Srijan Gupta - 2020CS50444, Nischay Diwan - 2020CS50433
COL215P: Digital Logic & System Design

Boolean Function Expansion

Approach

1) Add don't care terms to true terms list. Let this list be L
2) Convert all terms to binary
3) Make a list which contains the binary representation of terms and a set which only contains the index of the term, let this list be B
4) Initialize U as an empty list.
5) While B is not empty:
   (a) Iterate over all pairs of elements in the array and see if they can be combined to form a larger region.
   (b) If so, add the binary representation of the region & union of the 2 sets stored to B.
   (c) Remove the used terms from B and add them to U.
6) For each term, take the term which contains it in its set and has the most don't care bits, i. e. the largest region.

Let n be the number of input variables.
Let N be the number of true terms.
Let M be the number of don't care terms.
The time complexity of this algorithm is $O(n(N+M)^3)$

Tests

1) Testcase 1:
   True Values : ["ab'cd'","abcd'"]
   Don't Care Values : ["a'b'cd"]
   Output : ["acd'", "acd'"]
2) Testcase 2:
   True Values : ["a'b'c'd", "a'b'c'd'", "a'b'cd'", "abcd'"]
   Don't Care Values : ["abc'd'", 'abcd', "a'bcd", "ab'cd'", "a'bc'd'", "a'b'c'd'"]
   Output : ["a'b'c'", "a'b'c'", "a'b'd'", "abd'"]
3) Testcase 3:
   True Values : ["a'bc'd'", "abc'd'", "a'b'c'd", "a'bc'd", "a'b'cd"]
   Don't Care Values : ["abc'd"]
   Output : ["bc'", "bc'", "a'c'd", "bc'", "a'b'd"]
4) Testcase 4:
   True Values : ["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'"]
   Don't Care Values : ["abc'd'e'", "abc'd'e", "abc'de", "abc'de'"]
   Output : ["c'd'e'", "a'b'cd'e", "a'b'cde'", "bc'", "bc'", "bc'", "bc'", "c'd'e'", "ab'd'e'"]
5) Testcase 5:
   True Values : ["a'b'c'd'e'f'", "a'b'cd'ef", "a'b'cde'f", "a'bc'd'e'f'", "a'bc'd'ef", "a'bc'def", "a'bc'de'f", "ab'c'd'e'f", "ab'cd'e'f'"]
   Don't Care Values : ["abc'd'e'f", "abc'd'ef", "abc'def", "abc'de'f'"]
   Output : ["a'c'd'e'f'", "a'b'cd'ef", "a'b'cde'f", "a'c'd'e'f'", "bc'ef", "bc'ef", "a'bc'df", "ac'd'e'f", "ab'cd'e'f'"]
6) Testcase 6:
   True Values : ["a'b'c'd'e'", "a'b'cd'e", "a'b'cde'", "a'b'cde", "a'bc'd'e'", "a'bc'd'e", "a'bc'de", "a'bc'de'", "ab'c'd'e'", "ab'c'd'e", "ab'c'de'", "ab'c'de", "ab'cd'e'", "ab'cd'e", "ab'cde'", "ab'cde", "abc'd'e'", "abc'd'e", "abc'de'", "abc'de", "abcd'e'", "abcd'e", "abcde'", 'abcde']
   Don't Care Values : []

Output : ["c'd'e'", "b'ce", "b'cd", "b'ce", "bc'", "bc'", "bc'", "bc'", 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a', 'a']

