

arr  $\rightarrow$       <sup>0</sup>      <sup>1</sup>      <sup>2</sup>      <sup>3</sup>      <sup>4</sup>      <sup>5</sup>  
                  15      11      40      4      4      9

$x = 4$

return 4

$n = 10$

-1

$\rightarrow$   
 ①  
 expect

idx = 0

idx = 2

$\leftarrow$   
 ②

exp

idx = 2

idx = 5



0

15

1

11

idx  
2

4

3

40

4

4

5

9



n = 4

E  
→

idx = 0

4

f

idx = 1

4

findx = pairn(idx + 1)

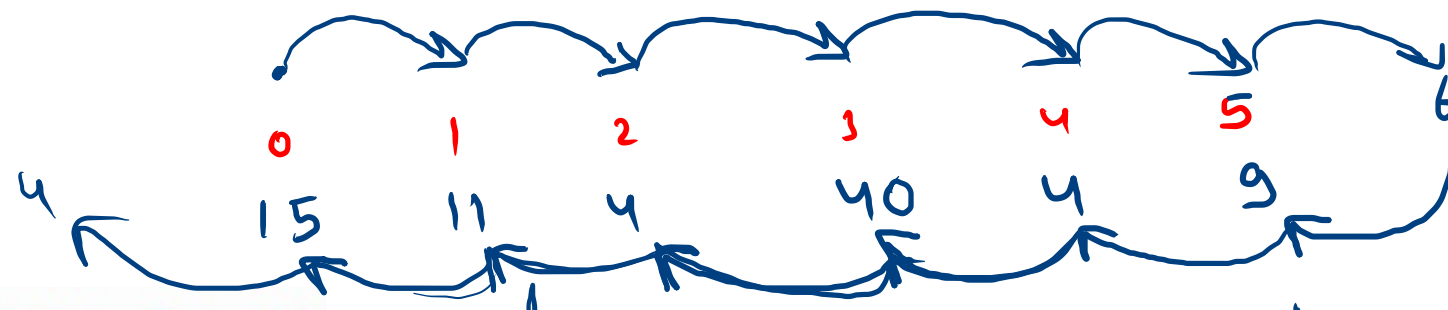
if (findx != -1) {  
    return findx;  
}

else {

    if (arr[idx] == n)  
        return idx

    else -1;

}

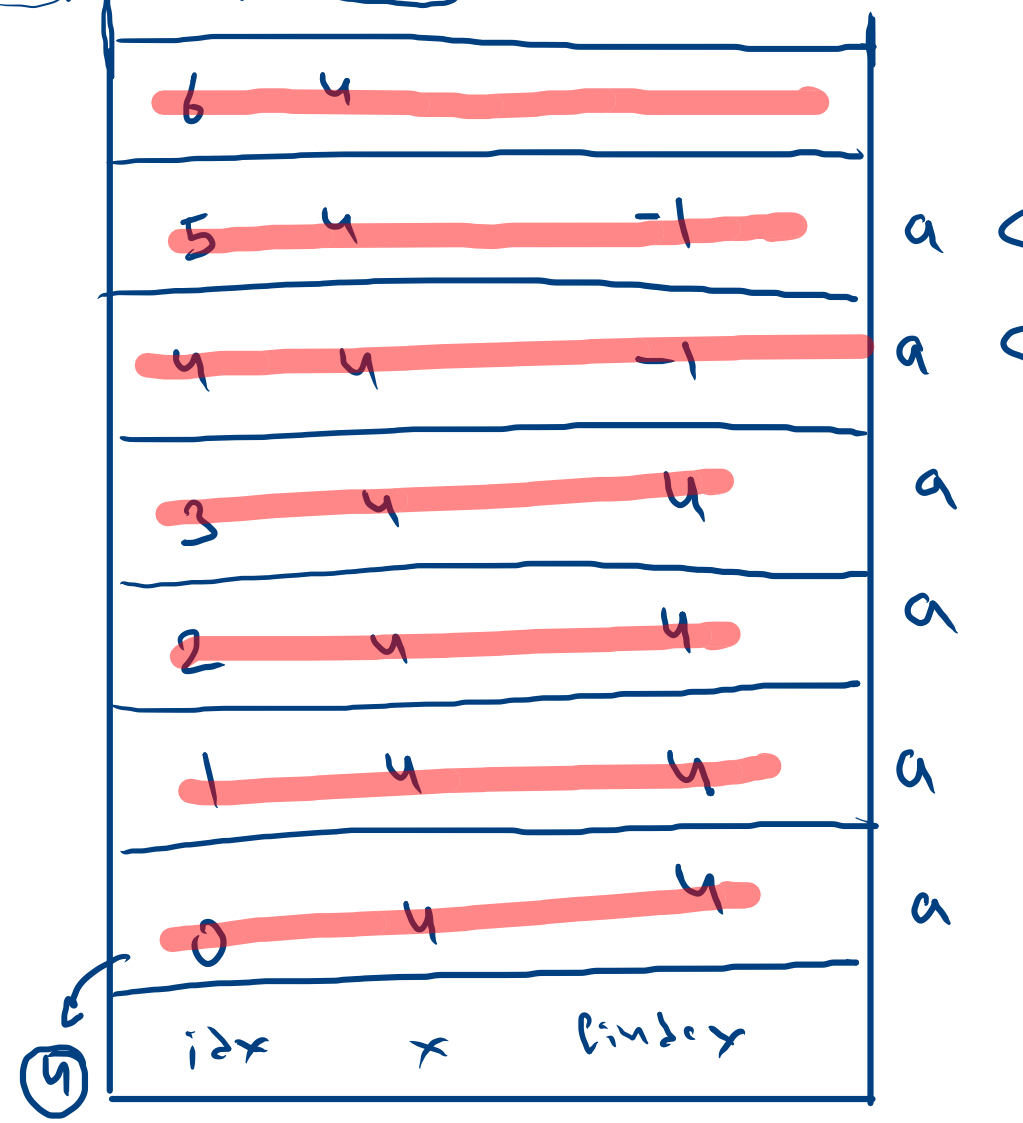


```

public static int lastIndex(int[] arr, int idx, int x){
    if(idx == arr.length)return -1;

a   int findex = lastIndex(arr, idx+1, x);
    if(findex != -1){
b       return findex;
    }else{
c       if(arr[idx] == x){
            return idx;
        }else{
            return -1;
        }
    }
}

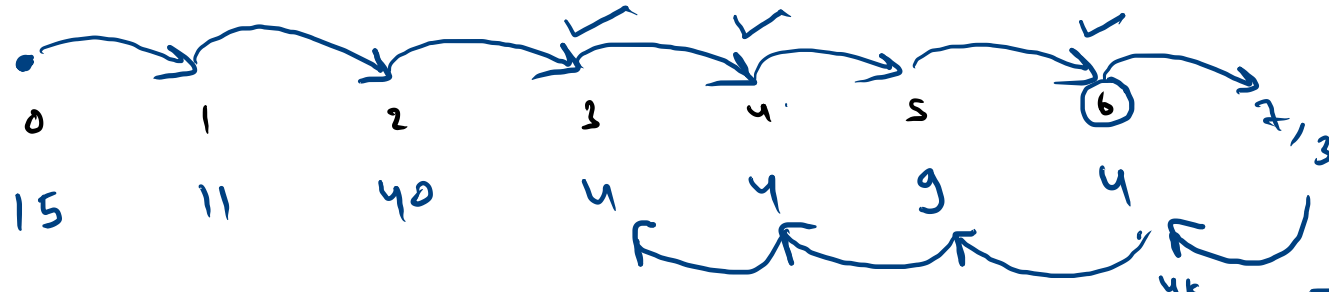
```





