

Selection \Rightarrow Combination

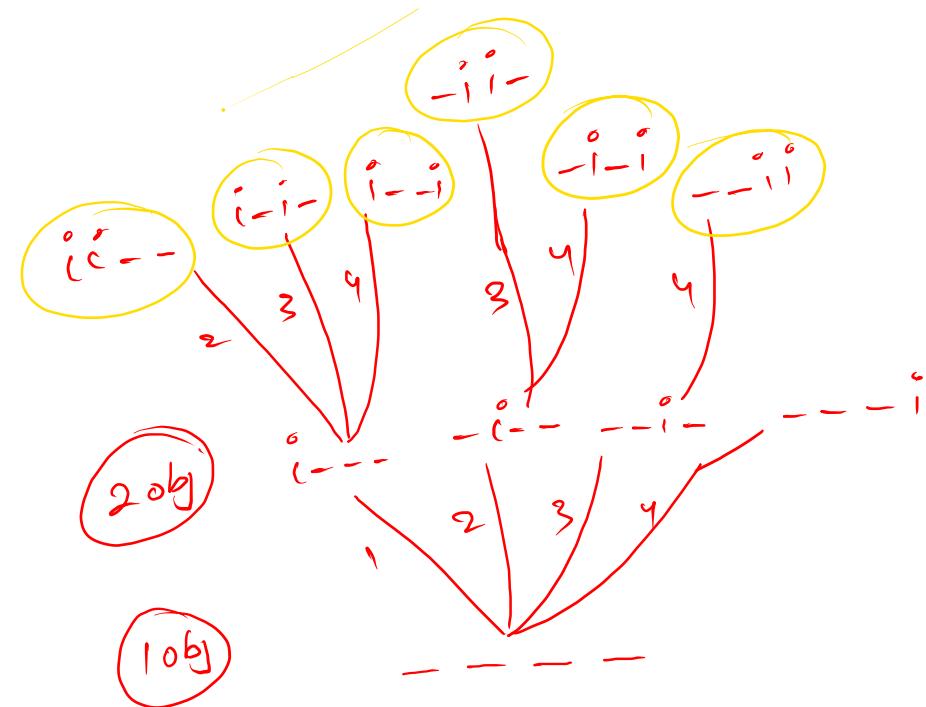
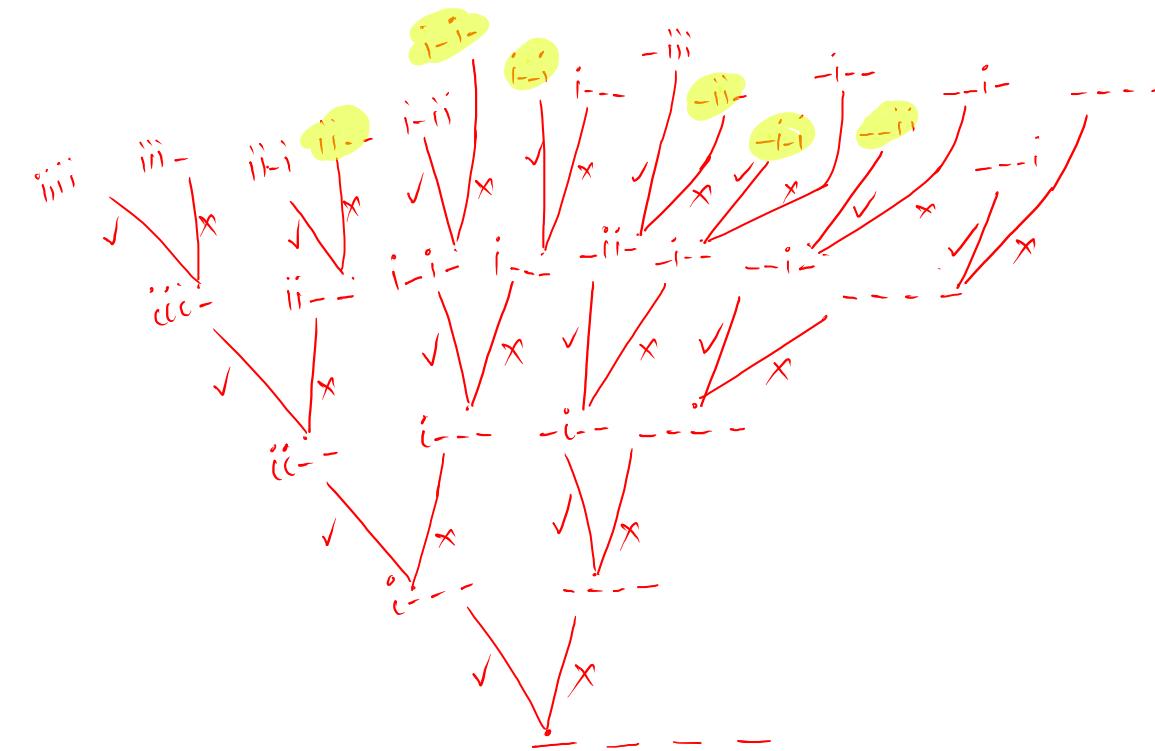
Box choose