7) Testcase 7:
True Values : ['abc', "a'bc", "abc'", "ab'c'"]
Don't Care Values : ["ab'c"]
Output : ['a', 'bc', 'a', 'a']
8) Testcase 8:
True Values : ["abc'defg", "a'bc'de'fg", "a'bc'de'f'g'", "ab'c'de'f'g'", "abc'd'efg", "a'b'cdef'g'", "abcd'e'f'g", "ab'cd'e'fg", "abc'de'fg'", "a'b'cd'ef'g", "a'bcd'e'f'g", "abcde'f'g", "a'bc'de'f'g", "ab'c'd'efg", "a'b'cd'e'fg","ab'cde'f'g'", "abc'd'efg'", "abc'defg'", "abc'd'e'f'g", "a'bcd'e'fg'"]
Don't Care Values : ["ab'cd'ef'g'", "a'b'cdefg", "a'b'cd'e'f'g'", "abcdefg'", "a'bcdefg'", "a'b'c'defg", "ab'c'd'efg'", "a'b'c'd'e'f'g", "ab'cd'efg'", "abcd'ef'g'", "a'b'cd'e'f'g", "ab'c'd'ef'g", "ab'c'def'g'", "a'bc'd'efg", "a'bc'def'g", "a'b'cd'ef'g'", "abcd'e'fg", "a'bc'd'e'fg", "a'bc'd'e'f'g", "ab'cdef'g'", "a'bcd'e'fg", "a'b'c'de'f'g", "abcd'e'fg'", "a'bcde'f'g", "abc'd'ef'g", "a'bcd'efg'", "ab'cde'f'g", "a'b'cdefg'", "ab'c'def'g", "a'bc'de'fg'", "a'b'c'd'efg", "abc'd'e'fg", "a'bcdefg", "a'bcd'efg", "a'b'c'def'g'", "ab'c'de'fg'","abc'd'e'fg'", "a'bcde'f'g'", "a'b'c'de'fg'", "ab'cd'e'f'g", "abc'de'fg", "ab'cde'fg", "a'b'c'de'f'g'", "abcd'ef'g", "a'bc'defg"]
Output : ["abc'f", "bc'fg", "a'bc'de'", "ab'df'g'", "abc'f", "b'cef'g'", "bd'e'g", "cd'e'g", "abc'f", "a'b'cd'f'", "bd'e'g", "bce'f'g", "a'bc'de'", "ac'd'ef", "cd'e'g", "ab'df'g'", "abc'f", "abc'f", "bd'e'g", "bcd'e'f"]
9) Testcase 9:
True Values : ["abc'de'fghi'j", "a'bc'd'e'fghij'", "ab'c'd'e'fg'hi'j", "a'b'c'd'e'fghi'j'", "a'b'cd'e'f'g'h'i'j'", "a'bcde'f'g'hij'", "abcd'e'fg'hi'j", "abcdef'gh'ij'", "abcdef'ghi'j'", "ab'c'de'fghij'", "a'bcd'e'f'g'h'ij", "ab'cdefgh'i'j'", "a'b'cd'e'fgh'ij'", "ab'c'd'e'f'gh'ij", "a'bcde'fg'hij'", "ab'c'defg'h'i'j", "abc'def'g'h'i'j'", "abc'de'fg'h'ij", "abc'de'fg'hij'", "ab'c'd'e'f'g'hi'j", "a'b'cd'e'fg'h'i'j'", "ab'cde'f'gh'i'j'", "a'b'c'd'e'fg'hi'j'", "ab'cdef'ghi'j", "a'bc'd'ef'g'hi'j", "a'bcde'fghi'j'", "abcd'ef'g'h'i'j", "ab'c'd'e'fg'h'i'j'"]
Don't Care Values : ["ab'c'd'e'fg'h'ij'", "abcdef'g'h'ij", "ab'c'def'ghi'j", "ab'cde'fghi'j", "a'bcd'efghij", "a'bc'd'e'fghi'j", "ab'c'd'e'f'g'h'i'j", "a'bc'd'ef'ghi'j'", "a'b'cde'f'gh'ij'", "a'b'cd'e'fg'hij", "ab'cd'ef'g'hi'j'", "a'b'cde'f'ghij'", "abc'defgh'i'j", "a'bcd'ef'g'h'ij'", "abcde'fg'h'i'j'", "a'bc'd'efg'hi'j", "ab'cd'ef'gh'ij'", "ab'cdefg'h'i'j'", "a'b'cd'efghij", "a'bc'd'efg'h'ij'", "ab'c'de'fg'h'i'j'", "abc'd'ef'gh