COL215 - Software Assignment 1- REPORT

The functions given below were created in the following order to arrive at the required output:

1) possible_bool_values(list) : the input is the possible value of boolean variables depending on the TERM and it returns them as a combination of gray code.

Eg. TERM=[None,1]   (term = b)
Input = [[0,1],[1]]   (set of possible values of a, set of possible values of b)
Output = [[0,1],[1,1]].   (set of ab coordinates in gray code)

This function identifies the larger list in the input and then iterates over its element to produce the new list. If length of both lists is same then iteration happens on both to produce output.

2) change_to_coords(l) : k- map works on gray code. This function iterates over the output of the previous function and converts coordinates from gray code to x -y system in the following manner:

Gray Code	0	1	00	01	11	10
X-Y system	0	1	0	1	2	3

3) coordinates(term) : This function finds the coordinates of the elements in the region based on TERM. It takes TERM as input. Iteration is done over elements of TERM to produce the elements of possible values(a list) of boolean variables . Then it calls the *possible_bool_values* function to find the gray code.

TERM	None	0	1
possible_values	[0,1]	[0]	[1]

For 2x2 K- map, TERM is of the form [a,b].
For 4x4 K-map, TERM is of the form [a,b,c,d]. We divide the list of possible values into two lists, one containing the possible values of a and b and the other for c and d.
For 2x4 K-map, TERM is of the form [a,b,c]. We divide the list of possible values into two lists, one containing the possible values of a and b and the other for c.

Then the function calls for *change_to_coordinate*s function to fetch the x and y coordinates separately using gray code into the following lists:

- x_coords: representing columns
- y_coords: representing rows.

We iterate over both the lists to get the (x,y) coordinates. We used (y,x) in function because the coordinates are in the form (row, column).

4) get_region (coord): This returns the top left and bottom right coordinates of the region from the set of all coordinates. The output depends on the type of region: continuous ,discontinuous row-wise, discontinuous column-wise and discontinuous both row-wise and column-wise.

5) is_legal_region (kmap_function,term) : It fetches the values present at the coordinates in the K- map via *get_region* and *coordinates* function and checks whether the region is legal or not. Then it forms another K-map from it. If the region is legal a green region is drawn on the k-map otherwise a red region is drawn. The output is tuple of the form (top left coordinate, bottom right coordinate, is_True), where is_True= True if the region is legal and false otherwise.

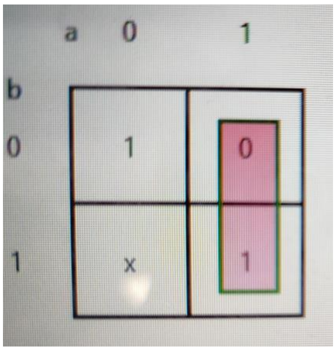
TEST CASES:

Input:

```
kmap_function : [[1,0],['x',1]]
term:   [1,None]
```

Output:

[(0, 1), (1, 1), False] (Correct Output)



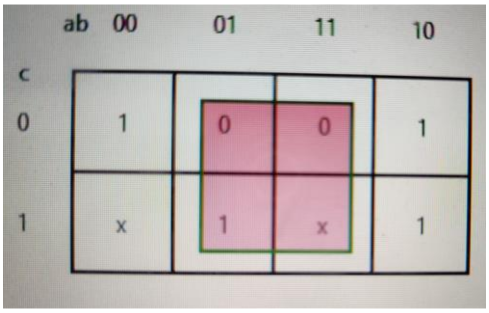
a	0	1
b		
0	1	0
1	x	1

Input :

```
kmap_function:  [[1,0,0,1],['x',1,'x',1]]
term: [None,1,None]
```

Output:

[(0, 1), (1, 2),False] (Correct Output)



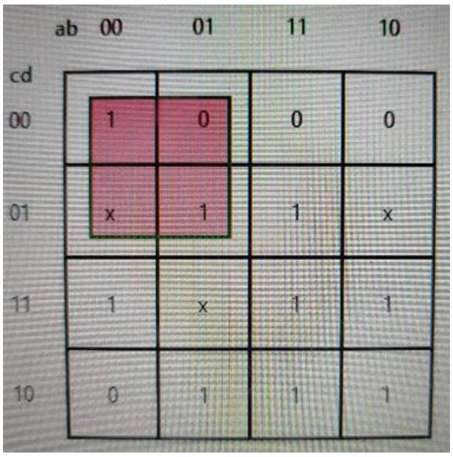
ab	00	01	11	10
c				
0	1	0	0	1
1	x	1	x	1

Input:

```
kmap_function:  [[1,0,0,0],['x',1,1,'x'],[1,'x',1,1],[0,1,1,1]]
term: [0,None,0,None]
```

Output:

[(0, 0), (1, 1), False] (Correct Output)



ab	00	01	11	10
cd				
00	1	0	0	0
01	x	1	1	x
11	1	x	1	1
10	0	1	1	1

By :
Kanishka Gajbhiye
Entry no:2021CS50131
Piyush Chauhan
Entry no: 2021CS11010