```
// Copyright 2022 Anson Cheang, and Andy Nguyen
#ifndef _HOME_IIFORCE_BADNAME_COMP4_PS5_EDISTANCE_H_
#define _HOME_IIFORCE_BADNAME_COMP4_PS5_EDISTANCE_H_

#include <iostream>
#include <string>
#include <vector>
#include <algorithm>

using namespace std; //NOLINT

class EDistance {
 public:
    /* accepts the two strings to be compared, and allocates any
     data structures necessary into order to do the work
     (e.g., the NÃ\227M matrix).*/
    explicit EDistance(string N, string M);

    /* returns the penalty for aligning chars a and b
     (this will be a 0 or 1)*/
    static int penalty(char a, char b);

    // returns the minimum of the three arguments
    static int min(int a, int b, int c);

    /* populates the matrix based on having the two strings, and returns
     the optimal distance (from the [0][0] cell of the matrix when done).*/
    int optDistance();

    /* traces the matrix and returns a string that can be printed to display
     the actual alignment. In general, this will be a multi-line string â\200\224 i.e.,
     with embedded \n's.*/
    string alignment();

 private:
    string n, m;
    vector<vector<int>> matrix;
};

#endif  // _HOME_IIFORCE_BADNAME_COMP4_PS5_EDISTANCE_H_
```