'i'j'", "a'bc'd'e'f'gh'i'j", "a'b'cdef'gh'ij", "a'b'c'def'g'h'ij'", "a'bc'def'ghij", "a'b'cd'e'fg'h'i'j", "abc'd'efgh'i'j", "ab'c'def'g'hi'j'", "ab'cd'e'f'ghij'", "a'b'cde'fgh'ij", "ab'c'de'f'gh'i'j'", "ab'c'de'fghij", "ab'c'd'e'fg'hij'", "a'bc'def'g'h'i'j", "a'bcd'efghij'", "a'b'c'd'e'f'ghi'j", "a'b'cdefgh'i'j", "a'b'cd'e'f'g'hi'j'", "ab'c'defg'hi'j", "ab'cdefghi'j'", "abcde'fg'hij", "a'b'c'd'e'fghij", "ab'cd'e'fghij", "a'bc'def'gh'ij'", "ab'cde'fg'h'ij", "a'b'c'd'efg'hi'j", "abcdefg'hi'j", "abcde'fgh'i'j'"]
Output : ["abc'de'fghi'j", "a'bc'd'e'fghij'", "ab'c'd'e'g'hi'j", "a'b'c'd'e'fhi'j'", "a'b'cd'e'g'h'i'j'", "a'bcde'g'hij'", "abcd'e'fg'hi'j", "abcdef'gh'ij'", "abcdef'ghi'j'", "ab'c'de'fghi", "a'bcd'e'f'g'h'ij", "ab'cdefh'i'j'", "a'b'cd'e'fgh'ij'", "ab'c'd'e'f'gh'ij", "a'bcde'g'hij'", "ab'c'defg'i'j", "abc'def'g'h'i'j'", "abc'de'fg'h'ij", "abc'de'fg'hij'", "ab'c'd'e'g'hi'j", "a'b'cd'e'g'h'i'j'", "ab'de'f'gh'i'j'", "a'b'c'd'e'fhi'j'", "ab'def'ghi'j", "a'bc'd'eg'hi'j", "a'bcde'fghi'j'", "abcd'ef'g'h'i'j", "ab'c'd'e'fg'h'j'"]
10) Testcase 10:
True Values : ["abc'de'fghi'j", "a'bc'd'e'fghij'", "ab'c'd'e'fg'hi'j", "a'b'c'd'e'fghi'j'", "a'b'cd'e'f'g'h'i'j'", "a'bcde'f'g'hij'", "abcd'e'fg'hi'j", "abcdef'gh'ij'", "a'bc'defgh'i'j'", "a'bc'defghij'", "a'b'cd'e'f'ghi'j", "a'bc'd'e'f'gh'ij'", "a'b'c'de'f'ghij", "a'bcdef'g'h'ij'", "ab'cde'fgh'ij", "ab'cd'ef'ghij'", "a'bcdefg'h'i'j", "ab'c'd'ef'gh'ij", "abc'def'gh'i'j'", "a'bcde'f'g'h'i'j'", "a'b'c'd'efg'hij'", "ab'cdefg'hij'", "a'b'c'd'ef'ghi'j'", "a'b'cd'e'f'ghij", "abc'defghi'j", "abcd'efghi'j'", "a'bc'de'f'g'hi'j'", "a'bc'def'g'hij'", "ab'c'd'efghij", "a'b'cd'ef'g'hi'j", "ab'cd'ef'ghi'j'", "a'bcdef'gh'ij'", "a'b'c'de'fghij", "abcd'e'fg'h'i'j'", "a'b'c'd'e'f'g'h'i'j'", "a'bc'd'e'f'gh'ij", "a'b'c'd'e'f'g'hij'", "a'b'cde'f'gh'i'j'", "a'bc'defg'hi'j", "a'b'cd'ef'g'hij'", "ab'cde'f'g'hij'", "a'b'c'd'ef'g'hij'", "a'b'c'de'fg'h'ij", "a'bc'def'ghi'j", "a'b'c'd'e'fg'h'ij'", "a'b'c'de'f'gh'i'j", "ab'c'd'ef'gh'i'j'", "a'b'cde'fghij'", "a'bc'd'efghij'", "a'b'cd'efg'h'ij'", "a'b'cd'ef'ghi'j", "a'bcde'f'g'hi'j'", "a'b'c'd'e'fg'h'i'j'", "a'bcd'e'fghi'j", "ab'c'd'e'f'ghi'j", "a'b'cde'fg'hij'", "abcdef'g'hij", "a'b'cd'e'f'g'hij", "a'b'c'de'fg'h'i'j'", "a'bcde'f'ghi'j'", "ab'c'def'g'hij", "a'bc'de'f'gh'i'j", "abc'def'gh'i'j", "a'bcde'fghij'", "abc'd'e'fg'h'i'j",

