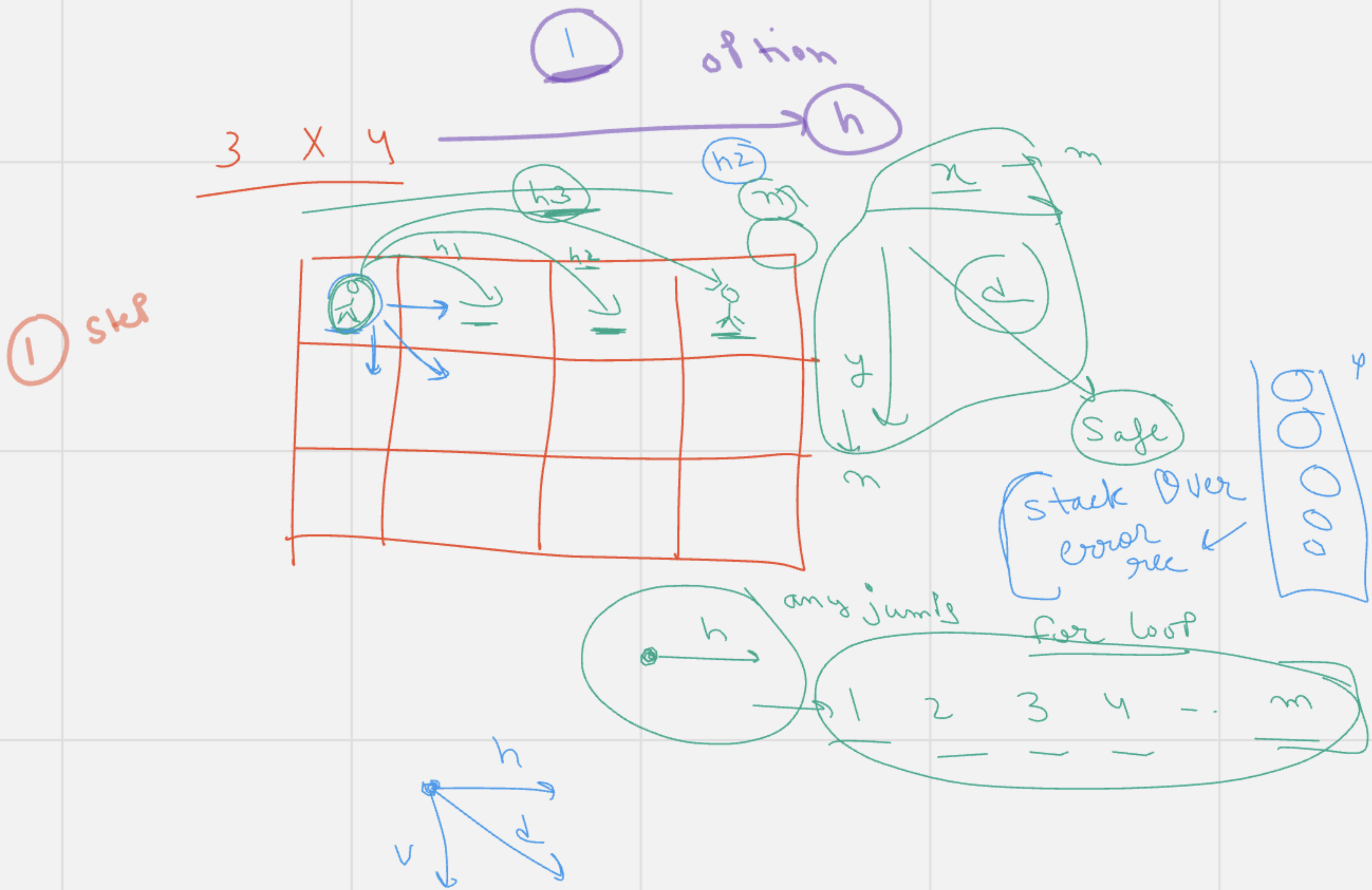
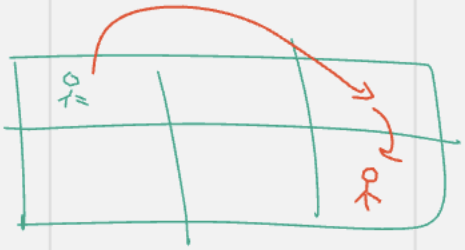
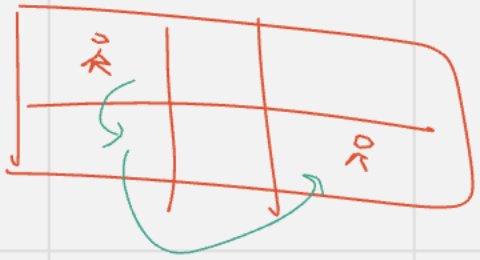