$${}^4C_2 \Rightarrow {}^4C_0 + {}^4C_1 + {}^4C_2 + {}^4C_3 + {}^4C_4$$

4 boxes \Rightarrow 22

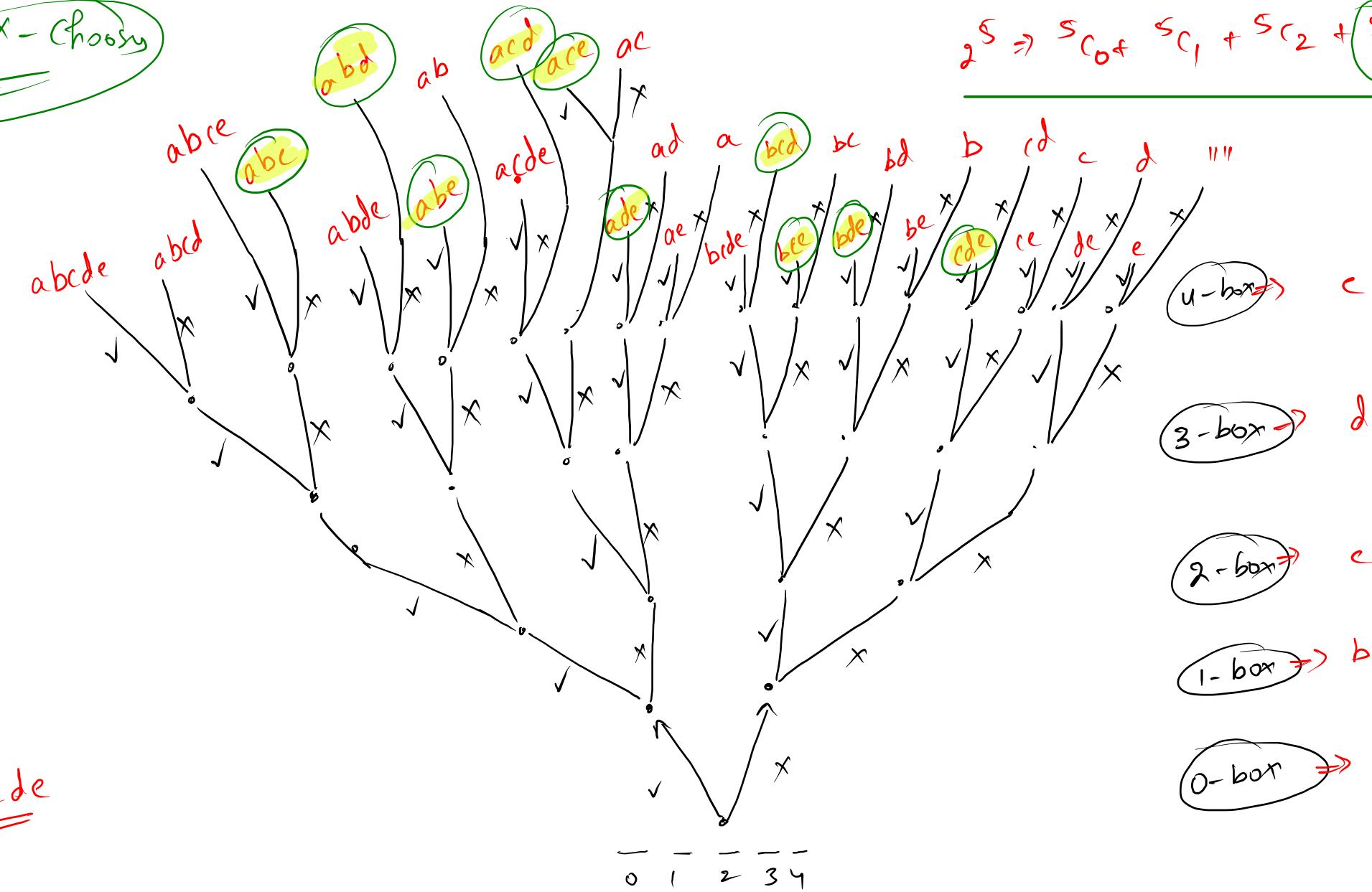
$${}^4C_2 = \frac{4!}{2!2!} \Rightarrow 6 \quad \checkmark$$

