

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>

const int MAX_LEN = 1000000;

int shift_and(char *text, char *pattern) {
    int len = strlen(pattern);
    int *p = (int *)calloc(sizeof(int), len); //定义数组p[]模拟公式中p的二进制表示
    int **d; //定义二维数组d[][]模拟公式中d[]数组的值

    d = (int **)calloc(sizeof(int*), 127);
    for (int i = 0; i < 127; ++i) d[i] = (int *)calloc(sizeof(int), len);
    for (int i = 0; pattern[i]; ++i) d[pattern[i]][i] = 1;

    int flag = 0;
    for (int i = 0; text[i]; ++i) { //遍历母串
        for (int j = len - 1; j >= 0; --j) p[j + 1] = p[j] && d[text[i]][j + 1]; //p[]数组整体向右移动，并与对应母串当前字母d[][]二维数组进行&&操作
        p[0] = 1 && d[text[i]][0]; //判断新进来的的节点是否匹配 让p[0]与与对应母串当前字母d[][]二维数组进行&&操作
        if (p[len - 1] == 1) { //模拟退出条件 (p[]数组最后一位等于1)
            flag = 1;
            break;
        }
    }

    return flag;
}

int main() {
    char text[MAX_LEN], pattern[MAX_LEN];
    scanf("%s%s", text, pattern);
    printf("%s = %d\n", pattern, shift_and(text, pattern));

    return 0;
}

```