#include <iostream>  
using namespace std;  
char str[100], ptn[20];  
int res, m, n, len, len1, i, j, k, table[1000];  
  
void shift(const char p[]) {  
    len = std::strlen(p);  
    for (i = 0; i < 1000; i++)  
        table[i] = len;  
    for (j = 0; j <= len - 2; j++)  
        table[p[j]] = len - 1 - j;  
}  
  
int horspool(const char p[], const char t[]) {  
    shift(p);  
    m = std::strlen(p);  
    n = std::strlen(t);  
    i = m - 1;  
    while (i <= n - 1) {  
        k = 0;  
        while (k <= m - 1 && (p[m - 1 - k] == t[i - k]))  
            k++;  
        if (k == m)  
            return i - m + 1;  
        else  
            i = i + table[t[i]];  
    }  
    return -1;  
}  
  
int main() {  
    struct timeval tv1, tv2;  
    struct rusage r\_usage;  
  
    std::cout << "Enter the text: " << std::endl;  
    std::cin.getline(str, sizeof(str));  
  
    std::cout << "Enter the pattern to be found: " << std::endl;  
    std::cin.getline(ptn, sizeof(ptn));  
  
    gettimeofday(&tv1, nullptr);  
    res = horspool(ptn, str);  
    gettimeofday(&tv2, nullptr);  
  
    if (res == -1)  
        std::cout << "\nPattern not found" << std::endl;  
    else  
        std::cout << "Pattern found at " << res + 1 << " position" << std::endl;  
  
    std::cout << "Time of Horsepool's Algorithm = "  
              << static\_cast<double>(tv2.tv\_usec - tv1.tv\_usec) << " microseconds" << std::endl;  
  
    getrusage(RUSAGE\_SELF, &r\_usage);  
    std::cout << "Memory usage: " << r\_usage.ru\_maxrss << " kilobytes" << std::endl;  
  
    return 0;  
}