\ 1\ 1\ 1$	$\rightarrow$	A 3
$\boxed{1\ 0\ 0\ 0}$	$\rightarrow$	1 B C D
$1\ 0\ 0\ 1$	$\rightarrow$	1 B C 1
$1\ 0\ 1\ 0$	$\rightarrow$	1 B 1 D
$1\ 0\ 1\ 1$	$\rightarrow$	1 B 2
$1\ 1\ 0\ 0$	$\rightarrow$	2 C D
$1\ 1\ 0\ 1$	$\rightarrow$	2 C 1
$1\ 1\ 1\ 0$	$\rightarrow$	3 D
$\boxed{1\ 1\ 1\ 1}$	$\rightarrow$	4

$\boxed{Question}$

$\boxed{Combination} \rightarrow \boxed{Select}$

$$2^4 = {^4C_0} + {^4C_1} + {^4C_2} + {^4C_3} + {^4C_4}$$

$\boxed{Permutation} \rightarrow \boxed{Arrange}$

$\boxed{Unique} \rightarrow \boxed{distinct}$

pep | 0   pe | 1   p1p | 0   p | 2   1ep | 0   1e | 1   2p | 0   — | 3

if (idx == str.length()){

}

(pep) ⇓

pe | 0

2   pe | 0

p | 1

1e | 0

— | 2

(pep) ⇓   0   1   p | 0   — | 1

(pep) ⇓   0   — | 0

pep   Pe⊥

(pep|0|3)   (pe|1|3)

