

# COL215 - Software Assignment 1

Name : Ujjwal Mehta | Entry No. : 2020CS10401

Name : Somaditya Singh | Entry No. : 2020CS10389

## Description of Function

Here in this assignment we created a `is_legal_region` function which will tell us whether a region in a given kmap grid corresponding to a given term is legal or not (means if all the contents are either 1 or x or not) and also return the corresponding left top and right bottom points for that term in the kmap grid given for 2,3 or 4 literals kmap.

## Approach for building the `is_legal_region` function

In order to build this `is_legal_region` function, our approach is to look at all the literal terms that are coming in our given term parameter of the function, then we eliminate all the rows and columns that do not correspond to those given literals and in order to do so, we have created a helper function named `row_col_invalid` which takes input as the input term and corresponding grid size and then returns all the rows or columns that should be eliminated from our `row_valid` or `col_valid` set (these are the sets that correspond to the row/column of valid kmap region of the given literal). In order to create such a function we exploited the pattern variation of the kmap rows/ columns (e.g. 00 01 11 10), so we first identify literal element and where it appears (i.e. row or column) then we iterate using while loop and store all the invalid rows/ columns and then finally return it. Now in our main function, initially we start with 2 valid sets (namely `row_valid` and `col_valid`) which contains all the rows and columns indices. Then we iterate using a for loop through our terms and then we delete the corresponding invalid rows and columns from `row_valid` and `col_valid`. Once the filtering of rows and columns is done, the only rows and columns left will correspond to the kmap region for the given input term. Then we finally iterate through all these valid region points left and check if there is any '0' appearing in this valid region in order to determine if this region is legal or not. **Hence we get whether the region corresponding to that term is legal or not.**

Now in order to find the left top and right bottom points of our kmap region we will sort the `row_valid` and `col_valid` set in order to get the valid rows and valid columns and if the rows are incrementing by 1 then that means that there is no roll over of the kmap region vertically and similarly for columns if they are incrementing by 1 then that means there is no roll over of kmap region horizontally hence the top left and right bottom points can be found by choosing the min value and max value row, column

respectively and if there are roll over in any direction (vertically or horizontally) then that direction coordinate are swapped in left top and right bottom points. **In this way we can find the end points of the region.**

---

**Test Examples Used to test the code are as follows :**

**(Here None means don't care terms 'x')**

1. The term used here is : a'

	a	0	1
b			
0		None	1
1		0	None

Here the output of our function is (0, 0) (1, 0) False

2. The term used here is : a'b'

	ab	00	01	11	10
c					
0		1	1	0	1
1		1	1	None	None

Here the output of our function is (0, 0) (1, 0) True

3. The term used here is : ac

	ab	00	01	11	10
c					
0		0	1	1	0
1		1	1	0	None

Here the output of our function is (1, 2) (1, 3) False

4. The term used here is : 1

	ab 00	01	11	10
c				
0	None	0	1	None
1	None	None	0	0

Here the output of our function is (0, 0) (1, 3) False

5. The term used here is : bc'

	ab 00	01	11	10
cd				
00	None	0	None	1
01	None	0	None	0
11	None	1	None	None
10	1	None	1	0

Here the output of our function is (0, 1) (1, 2) False

6. The term used here is : b'd'

	ab	00	01	11	10
cd					
00		1	None	0	1
01		None	None	None	0
11		0	1	0	1
10		1	None	1	1

Here the output of our function is (3, 3) (0, 0) True