8. Karnough Maps

10 POINTS

PROBLEM: Karnough or K Maps are a tool for finding the Boolean expression that produces a truth table and for simplifying the expression.

TRUTH TABLE

X	Y	C	S	C_1
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

K MAP FOR **S**

		C	
X	**Y**	**0**	**1**
0	0	0	1
0	1	1	0
1	1	0	1
1	0	1	0

K MAP FOR $\mathbf{C}_1$

		C	
X	**Y**	**0**	**1**
0	0	0	0
0	1	0	1
1	1	1	1
1	0	0	1

The K map for S takes the values in the S column in the Truth Table and puts them in a 4 by 2 table format with the values of X and Y listed vertically and the values of C listed horizontally. The K map of C_1 does the same. If you reviewed last year's All-Star questions, you should note that the X and Y values are in Gray Code order

The results in the K Map of C_1 can be converted to a Boolean expression by associating each 1 with an AND term and connecting them with OR's. Here we will use X' to denote NOT X. The 1 in row 2, column 4 is evaluated with X = 0, Y = 1 and C = 1. Since all the terms of an AND must be 1 for a result of 1, the expression X'YC produces that 1. Using the same procedure with each 1, produces the following Boolean expression for the K Map of C_1:

$$\mathbf{C_1 = X'YC + XYC' + XYC + XY'C}$$

In a like manner

$$\mathbf{S = X'Y'C + X'YC' + XYC + XY'C'}$$

Being good ACSL students, you know that the C_1 expression can be simplified. K maps also provide a method of simplification. To do this, identify all the groups of two adjacent 1's. There are three such groupings: 1_a, 1_b, and 1_c.

		C	
X	**Y**	0	1
0	0	**0**	**0**
0	1	**0**	$\mathbf{1_a}$
1	1	$\mathbf{1_c}$	$\mathbf{1_{abc}}$
1	0	**0**	$\mathbf{1_b}$

The top 1_a is formed by X=0, Y=1 and C=1.
The bottom 1_a is formed by X=1, Y=1 and C=1.
Since X can be both a 1 or a 0, it does not effect the outcome of 1_a.
Therefore, that pair can be represented by the simplified expression YC.
1_b can be represented by XC and 1_c can be represented by XY.
The entire simplified expression is **XC + YC + XY**. Do not simplify beyond the rules stated above.

In the four variable K map below an unsimplified translation could be found by evaluating each of the 8 1's as above.

		0	1	1	0	Z
W	X	0	0	1	1	Y
0	0	$\mathbf{1_a}$	**0**	**0**	**0**	
0	1	$\mathbf{1_{abc}}$	$\mathbf{1_b}$	**0**	$\mathbf{1_c}$	
1	1	$\mathbf{1_{abc}}$	$\mathbf{1_b}$	**0**	$\mathbf{1_c}$	
1	0	$\mathbf{1_a}$	**0**	**0**	**0**	

The unsimplified expression is:

F = W’X’Y’Z’+ W’XY’Z’ + WXY’Z’+ WX’Y’Z’ +W’XY’Z+WXY’Z+W’XYZ’+WXYZ’

A simplified expression for a 4 variable K map is found using the following rules:

1. Group the adjacent 1’s into blocks that contain a 2^n number of 1’s, and no 0’s.
2. Blocks can overlap
3. Opposite block boundaries are considered adjacent. That is the top and bottom rows and the first and last columns are considered adjacent.
4. Use the largest blocks first.

There are 3 possible blocks in the K map above. The 1_a block consists of the four 1’s in the first column. The 1_b block consists of the four 1’s in rows 2 and 3. The 1_c block consists of the four 1’s grouped from the first and last columns. Using the system described above the 1_a block translates to Y’Z’ because the W and X columns have both 1’s and 0’s and do not effect the result, but the Z and Y for the first column are both 0’s. The simplified translation is:

Y’Z’ + XY’ + XZ’

INPUT: There will be 5 inputs. Each line will give the size of the K-map (3 or 4 variable), and the values of the table as strings reading across the rows..

OUTPUT: Print the unsimplified Boolean expression **and** the simplified expression. One point is given for each of the 10 correct outputs. Terms can be printed in any order.

SAMPLE INPUT

1. 3, 00,01,11,01
2. 4, 1000, 1101, 1101, 1000

SAMPLE OUTPUT

1. X’YC + XYC’ + XYC + XY’C
2. XC + YC + XY
3. W’X’Y’Z’+ W’XY’Z’ + W’XY’Z+ W’XYZ’ +WXY’Z’+WXY’Z+WXYZ’+WX’Y’Z’
4. Y’Z’ + XY’ + XZ’