#include <iostream>

#include <string>

using namespace std;

class T9

{

**private:**

string wrongstr; // строка с ошибкой

string truestr; // строка исправленная

int\*\* a;

int n;

int m;

int min(int a, int b, int c) {

if (a <= b && a <= c)

return a;

if (b <= a && b <= c)

return b;

return c;

}

int c(int j, int i) {

// нумерация строк начинается с 0

if (wrongstr[i-1] == truestr[j-1])

return 0;

return 1;

}

void createArray()

{

// 1. заполнить первую строку

for (int i = 0; i < n; i++)

a[0][i] = i;

// 2. заполнить вторую строку

for (int j = 0; j < m; j++)

a[j][0] = j;

// 3. заполнить оставшиеся ячейки

for (int j = 1; j < m; j++)

for (int i = 1; i < n; i++)

{

int d = min(a[j - 1][i] + 1, a[j][i - 1] + 1, a[j - 1][i - 1] + c(j, i));

a[j][i] = d;

}

}

**public:**

T9(string str1, string str2) {

wrongstr = str1;

truestr = str2;

n = str1.length() + 1;

m = str2.length() + 1;

a = new int\*[m];

for (int j = 0; j < m; j++)

a[j] = new int[n];

createArray();

}

int getNum()

{

return a[m][n];

}

void printArray()

{

for (int j = 0; j < m; j++)

{

for (int i = 0; i < n; i++)

cout << a[j][i] << " ";

cout << endl;

}

}

string getCombo()

{

string result = "";

// старт - нижний правый угол

int j = m-1;

int i = n-1;

// пока не левый верхний угол

while (!(j == 0 && i == 0))

{

int d = min(a[j - 1][i - 1], a[j - 1][i], a[j][i - 1]);

// по диагонали

if (d == a[j - 1][i - 1])

{

if (a[j - 1][i - 1] == a[j][i])

result = 'M' + result;

else

result = 'R' + result;

j--;

i--;

}

else

// влево

if (d == a[j][i - 1])

{

result = 'D' + result;

i--;

}

else

// вверх

if (d == a[j - 1][i])

{

result = 'I' + result;

j--;

}

}

return result;

}

~T9() {

for (int j = 0; j < m; j++)

delete[] a[j];

delete[] a;

}

};

int main()

{

T9 example("ппивзет", "привет");

string combo = example.getCombo();

cout << combo << endl;

system("pause");

return 0;

}