#include <stdio.h>

#include <string.h>

#include <string>

// funkcia premiena znak 'A' az 'Z' alebo medzeru na cislo 0..26

int ctoi(char c)

{

if (c >= 'A' && c <= 'Z') return c - 'A';

if (c >= 'a' && c <= 'z') return c - 'a';

return 26;

}

// funkcia premiena cislo 0..26 na znak 'A' az 'Z' alebo medzeru

char itoc(int i)

{

if (i >= 0 && i <= 25) return i + 'A';

return ' ';

}

// funkcia a + b mod 27

int add(int a, int b)

{

return (a + b) % 27;

}

// funkcia a \* b mod 27

int mul(int a, int b)

{

return (a \* b) % 27;

}

// funkcia a^-1 mod 27

int inv(int a)

{

int b;

for (b = 1; b < 27; b++) {

if (mul(a,b) == 1) return b;

}

return 0;

}

// vstupny retazec

const char \*input = "LIYGTOGDPOAUPDFQNVPVDAQV";

// vystupny buffer

char output[1000];

int mostFrequent(const char \*text)

{

char pom[1];

int max = 0;

int count = 0;

int velkost = strlen(text);

for(int j = 0; j < 26; j++){

count = 0;

\*pom = itoc(j);

for(int i = 0; i < velkost; i++){

if(text[i] == pom[0])

count++;

}

if(count > max){

max = count;

}

}

return max;

}

int main(int argc, char \*argv[])

{

// sifrovacie kluce k1 a k2

int k1 = 1, k2 = 1;

int k1c, k2c;

//int medzery = mostFrequent(input);

// Predpokladany maximalny pocet medzier

//int maxMedzier = 3;

int maxMedzier = mostFrequent(input);

// ciselny kod priameho a zasifrovaneho pismena

int p, c;

// ukazovatele pozicie priameho a zasifrovaneho buffera

const char \*src = input;

char \*dest = output;

printf("\nZasifrovany text: %s", input);

printf("\n\n");

for(int k1 = 1; k1 < 27; k1++){

for(int k2 = 1; k2 < 27; k2++){

// ukazovatele pozicie priameho a zasifrovaneho buffera

const char \*src = input;

char \*dest = output;

// vypocet desifrovacich klucov

k1c = inv(k1);

k2c = mul(k1c, 27 - k2);

// desifrovanie

while (\*src != '\0') {

c = ctoi(\*src);

p = add(mul(k1c,c), k2c); // p = k1' \* c + k2' mod 27

\*dest = itoc(p);

src++;

dest++;

}

\*dest = '\0';

int pocetMedzier = 0;

for(int i = 0; i < strlen(input); i++)

{

if(output[i] == ' '){

pocetMedzier++;

}

}

if(pocetMedzier == maxMedzier){

if(k1 == 17 && k2 == 5){

printf("\Desifrovany text : %s, k1:%d, k2:%d, k1c:%d, k2c:%d\n", output, k1, k2 ,k1c, k2c);

}

//printf("\Desifrovany text : %s, k1:%d, k2:%d, k1c:%d, k2c:%d\n", output, k1, k2 ,k1c, k2c);

}

}

}

return 0;

}