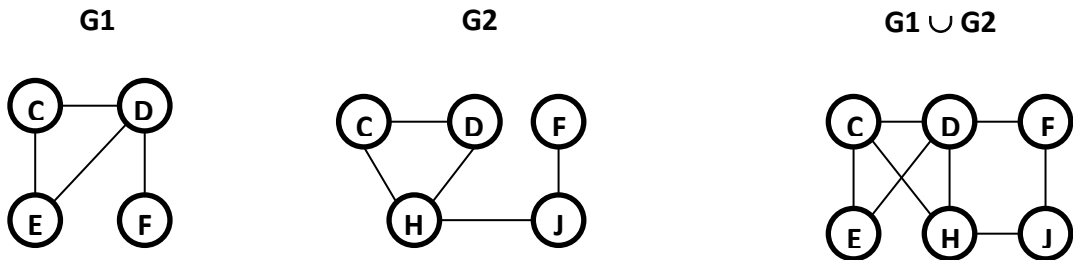


## Q6: A Consummation Devoutly To Be Wished

A simple, undirected graph  $G$  is defined by a set of vertices  $V$  and edges  $E$ . Each vertex in a graph is labeled by a unique capital letter and no more than one edge may exist between any two different vertices. Given two graphs,  $G_1$  and  $G_2$ , you are to compute the union graph  $G_3 = G_1 \cup G_2$  such that  $V_3 = V_1 \cup V_2$  and  $E_3 = E_1 \cup E_2$ . In particular, if the same letter label appears in both graphs, then the merged graph shall contain a single vertex representing the merger of the two vertices.



$C(DE)D(CEF)E(CD)F(D)$      $C(DH)D(CH)F(J)H(CD)J(FH)$      $C(DEH)D(CEFH)E(CD)F(DJ)H(CD)J(FH)$

Note how the final merged graph has six vertices (the union of the lettered vertices in  $G_1$  and  $G_2$ ) and every edge from  $G_1$  and  $G_2$  also exists in  $G_1 \cup G_2$ . The string representation of a graph has each vertex appearing in ascending order, and the adjacent vertices of each vertex are listed in parentheses also in ascending order. You can verify that the strings below each graph represent the corresponding graphs.

### Input

There will be two lines of input. The first line contains the string representation of graph  $G_1$  on a line by itself. The second line contains the string representation of graph  $G_2$  on a line by itself. Each graph will contain between 2 and 5 vertices and no more than 10 edges. The input will be properly formatted with each vertex appearing in ascending order and the adjacent vertices of each vertex listed in parentheses also appearing in ascending order. You can assume that every vertex has at least one edge connecting it to another vertex in the graph.

### Output

You shall output the string representation of the merged graph  $G_1 \cup G_2$  on a line by itself using the format as shown in the above example, where each vertex appears in ascending order, and the adjacent vertices of each vertex are listed in parentheses, also in ascending order.

### Sample Input and Output

| Input                                                                        | Output                                                 |
|------------------------------------------------------------------------------|--------------------------------------------------------|
| A (BC) B (A) C (A)<br>A (XY) X (A) Y (A)                                     | A (BCXY) B (A) C (A) X (A) Y (A)                       |
| A (B) B (A)<br>C (D) D (C)                                                   | A (B) B (A) C (D) D (C)                                |
| A (B) B (A) C (D) D (C)<br>A (C) C (A)                                       | A (BC) B (A) C (AD) D (C)                              |
| A (BCD) B (ADX) C (AX) D (AB) X (BC)<br>A (BE) B (AX) E (AFX) F (EX) X (BFE) | A (BCDE) B (ADX) C (AX) D (AB) E (AFX) F (EX) X (BCEF) |