求和(sum.cpp/c/pas)

【问题描述】

一条狭长的纸带被均匀划分出了 n 个格子，格子编号从 1 到 n。每个格子上都染了一种颜色 $color_i$（用[1，m]当中的一个整数表示），并且写了一个数字 $number_i$。

定义一种特殊的三元组：(x, y, z)，其中 x，y，z 都代表纸带上格子的编号，这里的三元组要求满足以下两个条件：

1. x, y, z 都是整数,$x < y < z$, $y - x = z - y$

2. $color_x=color_z$

满足上述条件的三元组的分数规定为$(x+z)*(number_x+number_z)$。整个纸带的分数规定为所有满足条件的三元组的分数的和。这个分数可能会很大，你只要输出整个纸带的分数除以 10,007 所得的余数即可。

【输入格式】

输入文件名为 sum.in。

第一行是用一个空格隔开的两个正整数 n 和 m，n 代表纸带上格子的个数，m 代表纸带上颜色的种类数。

第二行有 n 个用空格隔开的正整数，第 i 个数字 $number_i$ 代表纸带上编号为 i 的格子上面写的数字。

第三行有 m 个用空格隔开的正整数，第 i 个数字 $color_i$ 代表纸带上编号为 i 的格子染的颜色。

【输出格式】

输出文件名为 sum.out。

共一行，一个整数，表示所求的纸带分数除以 10,007 所得的余数。

【数据说明】

对于第 1 组至第 2 组数据，$1 \le n \le 100, 1 \le m \le 5$；

对于第 3 组至第 4 组数据，$1 \le n \le 3000, 1 \le m \le 100$；

对于第 5 组至第 6 组数据，$1 \le n \le 100000, 1 \le m \le 100000$，且不存在出现次数超过 20 的颜色；

对于全部 10 组数据，$1 \le n \le 100000$, $1 \le m \le 100000$, $1 \le color_i \le m$, $1 \le number_i \le 100000$。

【思路】

先分析一下，我们的任务是什么。题目的要求是求分数和，我们就得把所有符合条件的三元组“找”出来。

至少需要枚举三元组(x,y,z)中的一个元素，这里枚举的是 z（当然 x 也可以，不过不要选 y，因为 y 对分数没什么用）。

1、因为 $x<y<z$，所以只需向前枚举 x,y

2、因为 $y-x=z-y$，所以 $x+z=2y$，x、z 同奇偶，且分数与 y 无关，只需枚举 z 和 x。

3、因为 $colour_x=colour_z$，所以只需枚举 z 之前同奇偶且同色的 x。

这样的话时间复杂度是 $O(n^2)$，能得 40 分。如何快速枚举 x 呢？

其实不是快速枚举 x，是快速枚举分数和。

观察三元组分数：

$(x+z) \cdot (number_x+number_z)$

显然我们不方便处理多项式乘法，那就把它拆开

（事实上很多人到这步都放弃了，其实试一试立刻就明白了）

$=x\cdot number_x+x\cdot number_z+z\cdot number_x+z\cdot number_z$

那么对于 z 的所有合法决策 x1,x2,……,xk

根据乘法分配率，分数=$\Sigma(xi*number_{xi})+\Sigma(xi)*number_z+\Sigma(number_{xi})*z+\Sigma(z*number_z)$ (1<=i<=k)

由于 z 是枚举的，所以只需快速得到$\Sigma(x\cdot number_x)$，Σx，$\Sigma number_x$ 和 k（注意最后一项被算了 k 次，要乘 k）

这样我们就可以开 4 个累加器，分别记录这四个量。而对于不同奇偶性、不同颜色的 z 有不同的决策 x，所以要开一个 s[2][m][4]的累加器。

【时空复杂度】

O(n)，O(n+m)

【注意】题目数据较大，每次计算一定要模 10007，否则很容易出错。

【样例程序】

```
#include <cstdio>

const int maxn=100000;
const int maxm=100000;
const int p=10007;

int n,m,ans;
int number[maxn+1],colour[maxn+1];
int s[2][maxm+1][4];

void init()
{
	freopen("sum.in","r",stdin);
	freopen("sum.out","w",stdout);

	scanf("%d%d",&n,&m);
```

```
	for(int i=1;i<=n;i++)
		scanf("%d",&number[i]);
	for(int i=1;i<=n;i++)
		scanf("%d",&colour[i]);
}

void solve()
{
	for(int i=1;i<=n;i++)
	{
		int z=i%p,numz=number[i]%p,c=colour[i],t=i%2;
		int count=s[t][c][0]%=p,x=s[t][c][1]%=p,
		numx=s[t][c][2]%=p,x_numx=s[t][c][3]%=p;
		ans=(ans+((count*z)%p*numz)%p)%p;
		ans=(ans+x_numx)%p;
		ans=(ans+x*numz)%p;
		ans=(ans+z*numx)%p;
		s[t][c][0]++;
		s[t][c][1]+=z;
		s[t][c][2]+=numz;
		s[t][c][3]+=z*numz;
	}
}

void output()
{
	printf("%d\n",ans);

	fclose(stdin);
	fclose(stdout);
}

int main()
{
	init();
	solve();
	output();
	return 0;
}
```