4	7	3
4	6	2
4	5	2
4	4	1
4	3	0
4	2	0
4	1	0
4	0	0
n	idx	fst

arr →



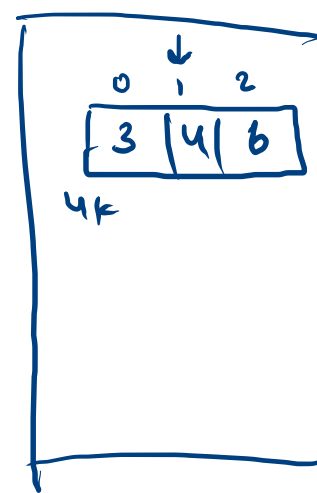
3 | 4 | 6

if (arr[idx] == n)  
allIndices(arr, n, idx+1, fst+1)

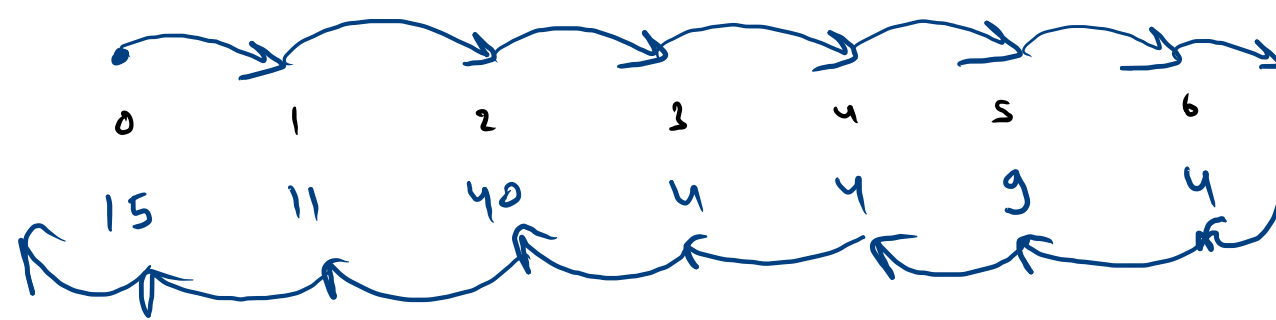
else

allIndices(arr, n, idx+1, fst)

