

# Number Sequence

## 一. 考察内容:

字符串 字符串哈希

## 二. 题目分析:

[题目大意]

给出一个字符串b, 找出该字符串再另一个字符串a中第一次出现的位置。

[写题思路]

这是一道字符串哈希的模板题, 首先预处理出a中每个长度为len(b)的子串哈希值, 但是直接预处理这些哈希值不能达到线性求解, 于是我们考虑维护其他信息。

我们维护出字符串a的每个后缀子串的哈希值, 然后对于任意一个子串, 我们只需要用前缀和的思想将子串两个端点的哈希值相减即可。

然后, 从字符串a第一个字符开始向后扫一遍, 判断每个长度为len(b)的字符是否与b的哈希值相同, 如果相同就输出。

## 三. 代码实现:

```
#define _CRT_SECURE_NO_DEPRECATE
/*****
*创建时间: 2018 09 18
*文件类型: 源代码文件
*题目来源: HDU
*当前状态: 已通过
*备忘录: 哈希
*作者: HtBest
*****/
#include <stdio.h>
#include <iostream>
#include <string>
#include <stdlib.h>
#include <string.h>
#include <algorithm>
#include <queue>
#include <bitset>
// #include <sys/wait.h>
// #include <sys/types.h>
// #include <unistd.h>
using namespace std;
#define MAXN 1000002
int n,m;
unsigned long long h[MAXN],H,Pow[10001];
/* Variable explain:

*/
void read()
{
    short a[MAXN];
    scanf("%d%d",&n,&m);
    for(int i=0;i<n;++i)scanf("%hd",&a[i]);
    for(int i=n-1;i>=0;--i)h[i]=h[i+1]*2333333333333333+a[i];
    for(int i=0;i<m;++i)scanf("%hd",&a[i]);
    H=0;
    for(int i=m-1;i>=0;--i)H=H*2333333333333333+a[i];
    return;
}
int main()
{
    // freopen(".in","r",stdin);
    // freopen(".out","w",stdout);
```

```
Pow[0]=1;
for(int i=1;i<=10000;++i)Pow[i]=Pow[i-1]*2333333333333333;
int T;
scanf("%d",&T);
while(T--){
    read();
    for(int i=0;i<=n-m+1;++i){
        // printf("%llu %llu\n",h[i]-h[i+m]*Pow[m],H);
        if(h[i]-h[i+m]*Pow[m]==H){printf("%d\n",i+1);break;}
        if(i==n-m+1)printf("%d\n",-1);
    }
}
return 0;
}
```

[<题目跳转>](#) [<查看代码>](#)