Object choose



Box - Chosy

$$2^S \Rightarrow S_{C_0} + S_{C_1} + S_{C_2} + S_{C_3} + S_{C_4} + S_{C_5}$$



abcde

0-box $\Rightarrow a$

1-box $\Rightarrow b$

2-box $\Rightarrow c$

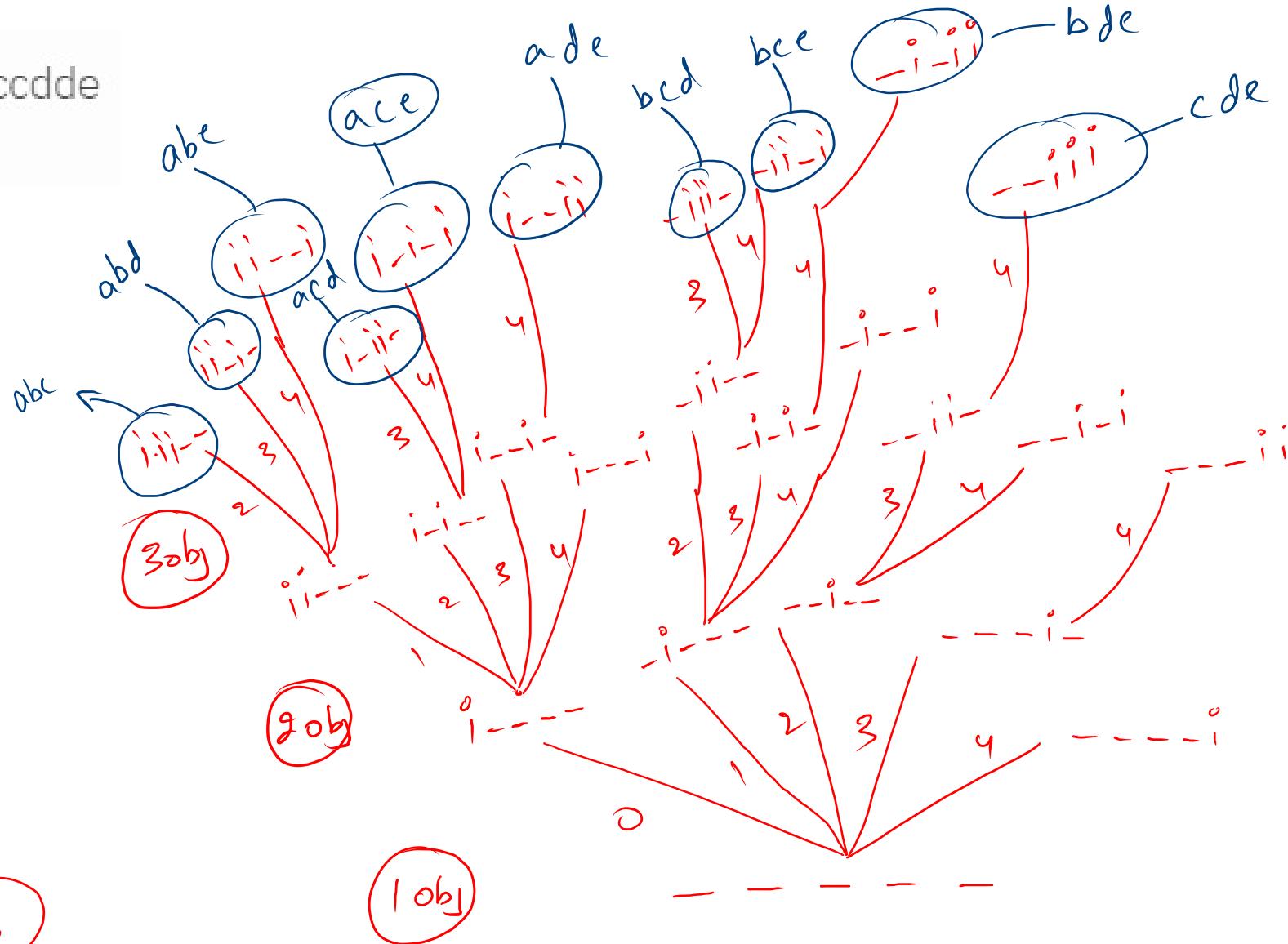
3-box $\Rightarrow d$

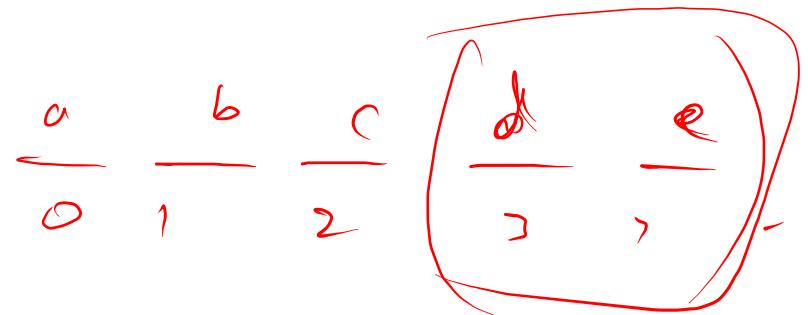
u-box $\Rightarrow c$

aabbccdde

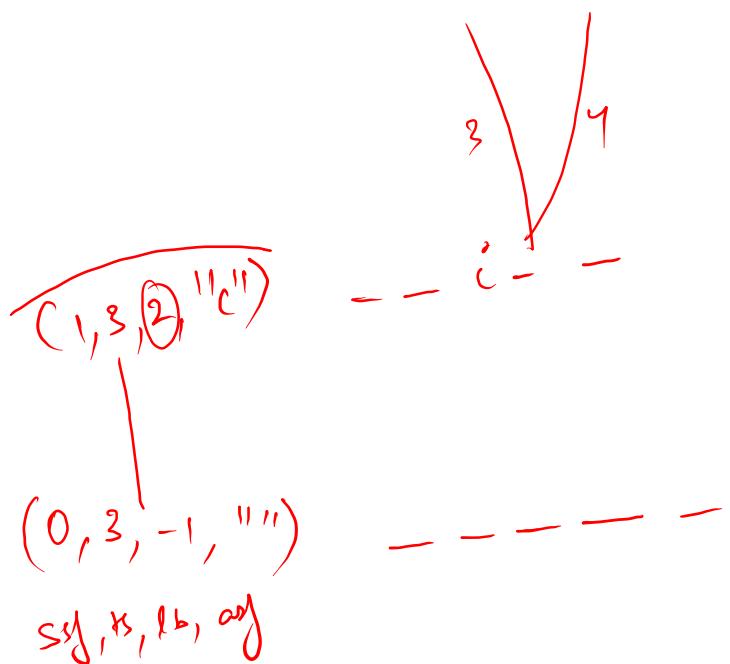
3

abcde





abcde



