K-Map Minimization Code

K-Maps

The Karnaugh Map or K-Map is a method of simplifying Boolean algebra expressions. The cells in the K-Map are ordered using the Gray Code. K-Map can be created using either SOP (sum of products) or POS (product of sums). Consider the variables to be w,x,y,z. So, for 2 variables the expression will use w,x; for 3 variables the expression will use w,x,y and for 4 variables the expression will use w,x,y,z. The order of the variables is very important in the K-Map.

This is a program in Python and a test file, which takes a function of maximum 4 variables as input and gives the corresponding minimized function(s) as the output (minimized using the K-Map methodology), considering the case of Don't Care conditions.

The user will input the number of variables in the expression. The boolean function will be input in the format of a string of numbers enclosed within round brackets (separated by commas) to denote the terms of the expression and followed by the alphabet d preceding the string of numbers (separated by commas) to denote the don't care positions. In case there are no don't care positions d will be succeeded by a hyphen "-". The output will be the simplified Boolean algebra expression in the Sum of Products form. There is no need to check for invalid inputs.

Example 1:

No. of variables: 4

Function: (0,1,2,4,5,6,8,9,12,13,14) d -

Simplified expression: y'+w'z'+xz'

Example 2:

No. of variables: 4

Function: (1,3,7,11,15) d (0,2,5)

Simplified expression: yz+w'x' OR yz+w'z

References:

1. https://en.wikipedia.org/wiki/Karnaugh_map
2. Digital Design: With an Introduction of Verilog, M. Morris Mano

Additional Information

In this assignment you will implement *Quine-McCluskey* and *Petrick* methods. These methods are used together to compute the minimization of a Boolean function. The methods basically involve two steps:

1. Finding the prime implicants of a given Boolean function.

2. Use of those prime implicants in a prime implicant chart to find the essential prime implicants of the function, as well as other prime implicants that are necessary to cover the function.

The details about the algorithms can be found at https://en.wikipedia.org/wiki/ Quine%E2%80%93McCluskey_algorithm, https://en.wikipedia.org/wiki/ Petrick%27s_method and http://www.cs.columbia.edu/~cs6861/handouts/ quine-mccluskey-handout.pdf

Output of the function:

The output would be a minimized Boolean function represented as a string. The output string is formed by concatenating the terms in lexicographic order.