heap



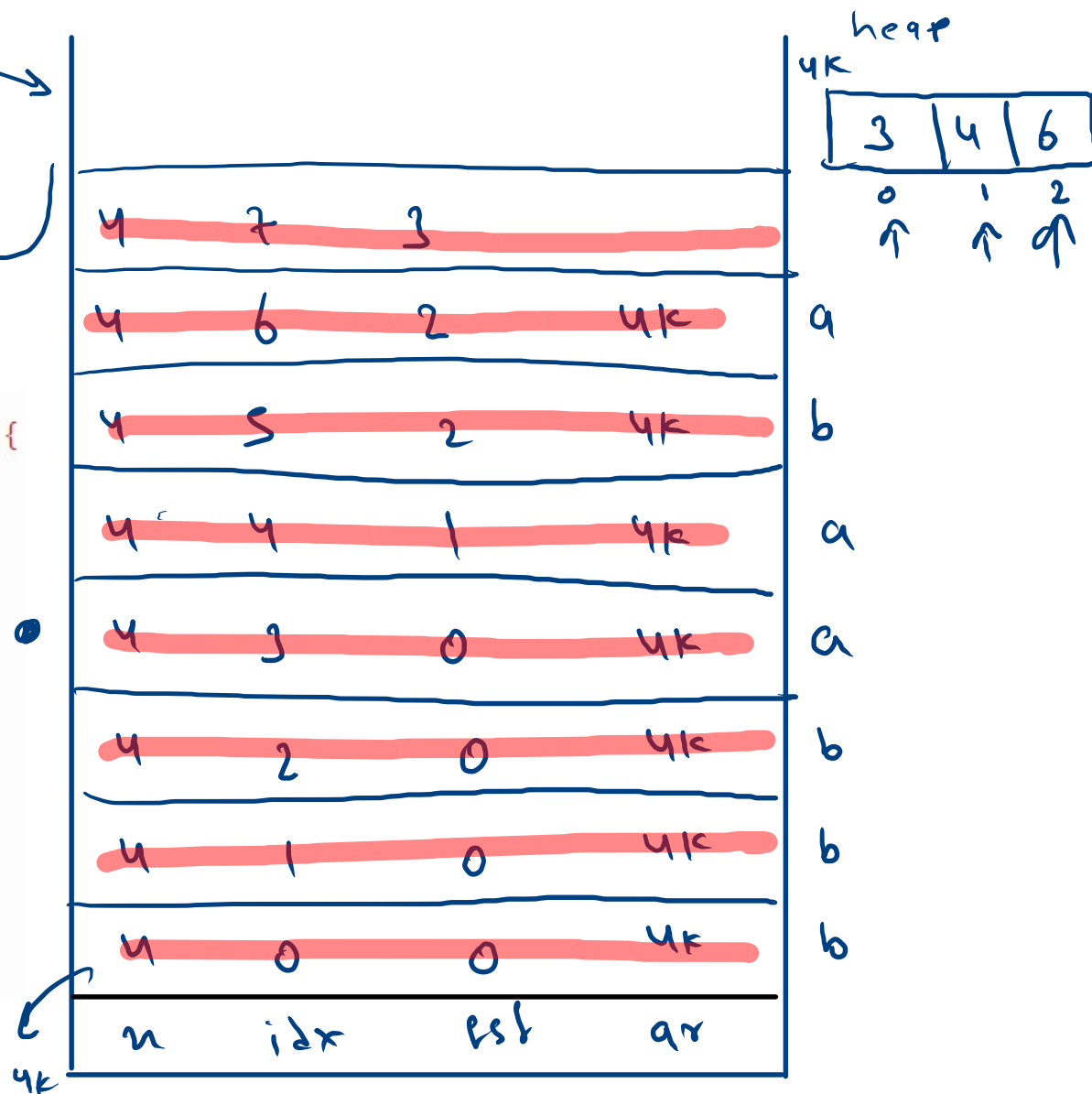
```
public static int[] allIndices(int[] arr, int x, int idx, int fst) {
    // write ur code here
}
```



```

public static int[] allIndices(int[] arr, int x, int idx, int fsf) {
    if(idx == arr.length){
        int ar[] = new int[fsf];
        return ar;
    }

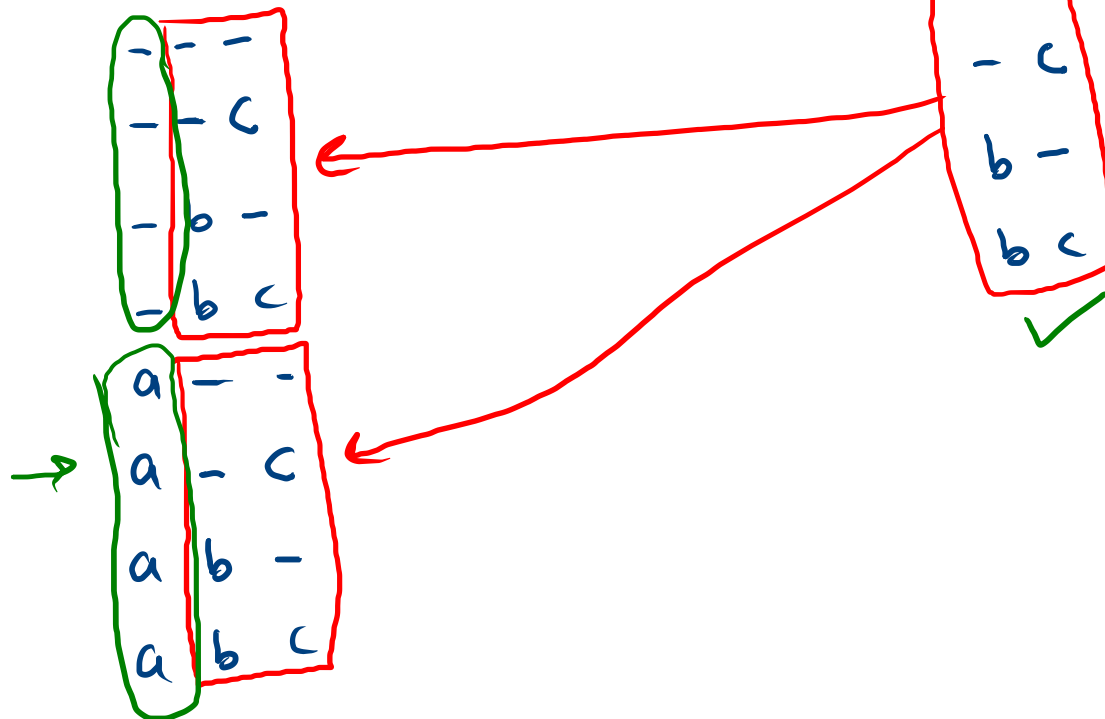
    if(arr[idx] == x){
        int ar[] = allIndices(arr, x, idx+1, fsf+1);
        ar[fsf] = idx;
        return ar;
    }else{
        int ar[] = allIndices(arr, x, idx+1, fsf);
        return ar;
    }
}
  
