KMP

给定一个模式串S,以及一个模板串P,所有字符串中只包含大小写英文字母以及阿拉伯数字。

模板串P在模式串S中多次作为子串出现。

求出模板串P在模式串S中所有出现的位置的起始下标。

输入格式

第一行输入整数N,表示字符串P的长度。

第二行输入字符串P。

第三行输入整数M,表示字符串S的长度。

第四行输入字符串S。

输出格式

共一行,输出所有出现位置的起始下标(下标从0开始计数),整数之间用空格隔开。

数据范围

 $1 \le N \le 10^5$

 $1 \le M \le 10^6$

输入样例:

3

aba

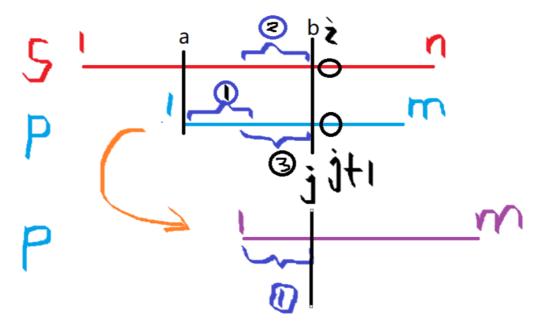
ababa

输出样例:

0 2

KMP主要分两步:求next数组、匹配字符串。个人觉得匹配操作容易懂一些,疑惑我一整天的是求next数组的思想。所以先把匹配字符串讲一下。

s串 和 p串都是从1开始的。i 从1开始, j 从0开始, 每次s[i] 和p[j+1]比较



当匹配过程到上图所示时,

当匹配过程到上图所示时,

s[a,b]=p[1,j] & & s[i]!=p[j+1]此时要移动p串 (不是移动1格, 而是直接移动到下次能匹配的位置)

其中1串为[1, next[j]], 3串为[j-next[j]+1,j]。由匹配可知 **1串等于3串**, **3串等于2串**。所以直接移动p串使1到3的位置即可。这个操作可由**j=next[j]**直接完成。如此往复下去,当 j== m时匹配成功。

KMP算法动画演示https://www.bilibili.com/video/BV1Qb411h7U6/

y总代码:

```
1 #include <iostream>
 3 using namespace std;
  5 const int N = 10010, M = 100010;
 8 char p[N], s[M];
9 int ne[N];
10
11 int main()
12 * {
13
         cin >> n >> p + 1 >> m >> s + 1;
14
15
         // 求next的过程
         for (int i = 2, j = 0; i <= n; i ++ )
16
17 -
             while (j && p[i] != p[j + 1]) j = ne[j];
if (p[i] == p[j + 1]) j ++;
18
19
             ne[i] = j;
20
21
 22
23
         // kmp 匹配过程
         for (int i = 1, j = 0; i \leftarrow m; i \leftrightarrow ++)
24
25 *
              while (j && s[i] != p[j + 1]) j = ne[j];
26
              if (s[i] = p[j + 1]) j ++;
if (j = n)
27
28
29 -
                  printf("%d ", i - n);
30
31
                  j = ne[j];
32
33
         }
34
35
         return 0;
36 }
```

```
using namespace std;
const int N = 100010, M = 1000010;
int n, m;
int ne[N];
char s[M], p[N];
int main()
{
   cin >> n >> p + 1 >> m >> s + 1;
   for (int i = 2, j = 0; i <= n; i ++ )
       while (j \&\& p[i] != p[j + 1]) j = ne[j];
       if (p[i] == p[j + 1]) j ++ ;
       ne[i] = j;
   }
   for (int i = 1, j = 0; i <= m; i ++ )
       while (j && s[i] != p[j + 1]) j = ne[j];
       if (s[i] == p[j + 1]) j ++ ;
       if (j == n)
           printf("%d ", i - n);
           j = ne[j];
       }
   }
   return 0;
}
作者: yxc
链接: https://www.acwing.com/activity/content/code/content/43108/
来源: AcWing
著作权归作者所有。商业转载请联系作者获得授权,非商业转载请注明出处。
```