# 72. Edit Distance

## SOLUTION IN JAVA

class Solution {

public int minDistance(String word1, String word2) {

final int m = word1.length();

final int n = word2.length();

// dp[i][j] := the minimum number of operations to convert word1[0..i) to

// word2[0..j)

int[][] dp = new int[m + 1][n + 1];

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

dp[i][0] = i;

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

dp[0][j] = j;

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

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

if (word1.charAt(i - 1) == word2.charAt(j - 1))

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

else

dp[i][j] = Math.min(dp[i - 1][j - 1], Math.min(dp[i - 1][j], dp[i][j - 1])) + 1;

return dp[m][n];

}

}