```
public static void combination(int ssf,int ts,int lb,String ustr,String asf) {
    if(ssf == ts){
        System.out.println(asf);
        return;
    }
    for(int idx = lb + 1; idx < ustr.length() ; idx++){
        char ch = ustr.charAt(idx);
        combination(ssf+1,ts,idx,ustr,asf+ch);
    }
}
```

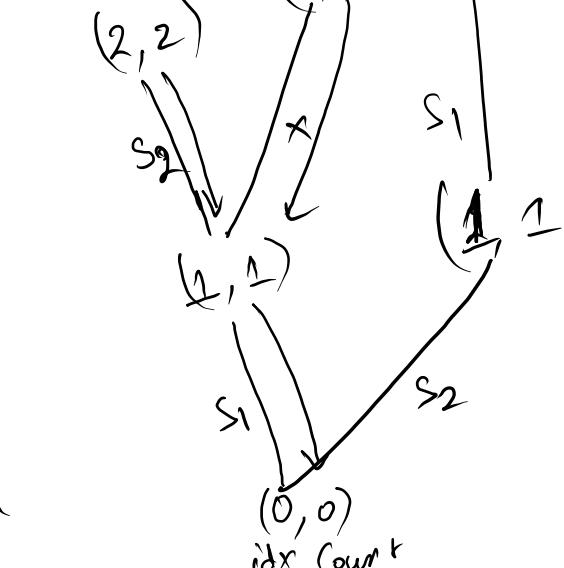
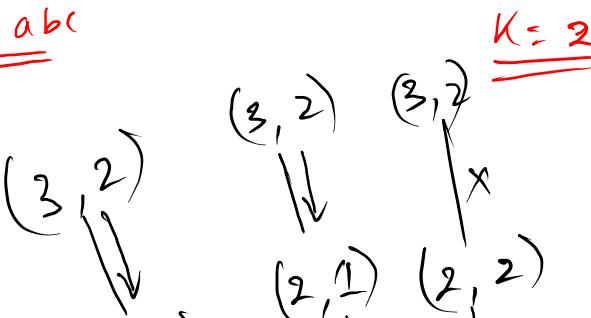


