

```

/**

NAME: CATRINA Mirela
GRADE: 11
SCHOOL: NCSC "Gr. Moisil"
DIVISION: SR3

**/

#include <iostream>
#include <fstream>
#include <cstring>
using namespace std;

ifstream f("factor.in");

char a[210], b[210], lcs[210];
int n, m, llcs, rez;

void strupr(char c[210])
{
    int l = strlen(c);
    for(int i = 0; i < l; ++ i)
        if(c[i] >= 'a' && c[i] <= 'z')
            c[i] = c[i] - 'a' + 'A';
}

void ignore_other(char c[210])
{
    int l = strlen(c);
    for(int i = 0; i < l; ++i)
        if(!(c[i] >= 'A' && c[i] <= 'Z'))
        {
            strcpy(c+i, c+i+1);
            --l;
            --i;
        }
}

void prepare()
{
    strupr(a);
    strupr(b);
    ignore_other(a);
    ignore_other(b);
}

void copy_in(char string_to_look_for[210], char a[201], int i_start, int
l)

```

```

{
    for(int i = i_start; i < i_start + l; ++i)
        string_to_look_for[i-i_start] = a[i];
    for(int j = i_start + l; j <= 201; ++j)
        string_to_look_for[j] = 0;
}

int cmp(char laux[210], char string_to_look_for[210], int l)
{
    for(int i = 0; i<l; ++i)
    {
        if(laux[i] < string_to_look_for[i])
            return -1;
        if(laux[i] > string_to_look_for[i])
            return 1;
    }
    return 0;
}

void longest_common_substring(char a[210], char b[210])
{
    bool found = 0;
    int length_of = 0;
    char laux[210]{0}, string_to_look_for[210]{0};

    int vmin = min(strlen(a), strlen(b));
    for(int lungime = vmin; lungime > 0 && found == 0; --lungime)
    {
        for(int i = 0; i <= strlen(a) - lungime; ++i)
        {
            copy_in(string_to_look_for, a, i, lungime);

            char *p = strstr(b, string_to_look_for);
            if(p != NULL)
            {
                length_of = lungime;
                if(found == 0)
                    strcpy(laux, string_to_look_for);
                else if(cmp(laux, string_to_look_for, lungime) > 0)
                    strcpy(laux, string_to_look_for);
                found = 1;
            }
        }
    }
    for(int i = 0; i <= 200; ++i)
        lcs[i] = 0;
    strcpy(lcs, laux);
    llcs = length_of;
}

void copy_left(char to[210], char from[210], int toPos)
{
    for(int i = 0; i<= toPos; ++i)

```

```

        to[i] = from[i];
    }

void copy_right(char to[210], char from[210], int fromPos)
{
    int l = strlen(from);
    for(int i = fromPos; i < l; ++i)
        to[i-fromPos] = from[i];
}

void solve(char a[210], char b[210])
{
    longest_common_substring(a, b);
    if(llcs == 0)
        return;
    rez += llcs;

    int poz_start_a = strstr(a, lcs) - a - 1;
    int poz_start_b = strstr(b, lcs) - b - 1;

    char st_a[210]{0}, st_b[210]{0}, dr_a[210]{0}, dr_b[210]{0};

    copy_left(st_a, a, poz_start_a);
    copy_left(st_b, b, poz_start_b);

    copy_right(dr_a, a, poz_start_a + llcs + 1);
    copy_right(dr_b, b, poz_start_b + llcs + 1);

    solve(st_a, st_b);
    solve(dr_a, dr_b);
}

int main()
{
    for(int acsl = 0; acsl < 5; ++acsl)
    {
        f.getline(a, 210);
        f.getline(b, 210);
        prepare();
        solve(a, b);
        cout<<rez<<'\n';
        rez = 0;
    }
    return 0;
}

```