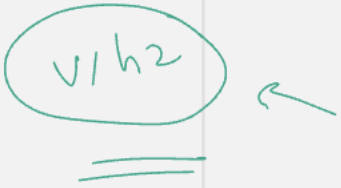
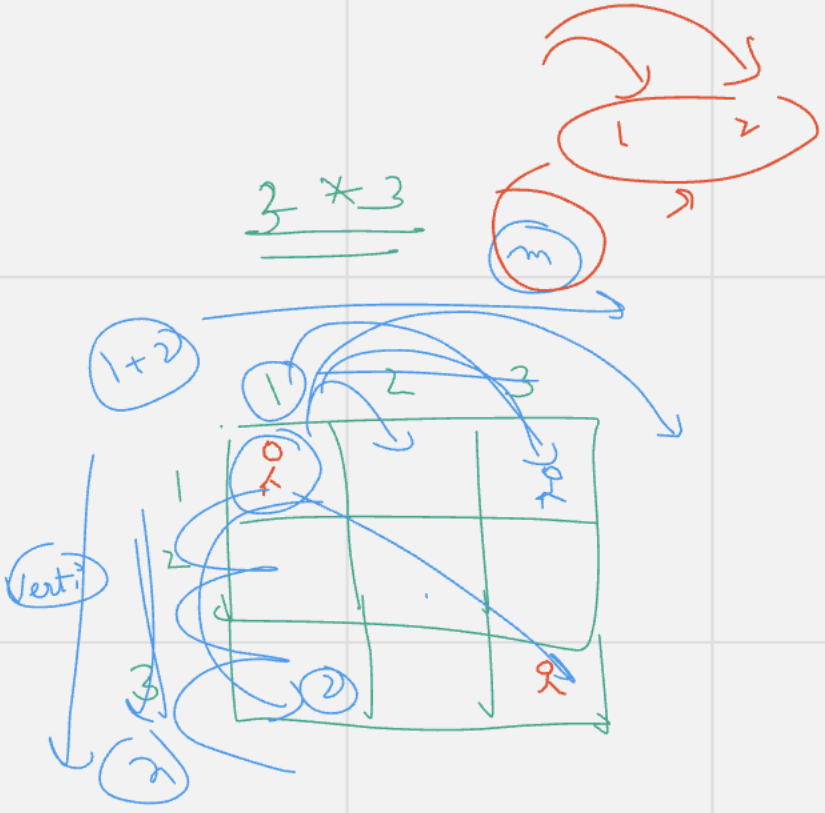