”a’bc’d’e’f’g’hi’j’”, ”a’b’c’defg’h’i’j”, ”abc’de’f’gh’ij’”, ”a’b’c’def’g’h’i’j’”, ”abcde’f’g’hi’j’”, ”a’b’c’d’ef’gh’i’j”,
”ab’cd’e’fg’hij”, ”ab’cde’f’g’hi’j”, ”abcdef’ghi’j’”, ”ab’c’de’fghij’”, ”a’bcd’e’f’g’h’ij”, ”ab’cdefgh’i’j’”,
”a’b’cd’e’fgh’ij’”, ”ab’c’d’e’f’gh’ij”, ”a’bcde’fg’hij’”, ”ab’c’defg’h’i’j”, ”abc’def’g’h’i’j’”, ”abc’de’fg’h’ij”,
”abc’de’fg’hij’”, ”ab’c’d’e’f’g’hi’j”, ”a’b’cd’e’fg’h’i’j’”, ”ab’cde’f’gh’i’j’”, ”a’b’c’d’e’fg’hi’j’”, ”ab’cdef’ghi’j”,
”a’bc’d’ef’g’hi’j”, ”a’bcde’fghi’j’”, ”abcd’ef’g’h’i’j”, ”ab’c’d’e’fg’h’i’j’”]
Don’t Care Values : [[”ab’c’d’e’fg’h’ij’”, ”abcdef’g’h’ij”, ”ab’c’def’ghi’j”, ”ab’cde’fghi’j”, ”a’bcd’efghij”,
”a’bc’d’e’fghi’j”, ”ab’c’d’e’f’g’h’i’j”, ”a’bc’d’ef’ghi’j’”, ”a’b’cde’f’gh’ij’”, ”a’b’cd’e’fg’hij”, ”ab’cd’ef’g’hi’j’”,
”a’b’cde’f’ghij’”, ”abc’defgh’i’j”, ”a’bcd’ef’g’h’ij’”, ”abcde’fg’h’i’j’”, ”a’bc’d’efg’hi’j”, ”ab’cd’ef’gh’ij’”,
”ab’cdefg’h’i’j’”, ”a’b’cd’efghij”, ”a’bc’d’efg’h’ij’”, ”ab’c’de’fg’h’i’j’”, ”abc’d’ef’gh’i’j’”, ”a’bc’d’e’f’gh’i’j”,
”a’b’cdef’gh’ij”, ”a’b’c’def’g’h’ij’”, ”a’bc’def’ghij”, ”a’b’cd’e’fg’h’i’j”, ”abc’d’efgh’i’j”, ”ab’c’def’g’hi’j’”,
”ab’cd’e’f’ghij’”, ”a’b’cde’fgh’ij”, ”ab’c’de’f’gh’i’j’”, ”ab’c’de’fghij”, ”ab’c’d’e’fg’hij’”, ”a’bc’def’g’h’i’j”,
”a’bcd’efghij’”, ”a’b’c’d’e’f’ghi’j”, ”a’b’cdefgh’i’j”, ”a’b’cd’e’f’g’hi’j’”, ”ab’c’defg’hi’j”, ”ab’cdefghi’j’”,
”abcde’fg’hij”, ”a’b’c’d’e’fghij”, ”ab’cd’e’fghij”, ”a’bc’def’gh’ij’”, ”ab’cde’fg’h’ij”, ”a’b’c’d’efg’hi’j”,
”abcdefg’hi’j”, ”abcde’fgh’i’j’”, ”ab’cd’e’f’g’h’ij’”, ”a’bc’d’e’fgh’i’j”, ”ab’cd’e’f’gh’i’j’”, ”a’b’c’de’f’ghij’”,
”a’bcdefghij”, ”abcd’e’f’g’hij”, ”a’bcde’f’ghij’”, ”a’bc’de’f’ghi’j’”, ”a’b’c’d’ef’g’h’ij’”, ”a’b’c’defg’hij”,
”a’b’c’d’ef’ghi’j”, ”a’b’c’defghij”, ”a’bcd’efgh’ij”, ”a’bc’d’ef’g’h’ij’”, ”ab’cd’ef’g’h’i’j”, ”abc’de’f’g’hi’j’”,
”abc’de’f’g’hij’”, ”a’b’cdefgh’ij”, ”a’bc’d’e’fgh’ij”, ”a’b’cd’ef’g’h’ij”, ”a’bc’de’fgh’i’j”, ”a’bc’de’fgh’i’j’”,
”a’b’cd’ef’g’h’i’j’”, ”a’b’cde’fg’hi’j’”, ”abcd’ef’gh’i’j’”, ”a’bc’d’efg’h’ij”, ”a’b’c’de’fg’h’ij’”, ”a’b’c’d’efg’hij”,
”a’bc’d’efg’hi’j’”, ”a’bcdef’g’h’ij”, ”abc’d’efghi’j’”, ”ab’c’defghi’j’”, ”ab’c’de’f’ghij’”, ”ab’c’d’ef’ghij”,
