1)

This problem is solved with an edit distance algorithm. The operations will be insertion, deletion, substitution, and no change. These operations will be represented by I, D, S, and N respectively. If the 3 strings provided are used the output is:

"watch the movie raising arizona?" to *"watch da mets raze arizona?"* = edit distance of 12

"this is what happens when I type slow" to *"htishisth whaty havpens when ui type fasht"* = edit distance of 14

"leonard skiena" to *"lynard skynard"* = edit distance of 16

```
    editDist( str1 , str2 , str1.length(), str2.length()); //initial call

    static int min(int x,int y,int z)  //calculates min cost
    {
        if (x<y && x<z){ System.out.print('I'); return x;}
        if (y<x && y<z){ System.out.print('D'); return y;}
        else{ System.out.print('S'); return z;}
    }

    static int editDist(String str1 , String str2 , int i ,int j)
    {
    // String 1 empty, so return length of string 2
    if (i == 0) return j;

    // String 2 empty, so return length of string 1
    if (j == 0) return i;

    // If last characters of two strings are same, no changes
    // need to be made
    if (str1.charAt(i-1) == str2.charAt(j-1))
         System.out.print('N');
         return editDist(str1, str2, i-1, j-1);

    // If last characters are not same, consider all three
    // operations on last character of first string, recursively
    // compute minimum cost for all three operations and take
    // minimum of three values.
    return 1 + min ( editDist(str1, str2, i, j-1),     // Insert
                     editDist(str1, str2, i-1, j),    // Delete
                     editDist(str1, str2, i-1, j-1) //
Substitute
                   );
    }
```