

H—— 解题报告

题意:给出一个长为 n 的,由字母表中前 k 个小写字母组成的字符串 s .求一个长度为 m 的字符串,要求该串也只能由字母表前 k 个小写字母组成,且其不能是 s 的子串.

思路:Hash 即可.由于 s 只有 k 种字母,于是可以把 s 表示成一个 n 位 k 进制的数,那么只要将 s 中所有长度为 m 的子串 Hash 统计,再用一个没有出现过的 Hash 值构造一个长为 m 的串即可.

为了方便对字符串 Hash 以及由 Hash 值构造字符串,可以将 k 进制数的 10 进制表示作为 Hash 值,这就解决了由 Hash 值构造字符串的问题.

```
#include <stdio>
#include <string>
#include <algorithm>

using namespace std;

typedef unsigned long long ull;

const int N = 10010;

char s[N];
int hn,n,m,k;
ull h[N];

ull power(ull a,int b)
{
    ull res=1;
    while(b)
    {
        if(b & 1)    res*=a;
        a*=a;
        b>>=1;
    }
    return res;
}

bool output(int num,int len)
{
    if(len==0)    return 1;
    char ch=num%k+'a';
    if(num%k==k-1) return 0;
    num/=k;
    if(output(num,len-1)==0)    return 0;
    putchar(ch);
    return 1;
}
```

```

int main(void)
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        scanf("%d%d%d",&n,&m,&k);
        scanf("%s",s);
        for(int i=0;i<n;i++)
            s[i]='a';
        k++;
        hn=0;
        ull hash=0;
        ull kn=power(k,m-1);
        for(int i=0;i<m-1;i++)
            hash=hash*k+s[i];
        for(int i=m-1;i<n;i++)
        {
            hash=hash*k+s[i];
            h[hn++]=hash;
            hash-=s[i-m+1]*kn;
        }
        sort(h,h+hn);
        hn=unique(h,h+hn)-h;
        int i=0,j=0;
        while(1)
        {
            for(;j<hn && i==h[j];i++,j++);
            if(output(i,m)) break;
            i++;
        }
        puts("");
    }
    return 0;
}

```