Ustr \rightarrow abc

$ch = c$

$ch = b$

$ch = a$



$Spots[] = \boxed{\cancel{null\ b} \quad \cancel{null\ a}}$

```
permutations(ustr, 0, new Character[k], 0);
```

```
}
```

```
public static void permutations(String ustr, int idx, Character[] spots, int count){
```

```
    if(idx == ustr.length()) {
```

```
        if(count == spots.length) {
```

```
            for(Character ch : spots) {
```

```
                System.out.print(ch);
```

```
            }
```

```
            System.out.println();
```

```
        }
```

```
        return;
```

```
    }
```

```
    char ch = ustr.charAt(idx);
```

```
    for(int i = 0; i < spots.length; i++) {
```

```
        if(spots[i] == null) {
```

```
            spots[i] = ch;
```

```
            permutations(ustr, idx+1, spots, count+1);
```

```
            spots[i] = null;
```

```
        }
```

```
    }
```

```
    permutations(ustr, idx+1, spots, count);
```

}

a b

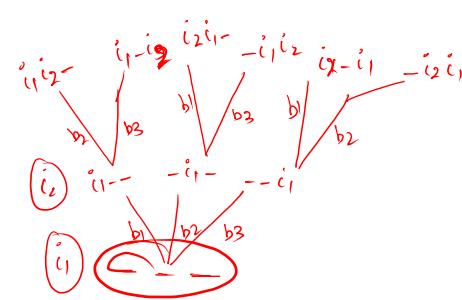
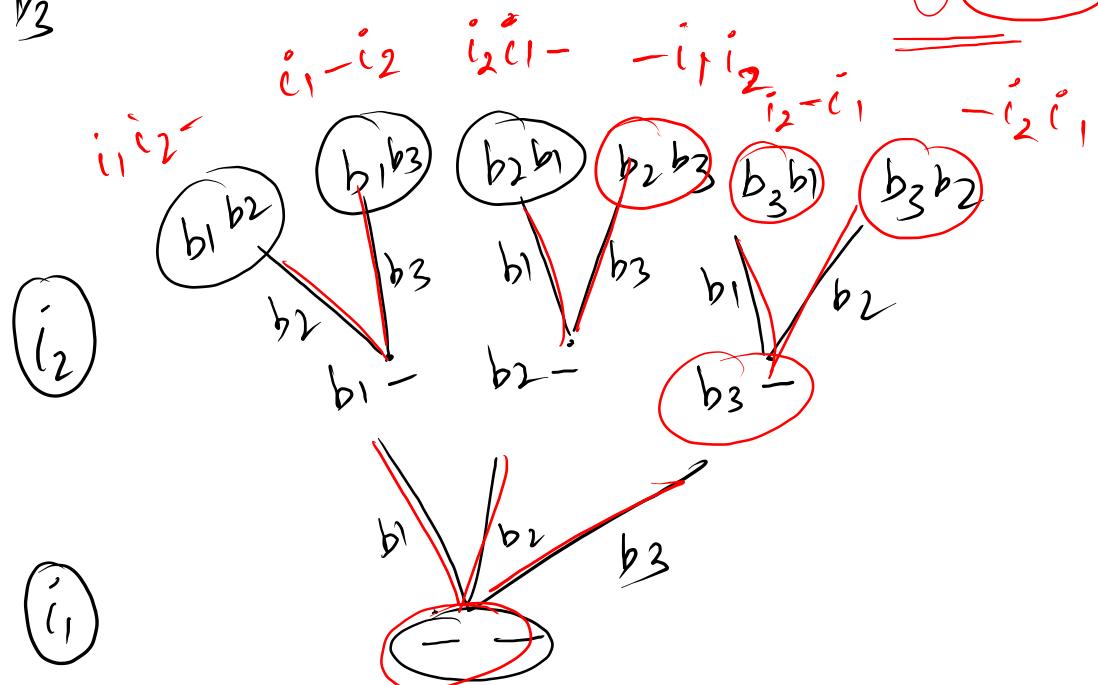
a c