1 option
3 X 4
h
h2
h3
h1
h2
m
1 step
x
m
y
n
d
Safe
Stack Over
error
rec
any jumps
for loop
1 2 3 4 - - m
h
v
d








```java
public static void printMazePaths(int i, int j, int n, int m, String psf) {

    // safe
    if(j > m || i > n){
        return ;
    }

    // base case
    if(i == n && j == m) {
        System.out.println(psf);
        return;
    }

    for(int x = 1; x <= m; x++) {
        // hori
        printMazePaths(i, j + x, n, m, psf + "h" + x);
    }

    for(int y = 1; y <= n; y++) {
        //vert
        printMazePaths(i + y, j , n, m, psf + "v" + y);
    }

    for(int d = 1; d <= Math.min(n, m); d++) {
        // dia
        printMazePaths(i + d, j + d, n, m, psf + "d" + d);
    }


}
```



a

3

table

1 2 3

1 3 2

(1-9) single

pair 10 ≤ 20 ≤ 26

10 23

0 23

" "

table

1 → a

2 → b

3 → c

9

10 → j

11 → k

12 → l

m

23 w

24 x

25 y

26 → z

abc

lc

aw

p

q

R

S

T

21 U

22 V

23 W

1 23

a w

123

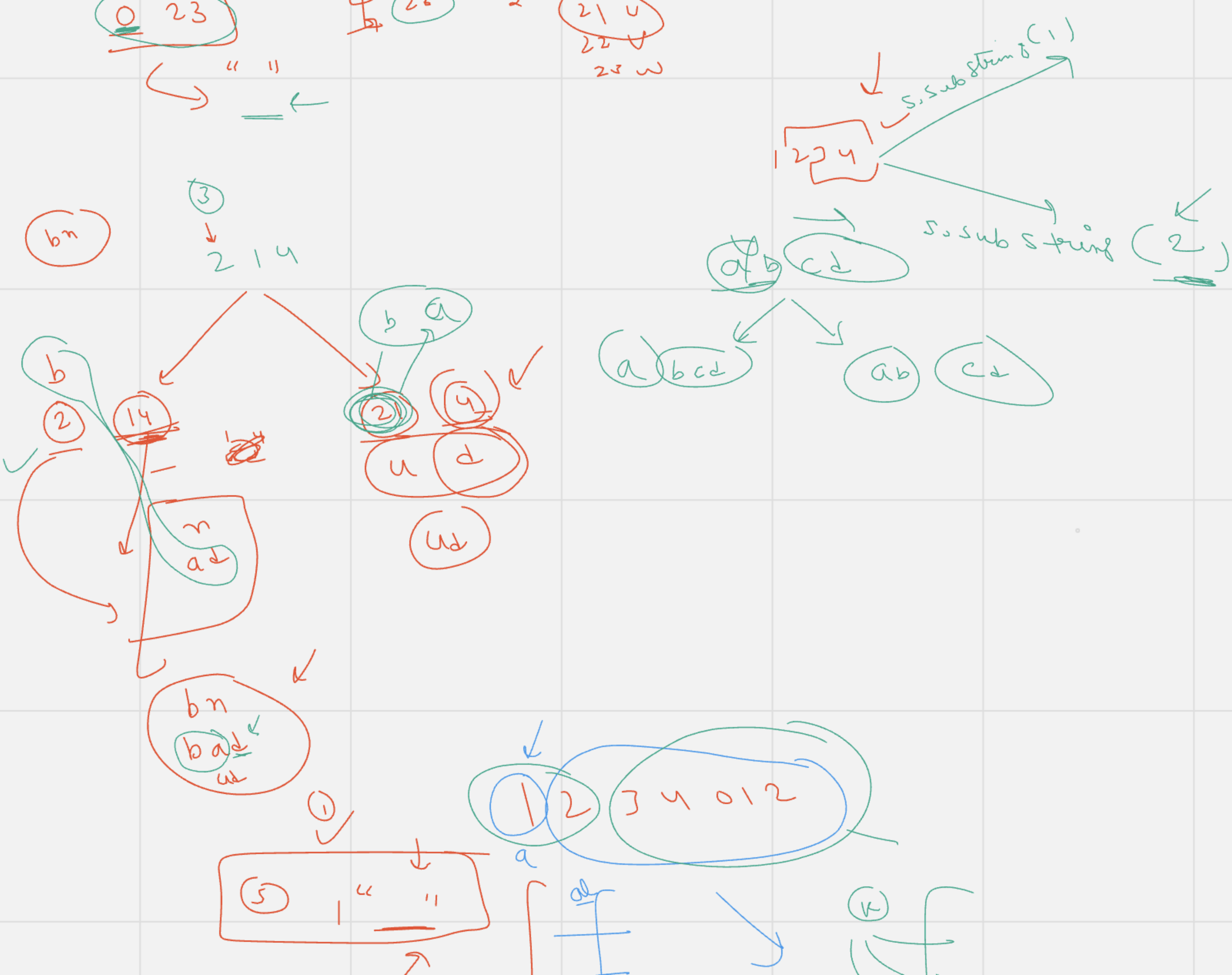
'1' '2' '3'