         L₁   L₃

pep   pe|0|2

              L₄

pep   p|0|1

pep   L₁

pep   -|0|0
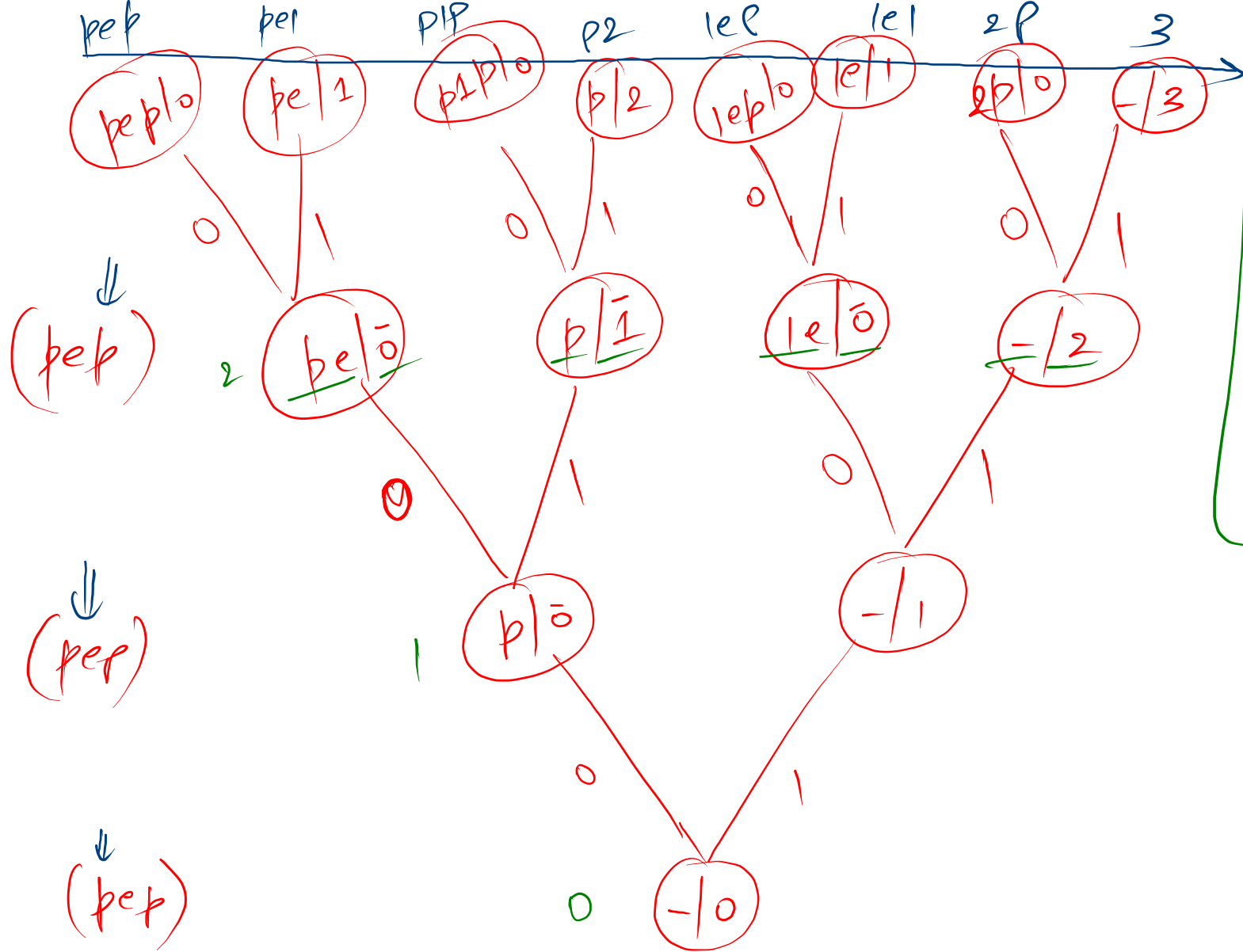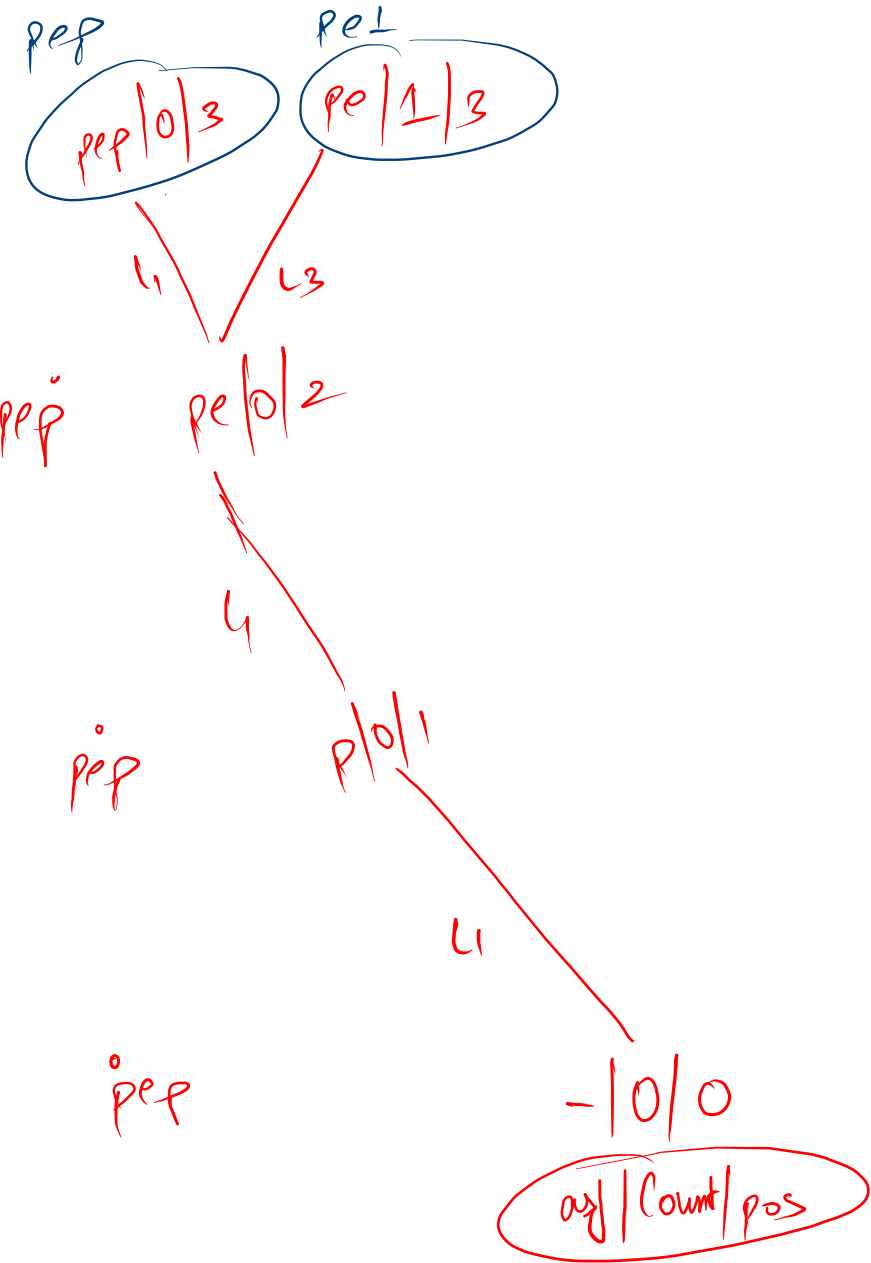
                (asf|Count|pos)

```java
public static void solution(String str, String asf,int count, int pos){
    if(pos == str.length()){
        if(count != 0){
            System.out.println(asf+count);
        }else{
            System.out.println(asf);
        }
        return;
    }

    char ch = str.charAt(pos);