b a

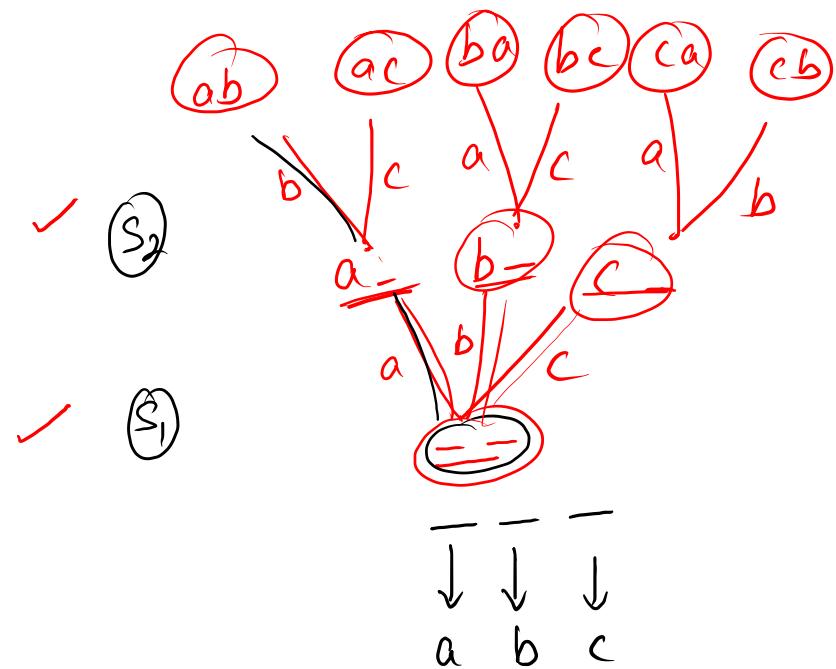
Permutation

Object choices

b_1	i_1
b_2	i_2
b_3	



a	s_1
b	s_2
c	

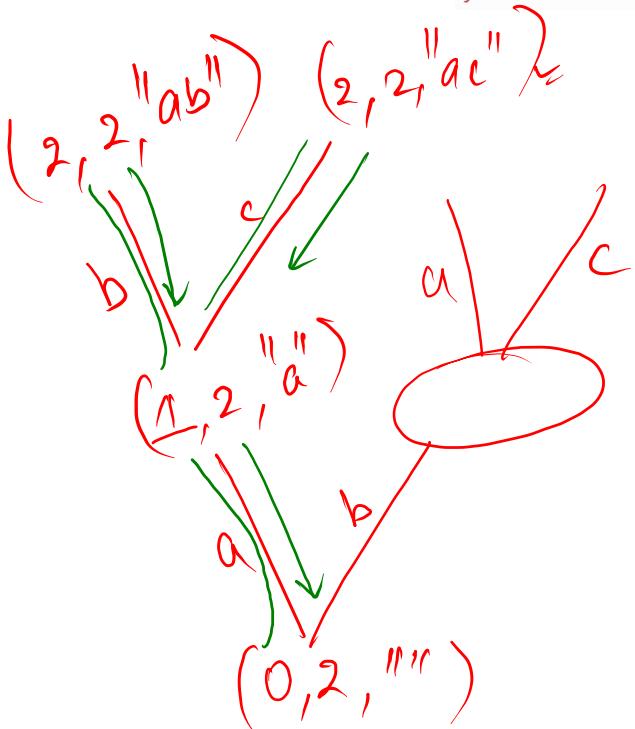


abc → 2 length word

ustr → a b c

ab

ac



```
public static void permutations(String ustr,int cs,int ts,HashSet<Character> charUsed,String asf){
    if(cs == ts){
        System.out.println(asf);
        return;
    }

    for(int i = 0 ; i < ustr.length() ; i++){
        char ch = ustr.charAt(i);
        if(charUsed.contains(ch) == false){
            charUsed.add(ch);
            permutations(ustr,cs+1,ts,charUsed,asf+ch);
            charUsed.remove(ch);
        }
    }
}
```

