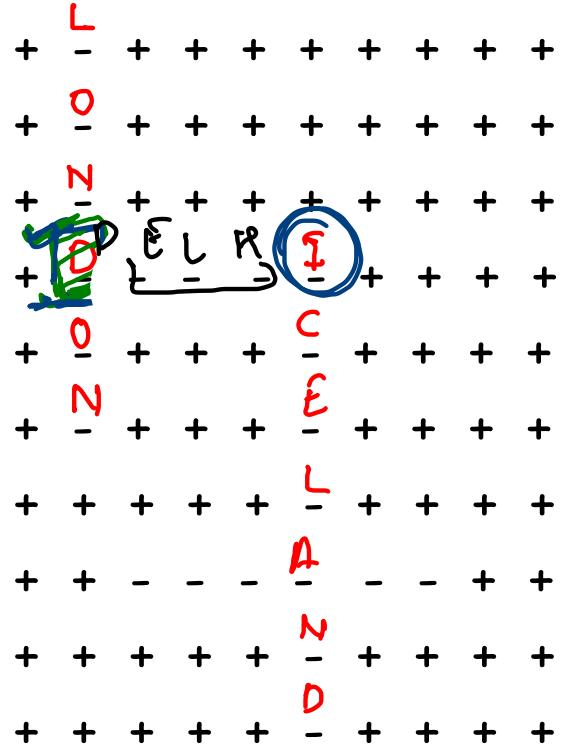


## Crossword Puzzle



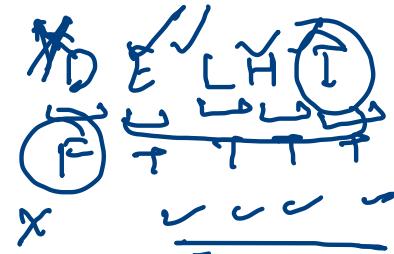
4  
LONDON  
DELHI  
ICELAND  
ANKARA



- ① Slot will completely fill by single word
- ② Intersection of slots have same character at intersection point in both word.

- ③ Two horizontal words are Not allowed
- ④ Two vertical words are Not allowed } perfectly fit

level → words  
option → cells

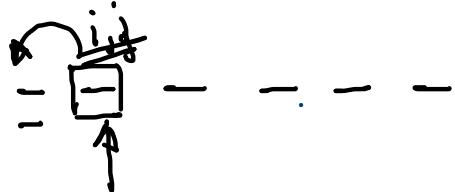


unplace is required because of backtracking

- ① Can I place word Horizontal
- ② place word Horizontally cell to Next word
- ③ unplace word Horizontally
- ④ Can I place word vertically
- ⑤ place word vertically cell to next word
- ⑥ unplace word vertically