”a’b’c’de’f’g’hij’”, ”a’bcd’ef’gh’i’j’”, ”abc’d’e’fg’hij’”, ”a’bc’d’efgh’i’j”, ”a’b’c’def’gh’i’j’”, ”ab’c’de’f’g’h’i’j’”,
”a’bcdef’g’hij”, ”abcd’efg’hi’j’”, ”a’b’c’d’e’f’gh’ij”, ”a’b’c’d’e’f’g’h’i’j’”, ”a’bcde’fg’hi’j’”, ”ab’cdefghi’j”,
”ab’c’d’efgh’i’j’”, ”abc’d’efg’hi’j’”, ”a’bcdefg’h’ij’”, ”ab’cd’efg’hi’j’”, ”abc’d’e’fghi’j’”, ”abcd’efg’h’i’j”,
”abc’d’efg’h’i’j”, ”abcde’f’g’hij’”, ”a’bcde’fg’h’i’j”, ”abc’d’e’f’ghij”, ”a’b’cdef’ghij’”, ”a’b’cde’fg’h’ij”,
”a’b’cde’fgh’i’j’”, ”abc’de’fg’hi’j’”, ”a’b’c’d’efg’h’i’j’”, ’abcdefghij”, ”ab’cde’fg’h’ij’”, ”a’bcd’e’fg’h’ij”,
”abc’de’f’g’h’ij’”, ”ab’c’d’efg’h’i’j”, ”a’b’cde’fghi’j’”, ”a’bc’d’e’f’g’h’ij’”, ”a’b’cde’f’ghi’j’”, ”a’bcd’ef’ghi’j’”,
”ab’c’def’g’h’i’j’”, ”a’b’c’defg’h’i’j’”, ”abcde’f’g’h’i’j”, ”a’bcd’efg’h’i’j”, ”a’b’cdef’gh’ij’”, ”abcd’e’fgh’ij”,
”abc’d’efg’h’ij’”, ”ab’cd’e’f’ghi’j”, ”abcd’efgh’ij”, ”a’b’c’de’f’ghi’j’”, ”a’bc’defg’h’i’j’”, ”a’bc’de’fg’h’i’j”,
”abcd’ef’g’h’ij’”, ”ab’cd’e’f’g’hi’j”, ”a’bc’d’ef’g’hij”, ”abc’def’ghij”, ”a’bcdef’g’hi’j”, ”a’bcd’efg’h’i’j’”,
”a’bc’def’g’h’ij’”, ”a’b’cd’efg’hi’j”, ”ab’c’de’fgh’i’j”, ”abc’de’fgh’ij”, ”a’bc’de’f’g’hij”, ”abcde’fgh’ij’”,
”a’b’cd’efg’hij”, ”a’b’c’d’ef’g’h’ij”, ”a’b’cde’f’g’h’ij”, ”a’bcd’e’fg’hi’j”, ”a’bcd’e’fg’hij”, ”ab’c’d’efgh’ij”,
”ab’cde’f’ghi’j’”, ”ab’c’de’f’gh’i’j”, ”a’b’cdef’gh’i’j”, ”a’b’cde’f’g’h’ij’”, ”ab’c’d’e’fg’hij”, ”ab’c’d’e’fghij’”,
”a’bc’d’e’f’g’h’i’j”, ”abcd’e’f’ghi’j’”, ”a’b’cde’f’g’hij”, ”abc’d’e’f’g’hij”, ”a’b’cd’e’f’ghij’”, ”a’bcd’efg’h’ij”,
”a’bcd’ef’g’hi’j”, ”a’bcd’efgh’i’j’”, ”a’b’c’d’ef’g’h’ij”, ”a’b’cde’f’g’hij”, ”abcd’ef’g’h’ij”, ”ab’cd’efg’hij’”,
”ab’c’d’e’f’ghij’”, ”a’bc’d’ef’g’h’i’j”, ”ab’c’d’e’f’gh’i’j’”, ”abcd’e’fghij”, ”abcd’e’f’ghij”, ”a’b’c’d’e’f’ghij”,
”abcdef’g’h’i’j’”, ”a’bcd’ef’gh’i’j”, ”a’bc’d’e’f’g’hi’j”, ”abcde’f’g’hij”, ”a’b’cd’efghij”, ”ab’cdefgh’i’j’”,
”a’b’cd’e’fgh’ij’”, ”ab’c’d’e’f’gh’ij”, ”a’bcde’fg’hij’”, ”ab’c’defg’h’i’j”, ”abc’def’g’h’i’j’”, ”abc’de’fg’h’ij”,
”abc’de’fg’hij’”, ”ab’c’d’e’f’g’hi’j”, ”a’b’cd’e’fg’h’i’j’”, ”ab’cde’f’gh’i’j’”, ”a’b’c’d’e’fg’hi’j’”, ”ab’cdef’ghi’j”,