```



str → abc

E  
str → abc

Path  
str → bc



[ "",  
"c",  
"b",  
"bc",  
... ]

```

public static void main(String[] args) throws Exception {

    Scanner scn = new Scanner(System.in);

    String str = scn.nextLine();
    ArrayList<String> list = gss(str);
    System.out.println(list);
}

```

```

public static ArrayList<String> gss(String str) {
    if(str.length() == 0){
        ArrayList<String> myAns = new ArrayList<String>();
        myAns.add("");
        return myAns;
    }

    String fstr = str.substring(1);
    ArrayList<String> fans = gss(fstr);

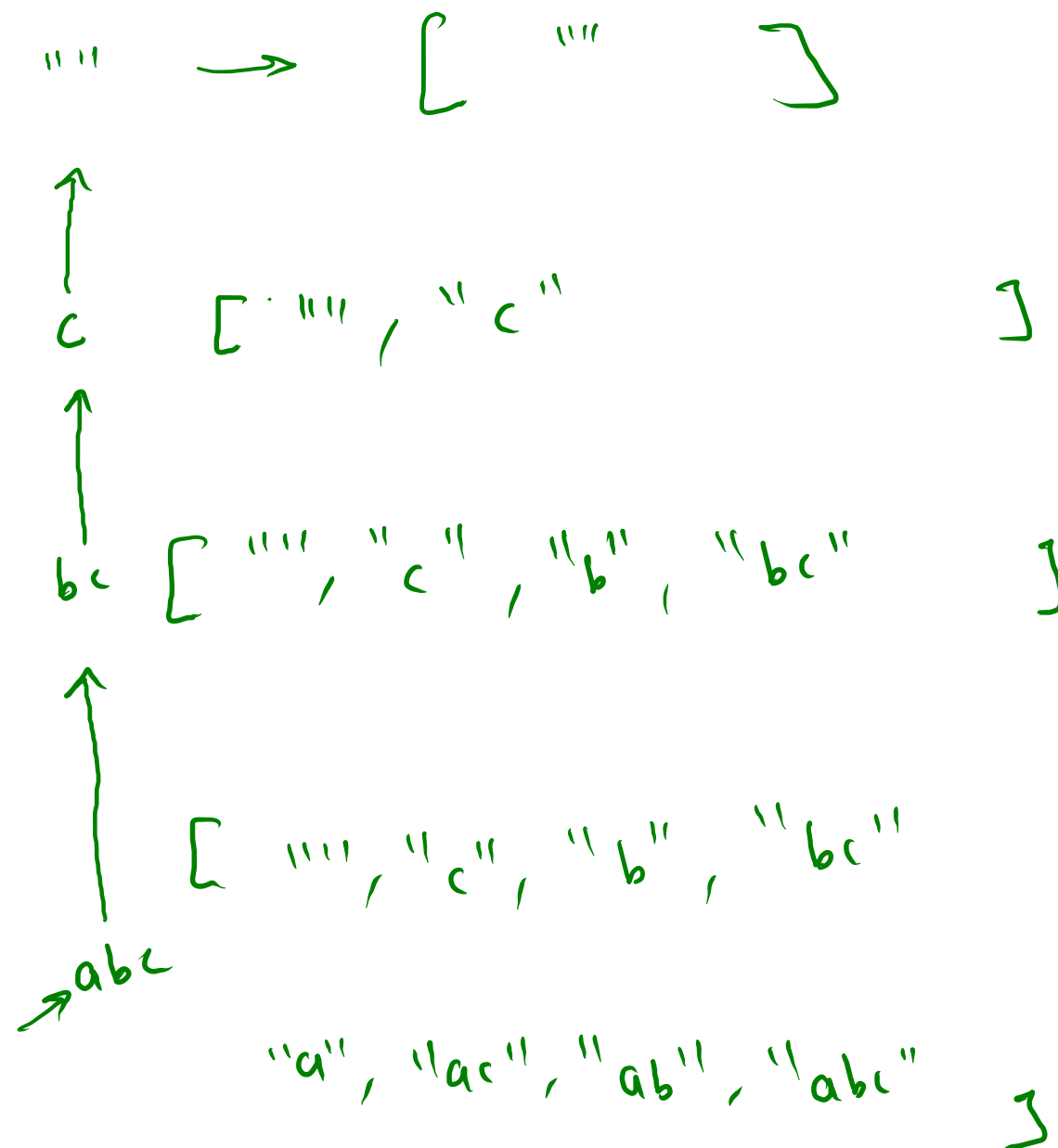
    ● ArrayList<String> myAns = new ArrayList<String>();

    [ for(String s: fans){
        myAns.add(s); ←
    }

    [ char ch = str.charAt(0);
      for(String s: fans){
        myAns.add(ch+s);
      }
    ]

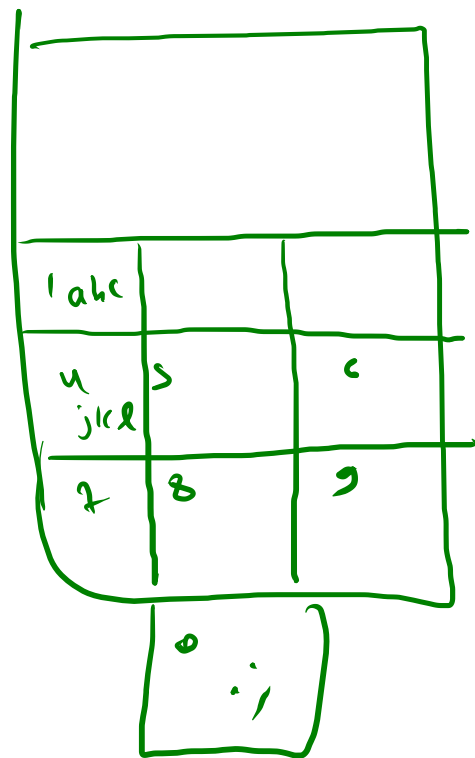
    return myAns;
}