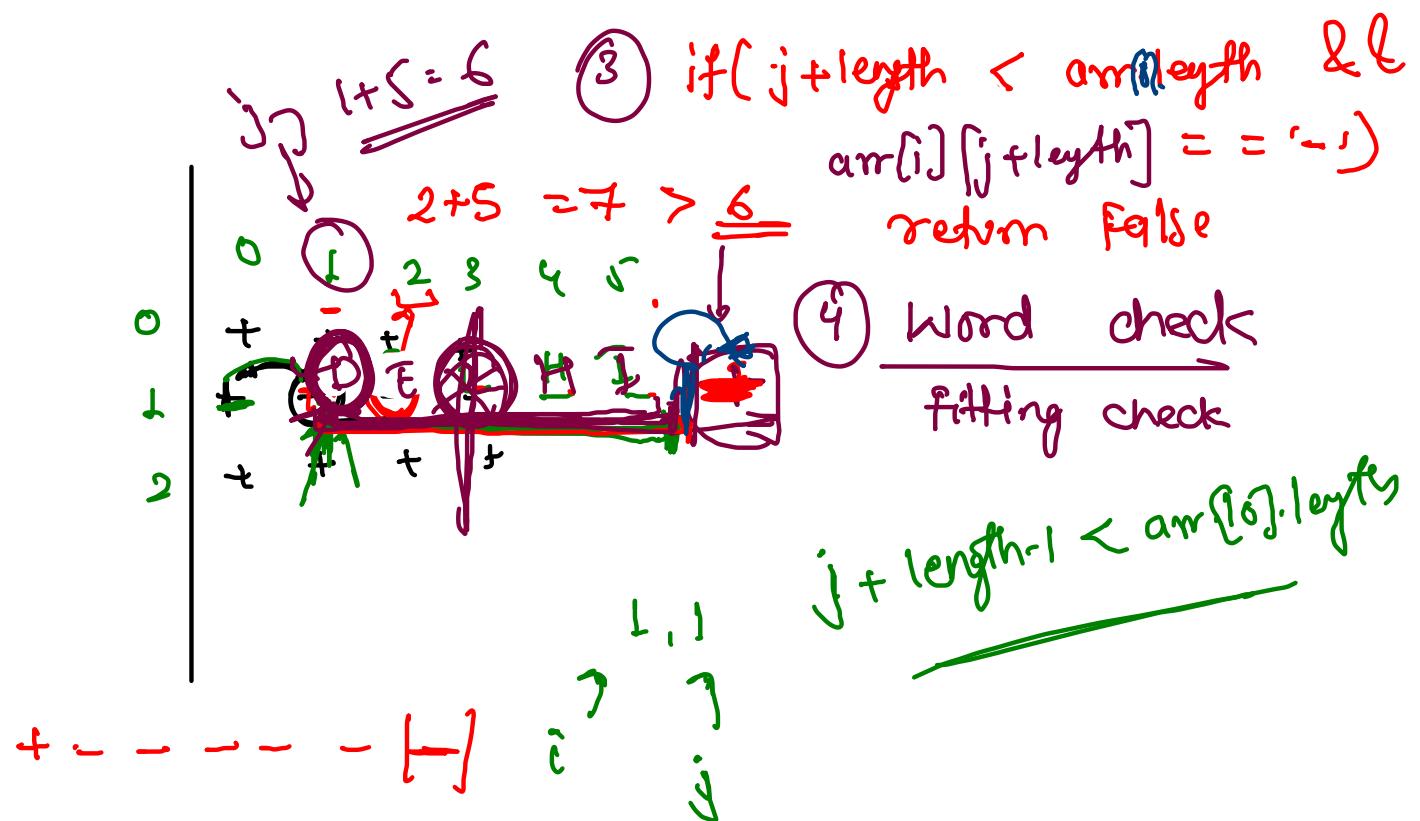
- \* cell must be starting point
  - if ( $j > 0$  &  $\text{arr}[i][j-1] == '-'$ ) return False.
- \* word will completely fit in slot
  - $\text{length} = \text{word.length()}$
  - if ( $j + \text{length} > \text{arr}[i].length()$ ) return false.

Horizontal →



Either  $j$  is '0'

otherwise  $\text{arr}[i][j-1]$  is equal to '+'



check  
 Management of starting point?  
 availability of space for word?  
 perfectly fitting?  
 In available space, all the cells are valid or not?  
 ↗ ① if( $j > 0 \&& arr[i][j-i] == '-'$ ) return False;  
 ↗ ② if( $j+length > arr[0].length$ ) return False;  
 ↗ ③ if( $j+length < arr[0].length \&& arr[i][j+length] == '-'$ )  
     return False;

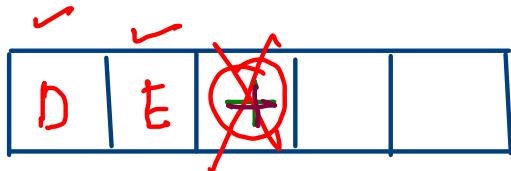
④ Word fitting check

For( $\int k=0 : k < length ; k++$ )

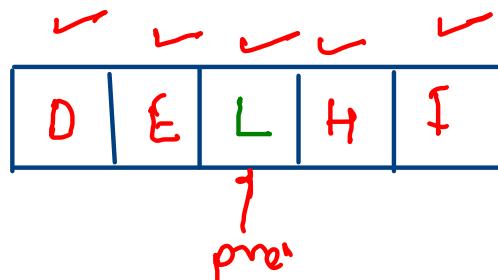
↗ if( $arr[i][j+k] != '-' \&& arr[i][j+k] \neq word.charAt(k)$ )  
     return False;

