COL215 Assignment 2

By Adithya Bijoy and Anish Banerjee

Our Approach

We made a list l that contains the expanded expression of all the minterms (including 'Don't Cares'). Then for each minterm(a) of the true list, we go through the list l to find the term with minimum literals containing a.

We made the list l by pairing up the minterms to form pairs and the minterms that can't be paired are added to the list l. The pairs formed are added to another list (l') and we pair elements of that list and the elements that can't be paired are added to the list l. This process is continued till no more pairs can be formed.

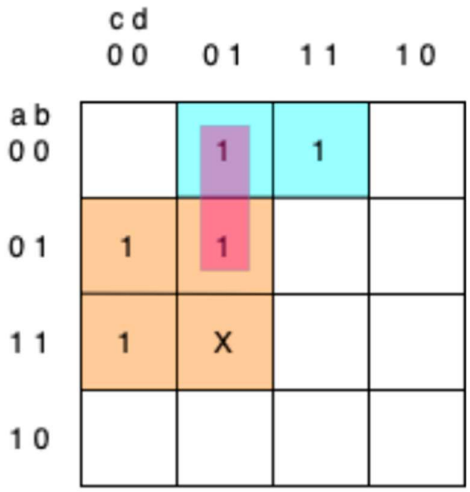
Questions

• Do all expansions result in an identical set of terms?

No, all expansions don't result in an identical set of terms. For example, in testcase 3 (see below), the term a'b'c'd can be expanded to both a'b'd and a'c'd

• Are all expansions equally good, assuming that our objective is to maximally expand each term? Explain.

No, all expansions are not equally good. For example, look at the KMap below:



Here the term a'bc'd can be expanded to a'c'd as well as bc'. The expansion to bc' is better aligned to our objective since it has lesser number of literals.

Test Cases

S.no.	Test Case	Output
1	func_TRUE = ["abcdefghij","abcdefghij'","abcdefghi'j"] func_DC = ["abcdefghi'j'"]	['abcdefgh', 'abcdefgh', 'abcdefgh']
2	func_TRUE = ["a'b'c'd'e'", "a'b'cd'e", "a'b'cde'", "a'bc'd'e'", "a'bc'd'e", "a'bc'de", "a'bc'de'", "ab'c'd'e'", "ab'cd'e'"] func_DC = ["abc'd'e'", "abc'd'e", "abc'de", "abc'de'"]	["c'd'e'", "a'b'cd'e", "a'b'cde'", "bc'", "bc'", "bc'", "bc'", "c'd'e'", "ab'd'e'"]
3	func_TRUE = ["a'bc'd'", "abc'd'", "a'b'c'd", "a'bc'd", "a'b'cd"] func_DC = ["abc'd"]	["bc'", "bc'", "a'b'd", "bc'", "a'b'd"]
4	func_TRUE = ["a'bc'd'efghijklmnopqrst", "abc'd'efghijklmnopqrst", "a'b'c'defghijklmnopqrst", "a'bc'defghijklmnopqrst", "a'b'cdefghijklmnopqrst"] func_DC = ["abc'defghijklmnopqrst"]	["bc'efghijklmnopqrst", "bc'efghijklmnopqrst", "a'b'defghijklmnopqrst", "bc'efghijklmnopqrst", "a'b'defghijklmnopqrst"]
5	func_TRUE = ["ab", "ab'", "a'b","a'b'"] func_DC = []	[None, None, None, None]