```

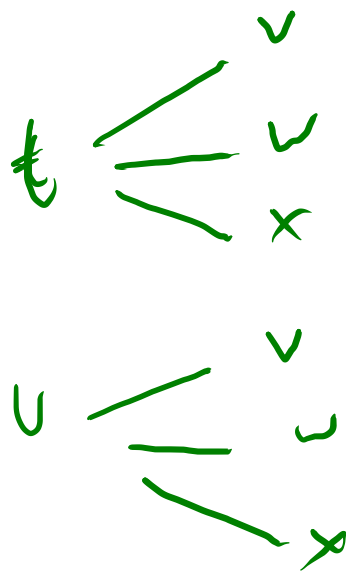




0 -> ;  
 1 -> abc  
 2 -> def  
 3 -> ghi  
 4 -> jkl  
 5 -> mno  
 6 -> pqrs  
 7 -> tu  
 8 -> vwx  
 9 -> yz



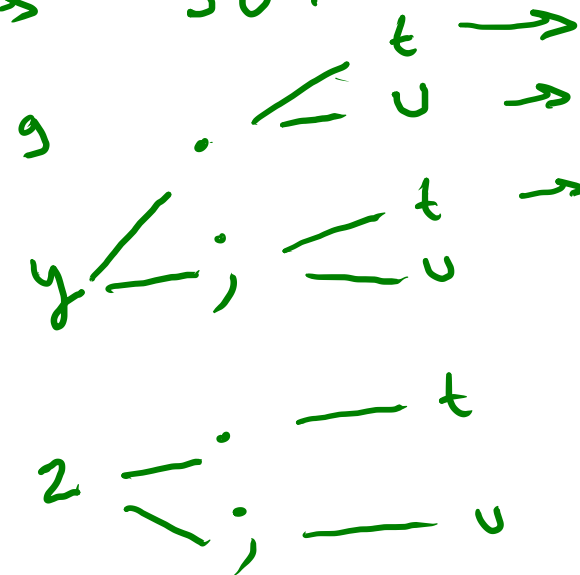
"7 8"



[

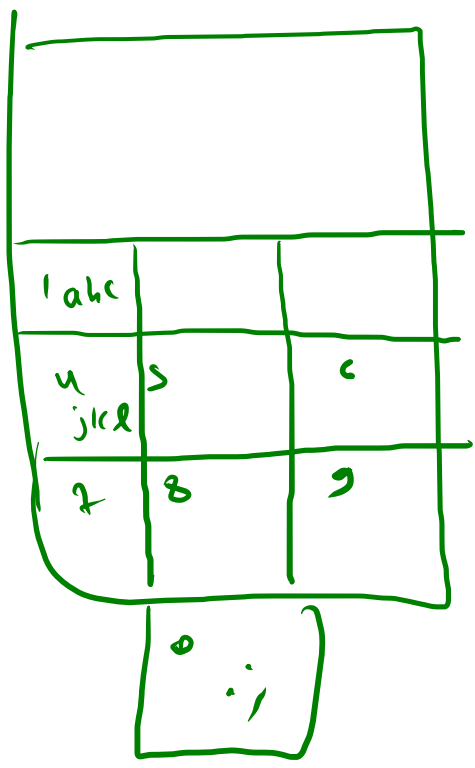
t v  
 t u  
 t x  
 u v  
 u u  
 u x }

str -> "g o t"

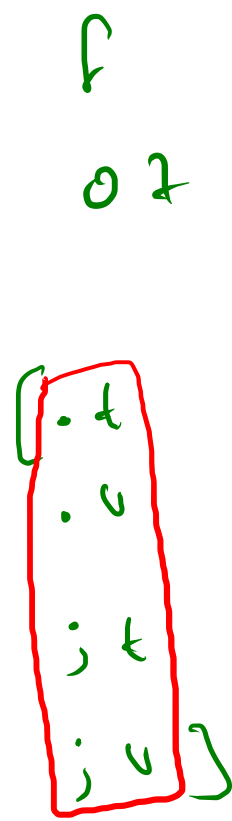
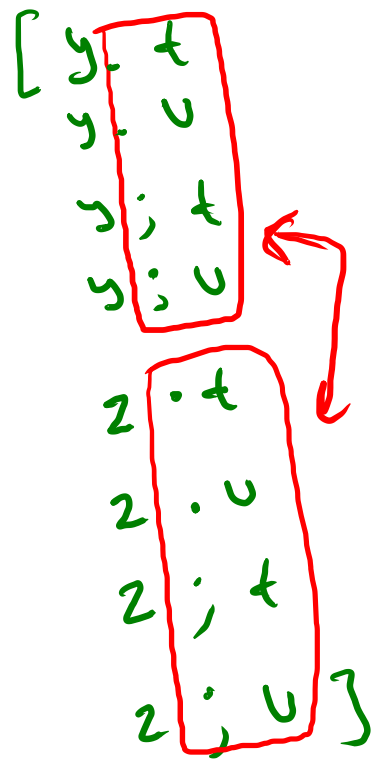