Vertical check is same

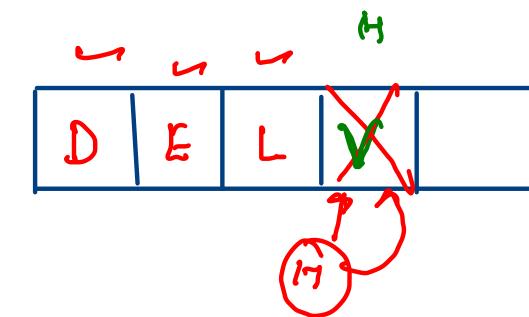
swap i, j



return False.



Existing character



return False.

→ After all ④ checks Return TRUE.

place word horizontally

word → DELHI

D	E	L	H	I
D	E	L	H	I

$\nwarrow \uparrow \swarrow \uparrow$   
 $k \quad k$

gInfo

F	T	F	F	F
---	---	---	---	---

Horizontally Boolean  
if (am[i][j+k] == '-' ) {  
    am[i][j+k] = word.charAt(k)  
    gInfo[k] = True;

D	E	L	H	I
D	E	L	H	I

F	F	T	F	F
---	---	---	---	---

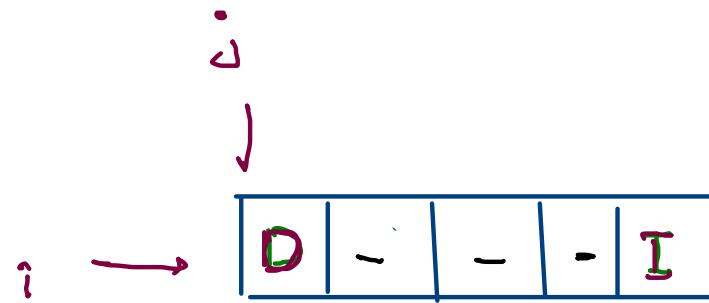
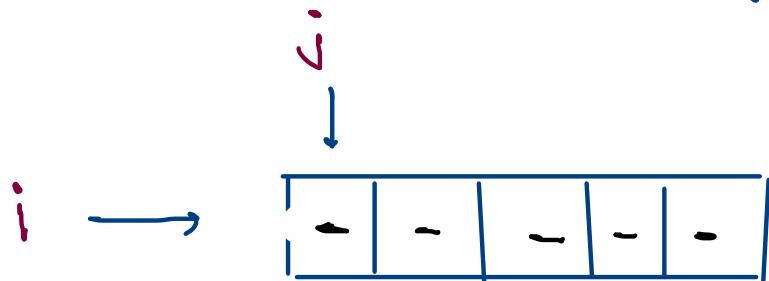
Boolean.

place character

if cell have '-'

}

Unplace word Horizontally-



T	T	T	T	T
↑				

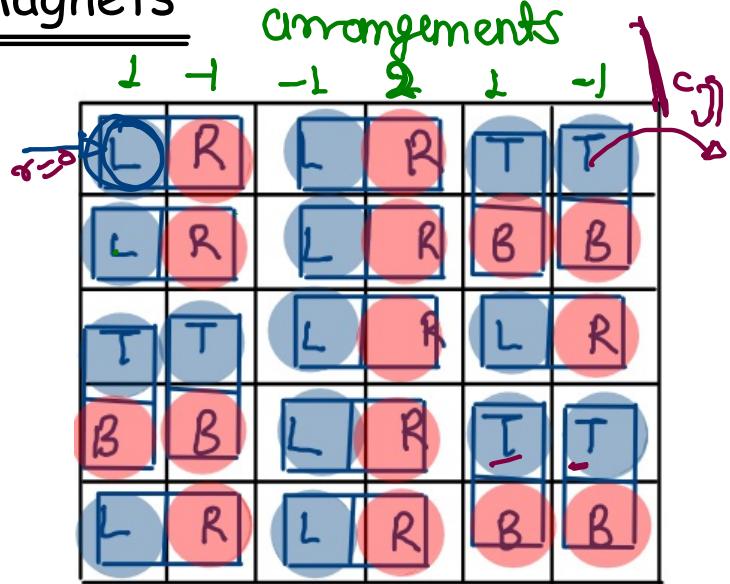
K=0

```
for(int k=0; k < ginfo.length; k++) {  
    if(ginfo[k] == true) {  
        //unplace  
        arr[i][j+k] = '_';  
    }  
}
```