”a’bc’d’ef’g’hi’j”, ”a’bcde’fghi’j’”, ”abcd’ef’g’h’i’j”, ”ab’c’d’e’fg’h’i’j’”]
Output : [”abc’dfghi’j”, ”a’bc’d’fghij’”, ”ab’c’d’e’g’hi’j”, ”a’b’c’d’e’fhi’j’”, ”a’b’d’e’g’h’i’j’”, ”a’bcde’hj’”,
”bcd’e’fg’hi’j”, ”bcdef’gh’ij’”, ”a’bc’dfgh’i’j’”, ”a’bc’efghij’”, ”a’b’d’e’f’ghj”, ”a’bc’d’e’f’gh’i”, ”b’c’de’ghij”,
”a’bdef’h’ij’”, ”b’cde’fh’ij”, ”ab’cd’ef’ghj’”, ”a’bcdfg’h’i’j”, ”ab’c’d’egij”, ”abc’def’gh’i’”, ”a’bcde’f’g’i’j’”,
”a’b’c’d’eg’hij’”, ”ab’cdefhij’”, ”a’c’d’ef’ghi’j’”, ”a’b’d’e’f’ghj”, ”abc’dfghi’j”, ”abd’efhi’j’”, ”a’bde’f’hi’j’”,
”a’bc’def’g’ij’”, ”ab’c’d’egij”, ”a’b’cd’ef’hi’j”, ”ab’cd’ef’ghj’”, ”a’bdef’h’ij’”, ”b’c’de’ghij”, ”abce’fg’h’i’j’”,
”a’b’d’e’g’h’i’j’”, ”a’bc’d’e’gh’j”, ”a’b’c’d’f’g’hij’”, ”b’cde’f’gi’j’”, ”a’bc’efg’hi’j”, ”a’b’d’ef’g’ij’”, ”acde’f’g’hij
”a’b’d’ef’g’ij’”, ”a’b’c’de’fg’h’i”, ”a’bc’def’ghj”, ”a’b’c’e’fg’h’j’”, ”a’c’de’f’gh’i’j”, ”ac’d’ef’gh’i’j’”, ”a’cde’fhj’’
”a’bc’d’fghij’”, ”a’b’cd’eg’h’ij’”, ”a’b’d’f’ghi’j”, ”a’bcde’hj’”, ”a’b’d’e’g’h’i’j’”, ”a’bd’e’fghi’j”, ”b’d’e’f’ghi’j”,
”a’cde’fhj’”, ”bcdef’g’ij”, ”a’b’cd’e’f’hij”, ”a’b’c’e’fg’h’j’”, ”a’bcde’hj’”, ”ab’c’def’g’hij”, ”a’bc’e’gh’i’j”,
”abc’def’gh’i’”, ”a’bcde’hj’”, ”abc’d’fg’h’i’j”, ”a’bc’e’f’g’hi’j’”, ”b’c’defg’h’i’j”, ”abc’de’f’h’ij’”, ”a’b’c’def’g’h
”bcde’f’g’hj’”, ”a’b’c’d’ef’gi’j”, ”b’cd’e’fg’hij”, ”ab’ce’f’g’hi’j”, ”abcef’ghi’j’”, ”ab’c’de’ghi”, ”a’bcd’e’g’h’ij”,
”ab’cdefh’i’j’”, ”a’cd’e’fgh’ij’”, ”ab’c’d’f’gh’ij”, ”a’bcde’hj’”, ”b’c’defg’h’i’j”, ”abc’def’h’i’j’”, ”abc’de’fh’ij”,

”abc’de’g’hj’”, ”ab’d’e’f’hi’j”, ”a’b’d’e’g’h’i’j’”, ”b’cde’f’gi’j’”, ”a’b’c’d’e’fhi’j’”, ”ab’def’ghi’j”, ”a’bc’d’f’g’i’j”
”a’bcde’hj’”, ”acd’ef’g’h’i’j”, ”b’c’d’e’fg’h’j’”]

Screenshot of above outputs:

Thought Questions

1) No all expansions do not lead to identical set of terms because there are more than one ways for terms to expand, for equally good expansion.
2) No all expansions are not equally good, because some expansions can lead to sub-optimal expansion like in case of testcase 7, abc could combine with a’bc to give bc, or abc could combine with rest of three to give a.