a b c

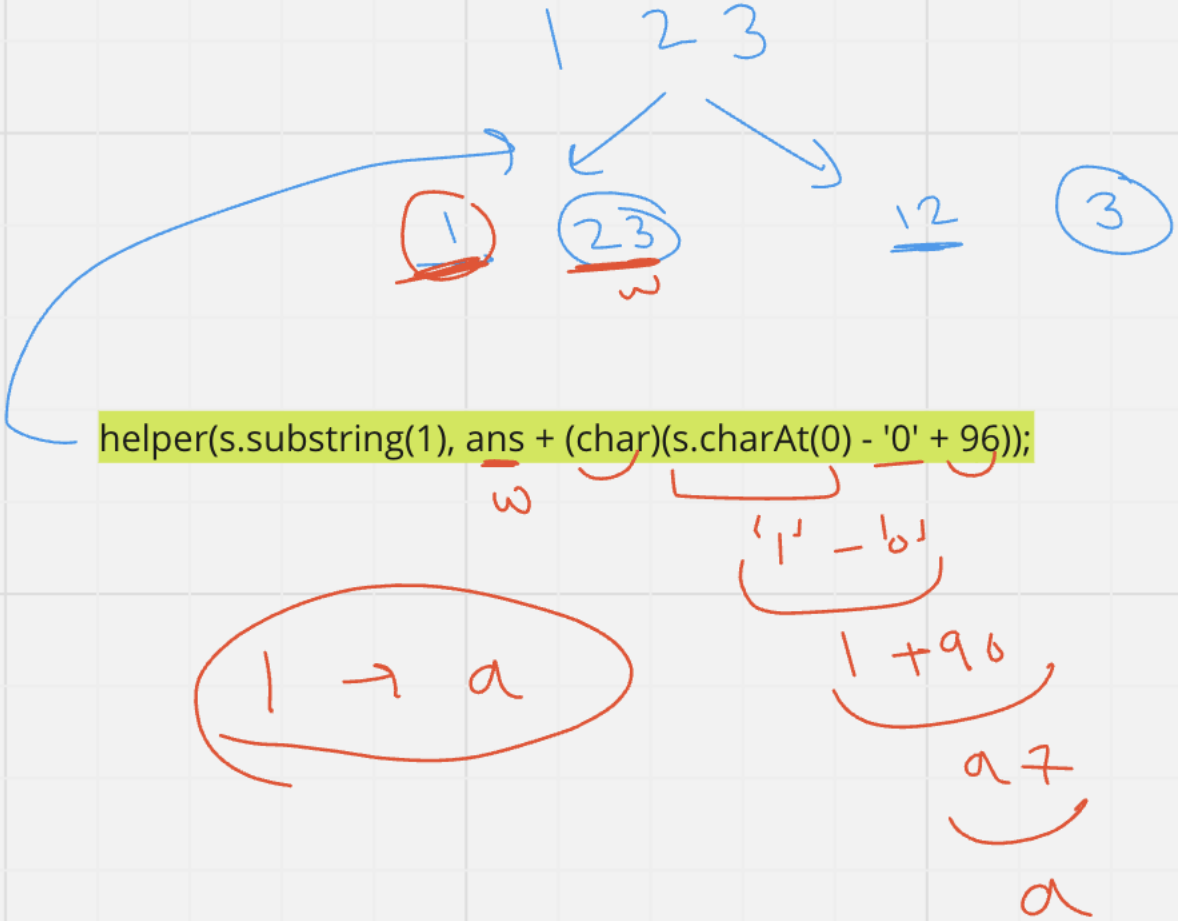
12 3

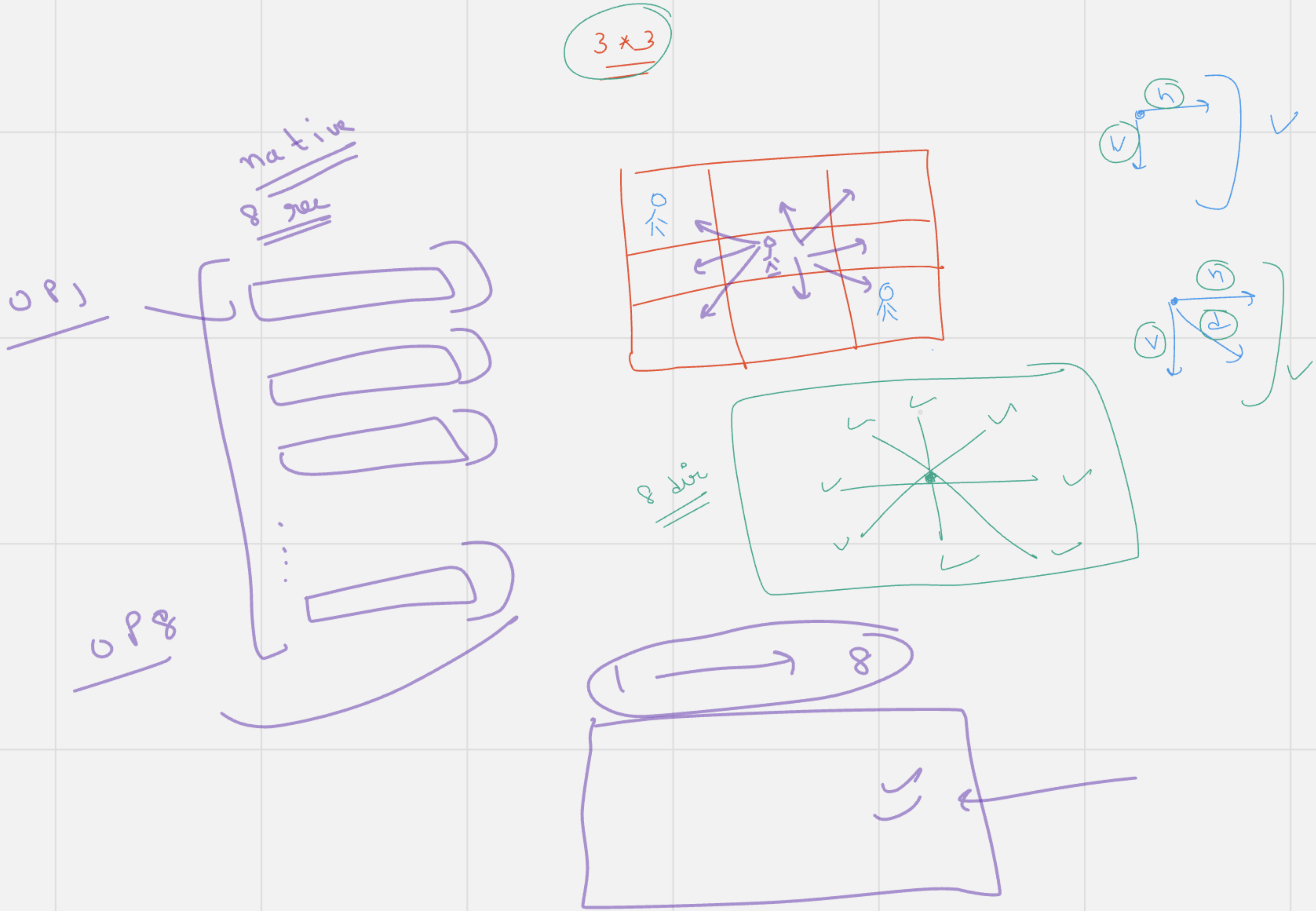