[ g. t  
 y. u  
 y; t  
 y; u  
 z. t  
 z; u ]

- 0 -> ;
- 1 -> abc
- 2 -> def
- 3 -> ghi
- 4 -> jkl
- 5 -> mno
- 6 -> pqrs
- 7 -> tu
- 8 -> vwx
- 9 -> yz



E  
sr -> "907"



'j'  
 0 → 49  
 1 → 50  
 2 → 51  
 ...  
 9

char ch = s.charAt(0)  
 int i

- 0 -> ;
- 1 -> abc
- 2 -> def
- 3 -> ghi
- 4 -> jkl
- 5 -> mno
- 6 -> pqrs
- 7 -> tu
- 8 -> vwx
- 9 -> yz

"ghr"  
 ↑  
 ch

1 → keys[index]



a.t	b.t	c.t
a.u	b.u	c.u
a;j	b;j	c;j
a;v	b;v	c;v

f  
 0 2  
 [.t  
 .u  
 ;t  
 ;v]

$$\begin{matrix} ch \\ \textcircled{11} \end{matrix} - \begin{matrix} lo' \\ 50 \end{matrix} - 49 = 1$$

$$12' - 10' = 2$$

$$51 - 49 = 2$$

```

public static ArrayList<String> getKPC(String str) {

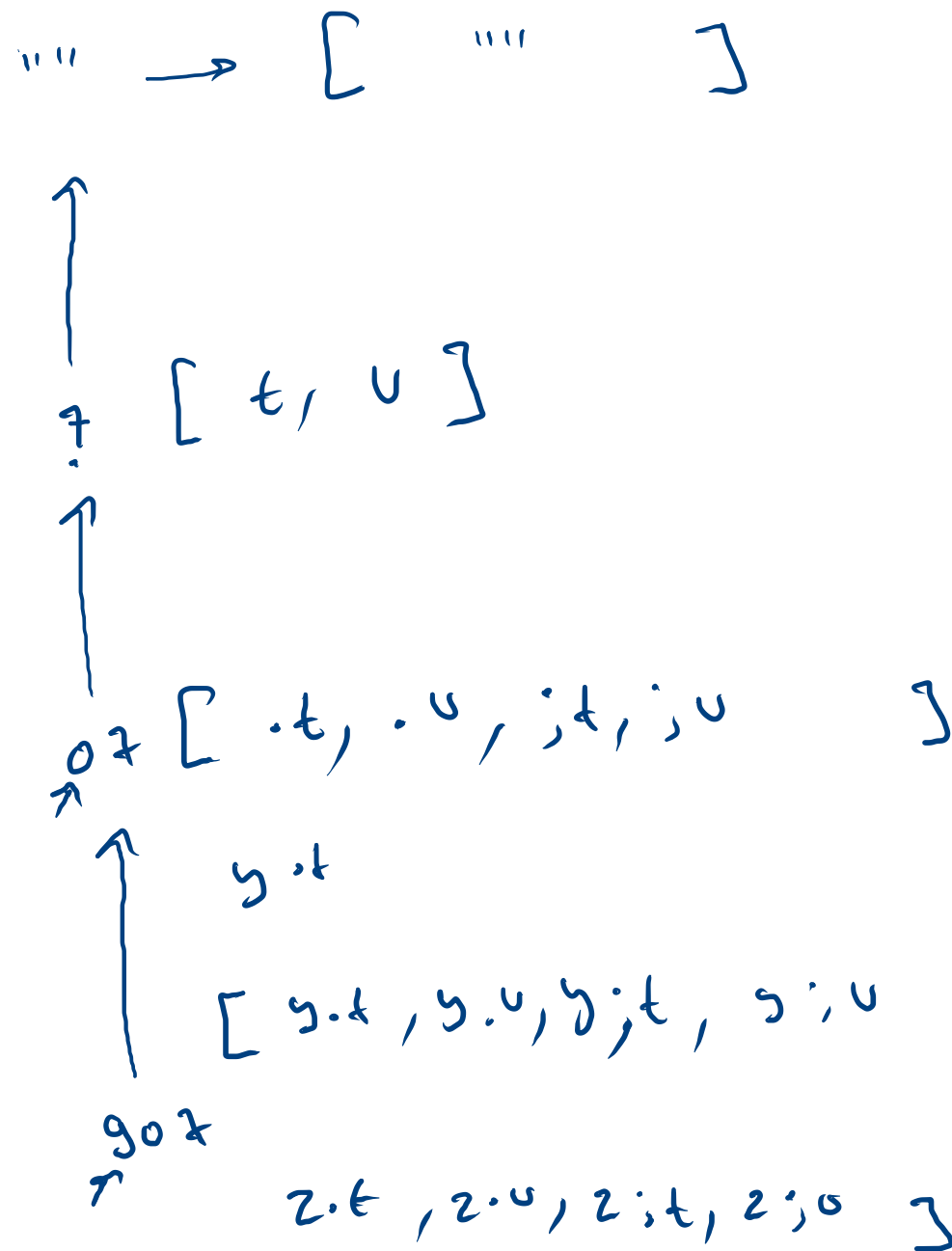
    String fstr = str.substring(1);
    ArrayList<String> fans = getKPC(fstr);

    • ArrayList<String> myAns = ArrayList<String>();

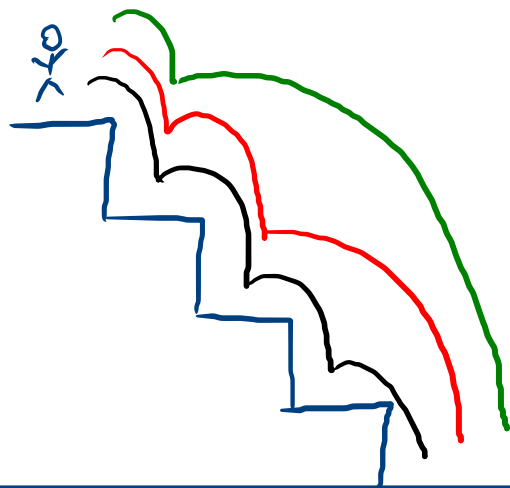
    • int index = s.charAt(0) - '0';
    → String code = keys[index]; "yz"
    for(int i=0; i<code.length(); i++){
        char ch = code.charAt(i);
        for(String s: fans){
            myAns.add(ch+s);
        }
    }

    return myAns;
}