3

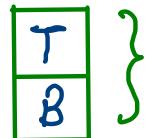
} unplace character  
if ginfo have true

## Magnets



left

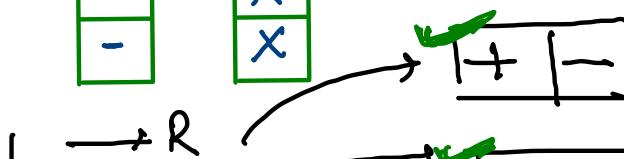
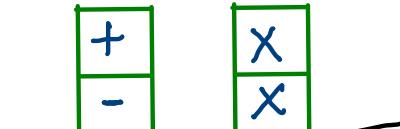
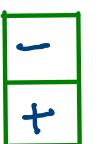
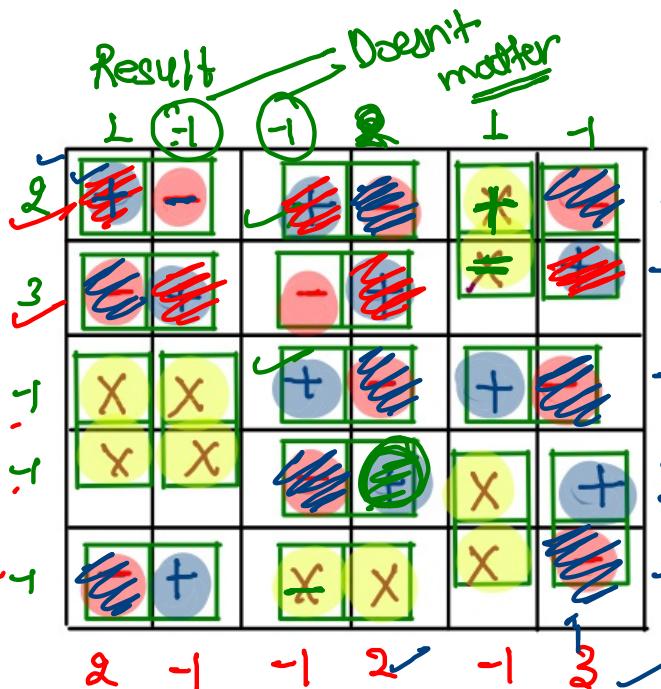
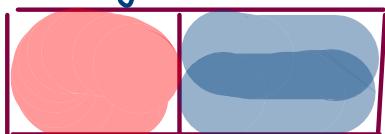
Right



top

Bottom

Magnets



$\rightarrow R$

T  
B

Data of signs

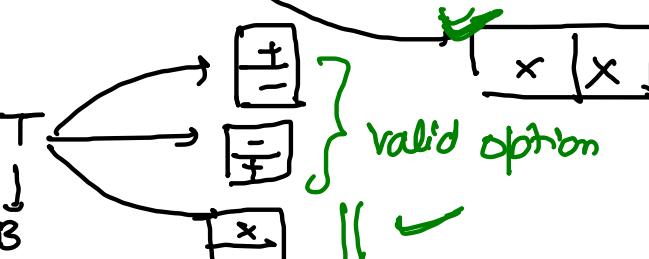
0	1	2	3	4	5
1	-1	-1	2	1	-1
2	3	-1	-1	-1	
0	1	2	3	4	5
2	-1	-1	2	-1	3
0	1	2	3	4	
-1	-1	-1	1	-1	