lc

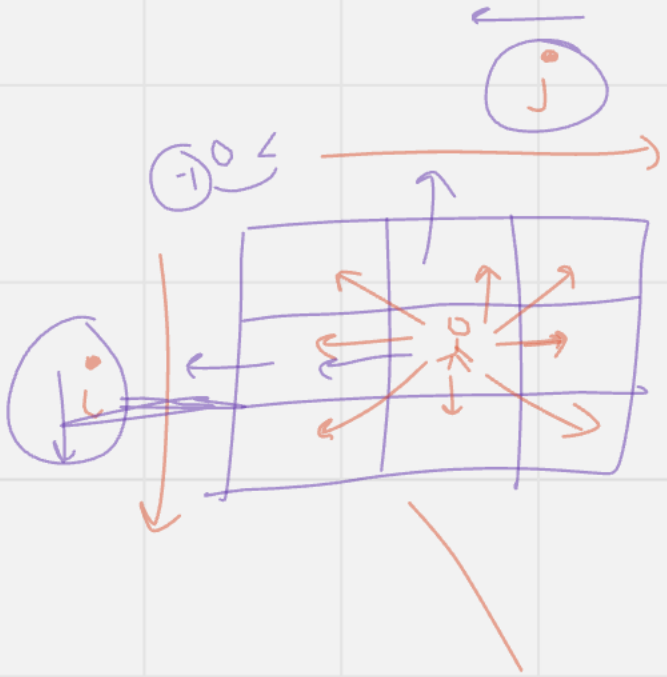
s.substring(1)



