//longest common subsequence

#include <iostream>

#include <string>

#include <vector>

using namespace std;

string s1, s2;

int n1, n2;

vector<vector<int>> value;

vector<vector<char>> arrow;

void input() {

    cout << "enter the string 1: ";

    cin >> s1;

    cout << "enter the string 2: ";

    cin >> s2;

    n1 = s1.length();

    n2 = s2.length();

}

void logic() {

    value.resize(n1+1, vector<int>(n2+1,0));

    arrow.resize(n1+1, vector<char>(n2+1,0));

    for (int i = 0; i <= n1; i++) {

        for (int j = 0; j <= n2; j++) {

            if (i == 0 || j == 0) {

                value[i][j] = 0;

                continue;

            }

            if (s1[i-1] == s2[j-1]) {

                value[i][j] = value[i-1][j-1] + 1;

                arrow[i][j] = 'd';

            } else if (value[i-1][j] > value[i][j-1]) {

                value[i][j] = value[i-1][j];

                arrow[i][j] = 'u';

            } else if (value[i-1][j] < value[i][j-1]) {

                value[i][j] = value[i][j-1];

                arrow[i][j] = 'l';

            }

        }

    }

}

void printmatrix() {

    for (int i = 0; i <= n1; i++) {

        for (int j = 0; j <= n2; j++) {

            cout << value[i][j]<<" ";

        }

        cout << endl;

    }

    cout << endl;

    for (int i = 0; i <= n1; i++) {

        for (int j = 0; j <= n2; j++) {

            if(arrow[i][j]==0)

            {

                cout<<"0"<<" ";

                continue;

            }

            cout << arrow[i][j]<<" ";

        }

        cout << endl;

    }

    cout << endl;

}

void printsol(int i, int j) {

    if (i == 0 || j == 0) {

        return;

    }

    if (arrow[i][j] == 'd') {

        printsol(i-1, j-1);

        cout << s1[i-1];

    } else if (arrow[i][j] == 'u') {

        printsol(i-1, j);

    } else {

        printsol(i, j-1);

    }

}

int main() {

    input();

    logic();

    printmatrix();

    cout << endl;

    printsol(n1, n2);

    cout << endl;

    return 0;

}

Shape, arrow

Description automatically generated