    if(count == 0){
        solution(str,asf+ch,0,pos+1);       -1
    }else{
        solution(str,asf+count+ch,0,pos+1);  -2
    }

    solution(str,asf,count+1,pos+1);     -)
}
```
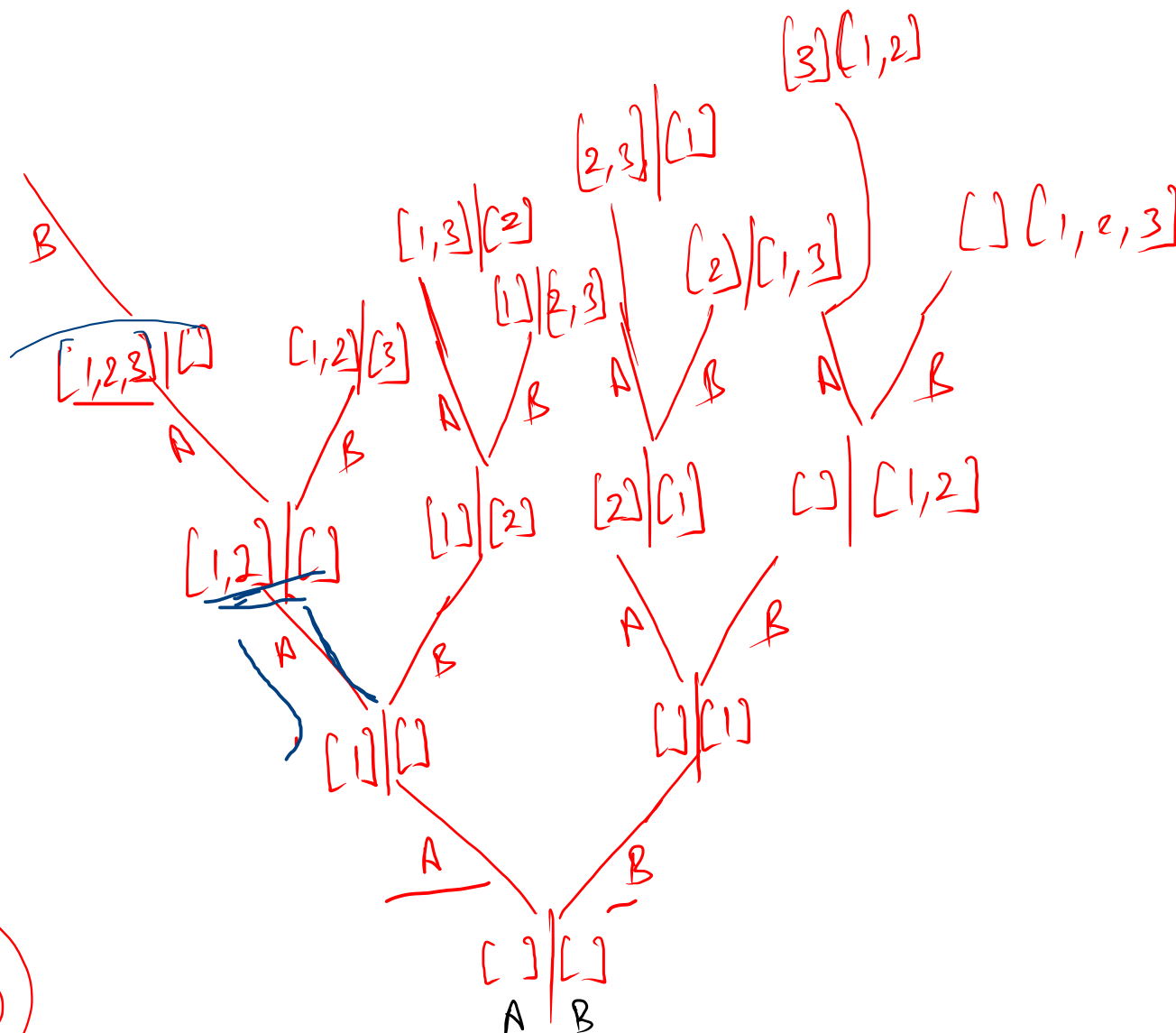
A

B

③

(3,3)

6 1 2 3 4 5 6

[1,2,3] | [ ]

[1,2] | [3]

[1,3] | [2]

[1] | [2,3]

[2,3] | [1]

[2] | [1,3]

[3] | [1,2]

[ ] | [1,2,3]

B

A   B

A   B   A   B   A   B

[1,2] | [ ]

[1] | [2]   [2] | [1]   [ ] | [1,2]

A   B

[ ] | [ ]   [ ] | [1]

A   B

[ ] | [ ]

A   B