s.substring(1)
s.substring(2)
1234
2 1 4
abcd
a bcd
ab cd
bn
ud
34012

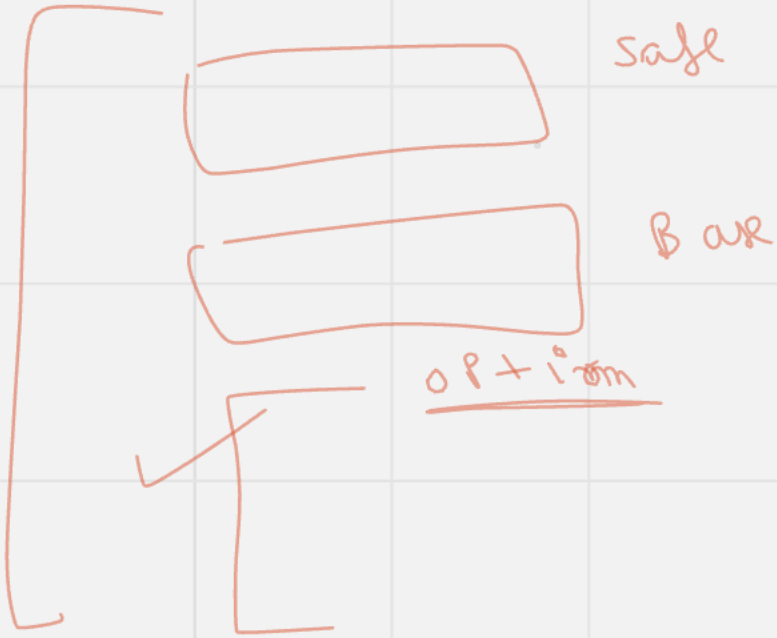

1 2 3
1
23
12
3
helper(s.substring(1), ans + (char)(s.charAt(0) - '0' + 96));
'1' - '0'
1 → a
1 + 96
97
a

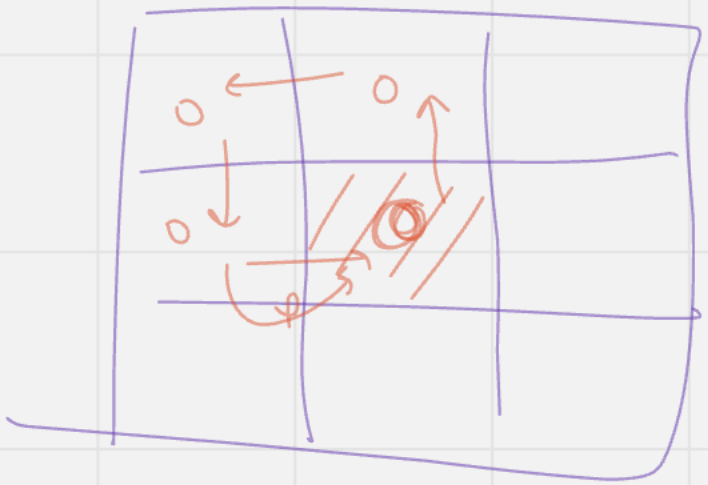

3 * 3
native
OP1
OP8
8 dir
h
v
d
1
8



i [0 , 1 , 1 , 1 , 0 , -1 , -1 , -1]

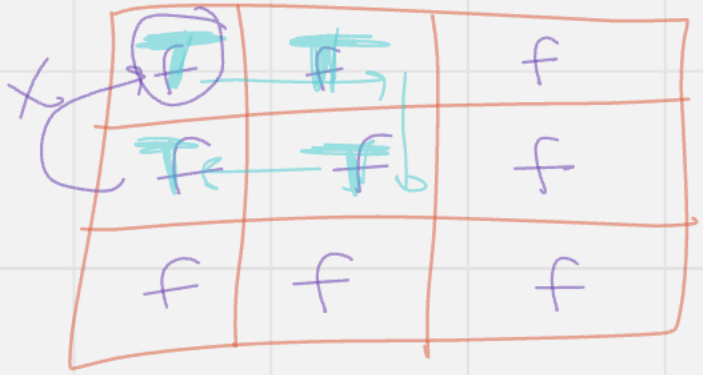
j [1 , 1 , 0 , -1 , -1 , -1 , 0 , 1]





TLE

TLE



```
static int iDir[] = {0, 1, 1, 1, 0, -1, -1, -1};
static int jDir[] = {1, 1, 0, -1, -1, -1, 0, 1};

static boolean isSafe(int i, int j, int n, int m) {
    // safe cond
    if(i > n || j > m || i < 0 || j < 0) {
        return false;
    }

    return true;
}

static int countAllPath(int i, int j, int n, int m, boolean vis[][])
{
    if(i == n && j == m) {
        return 1;
    }

    int ans = 0;

    for(int k = 0; k < 8; k++) {
        int newI = i + iDir[k];
        int newJ = j + jDir[k];
        if(isSafe(newI, newJ, n, m) && !vis[newI][newJ]) {
            vis[newI][newJ] = true;
            ans += countAllPath(newI, newJ, n, m, vis);
            vis[newI][newJ] = false;
        }

    }

    return ans;
}
```