+ve sign top  
left Row

-ve sign Right

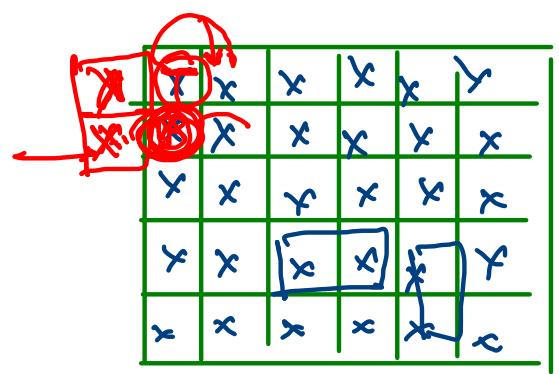
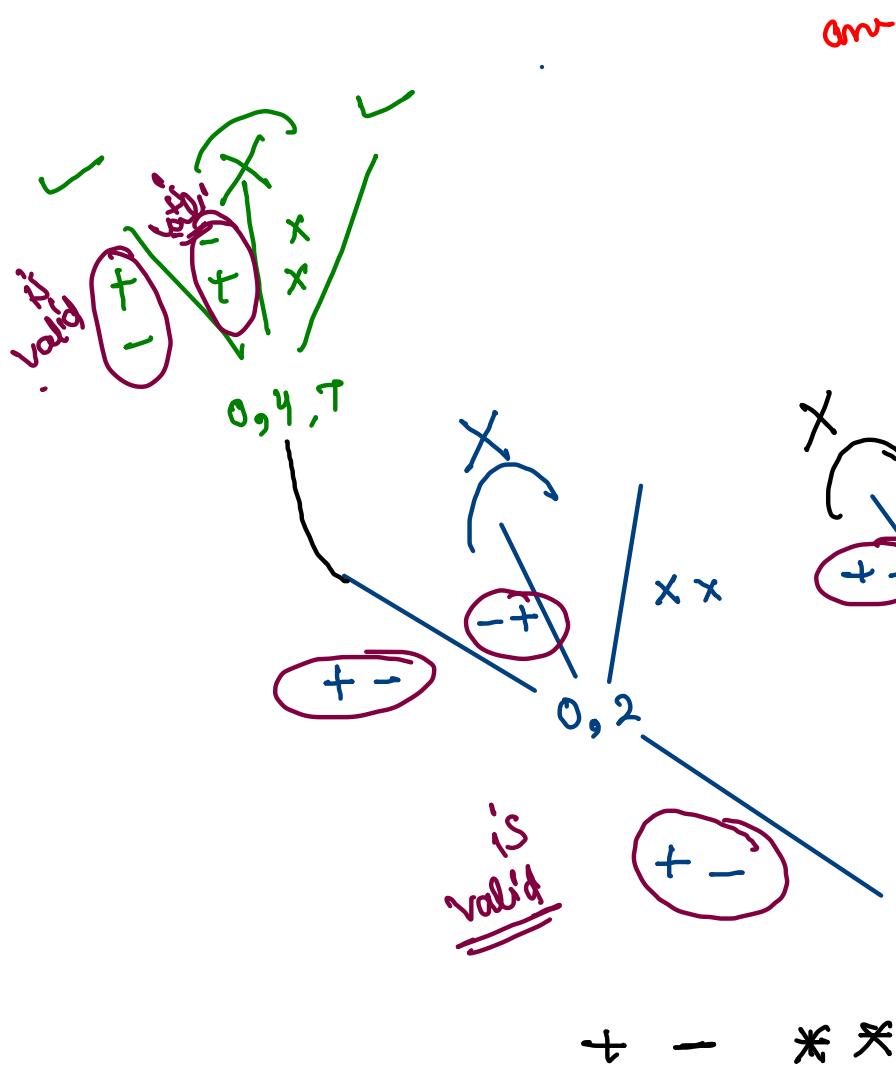
Bottom

valid option



valid option

## How to Approach toward implementation →



① if ( $\text{arr}[i][j] == \text{'L'}$ ) {  
 } filling in Result  
 }  
 else if ( $\text{arr}[i][j] == \text{'T'}$ ) {  
 }  
 No call  
 }  
 No  
 }  
 move ahead

arrangement