//Longest Common SubSequence

//Dynamic Programming

char a[210], b[210];

int dp[210][210], len\_a, len\_b;

//LCS is the same sequence in two strings

//a s x z

//s x z a

//LCS is 3

int LCS() //longest common subsequence

{

for(register int i = 1; i <= len\_a; i++)

for(register int j = 1; j <= len\_b; j++) {

if(i == 0 || j == 0) //base case

dp[i][j] = 0;

else if(a[i-1] == b[j-1]) //if a match found

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

else

dp[i][j] = max(dp[i-1][j], dp[i][j-1]);

// dp[i][j] = max(ignoring b[j-1] (taking b[j]), ignoring a[i-1] (taking a[i]))

}

return dp[len\_a][len\_b];

}

int main() {

int Case = 1;

while(1) {

fgets(a, 150, stdin);

if(a[0] == '#')

break;

fgets(b, 150, stdin);

len\_a = strlen(a);

len\_b = strlen(b);

//fgets function also inputs the '\n' which is also a common sub sequence, so minus 1 for it

printf("Case #%d: LCS is %d\n", Case++, LCS()-1);

}

}