```



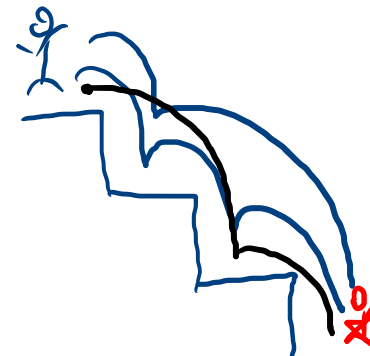
$h = 4$



1 1 1 1  
1 1 2  
1 3

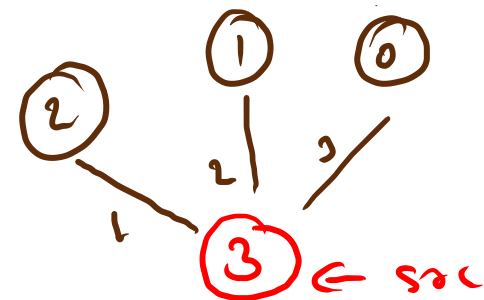
surps  $\rightarrow [1, 2, 3]$

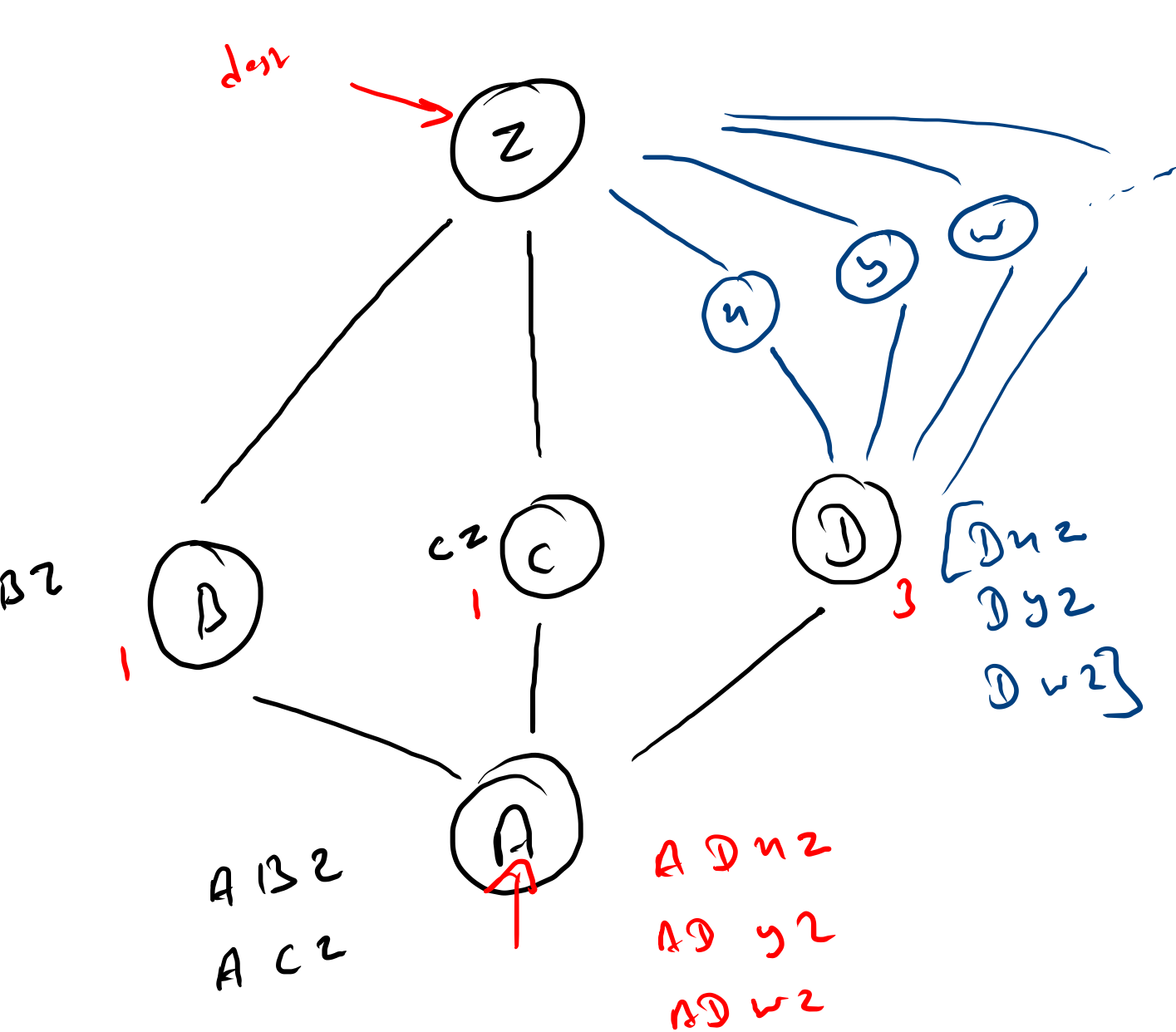
$h = 3$

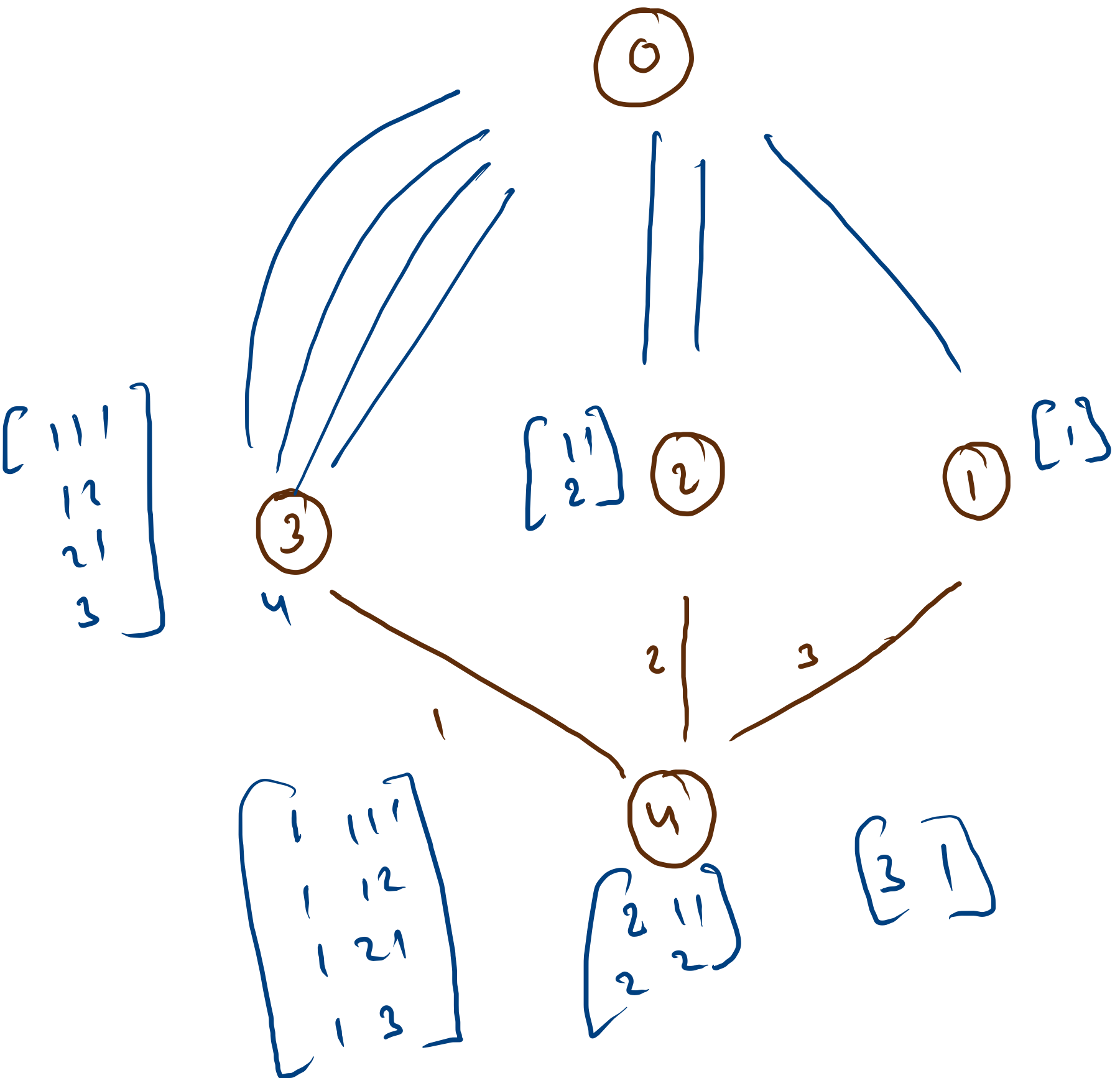


"1 1 1"  
"1 2"  
"2 1"  
"3" 3

0  $\leftarrow$  des







```

public static ArrayList<String> getStairPaths(int n) {
    if (n == 20) {
        ArrayList<String> ans = new ArrayList<String>();
        if (n-1 >= 0) {
            ArrayList<String> one = getStairPaths(n-1);
            for (String s: one) {
                ans.add("1"+s);
            }
        }
        if (n-2 >= 0) {
            ArrayList<String> two = getStairPaths(n-2);
            for (String s: two) {
                ans.add("2"+s);
            }
        }
        if (n-3 >= 0) {
            ArrayList<String> three = getStairPaths(n-3);
            for (String s: three) {
                ans.add("3"+s);
            }
        }
        return ans;
    }
}

```

n 2 3

