**算法训练 貌似化学**

时间限制：1.0s   内存限制：256.0MB

问题描述

　　现在有a,b,c三种原料，如果他们按x:y:z混合，就能产生一种神奇的物品d。  
　　当然不一定只产生一份d，但a,b,c的最简比一定是x:y:z  
　　现在给你3种可供选择的物品:  
　　每个物品都是由a,b,c以一定比例组合成的，求出最少的物品数，使得他们能凑出整数个d物品（这里的最少是指三者个数的总和最少）

输入格式

　　第一行三个整数，表示d的配比（x,y,z）  
　　接下来三行，表示三种物品的配比，每行三个整数（<=10000）。

输出格式

　　四个整数，分别表示在最少物品总数的前提下a,b,c,d的个数（d是由a,b,c配得的）  
　　目标答案<=10000  
　　如果不存在满足条件的方案，输出NONE

样例输入

3 4 5  
1 2 3  
3 7 1  
2 1 2

样例输出

8 1 5 7

本题的C++参考代码如下：

#include <iostream>

#include <iomanip>

#include <queue>

#include <set>

#include <algorithm>

#include <cstdio>

#include <cstring>

#include <cmath>

using namespace std;

void s(long long a[][4],int x,int y)

{

long long t=a[x][y];

a[x][y]=a[y][x];

a[y][x]=t;

}

long long gcd(long long a,long long b)

{

long long c;

while(b)

{

c=a%b;

a=b;

b=c;

}

return a;

}

int main()

{

long long x,y,z;

long long a[4][4];

cin>>x>>y>>z;

for(int i=1;i<4;i++)for(int j=1;j<4;j++)cin>>a[i][j];

long long det=a[1][1]\*(a[2][2]\*a[3][3]-a[3][2]\*a[2][3])-a[1][2]\*(a[2][1]\*a[3][3]-a[3][1]\*a[2][3])+a[1][3]\*(a[2][1]\*a[3][2]-a[3][1]\*a[2][2]);

long long b[4][4];

b[1][1]=a[2][2]\*a[3][3]-a[3][2]\*a[2][3];

b[1][2]=-a[2][1]\*a[3][3]+a[3][1]\*a[2][3];

b[1][3]=a[2][1]\*a[3][2]-a[3][1]\*a[2][2];

b[2][1]=-a[1][2]\*a[3][3]+a[3][2]\*a[1][3];

b[2][2]=a[1][1]\*a[3][3]-a[3][1]\*a[1][3];

b[2][3]=-a[1][1]\*a[3][2]+a[3][1]\*a[1][2];

b[3][1]=a[1][2]\*a[2][3]-a[2][2]\*a[1][3];

b[3][2]=-a[1][1]\*a[2][3]+a[2][1]\*a[1][3];

b[3][3]=a[1][1]\*a[2][2]-a[2][1]\*a[1][2];

s(b,1,2);

s(b,1,3);

s(b,2,3);

long long n1=x\*b[1][1]+y\*b[2][1]+z\*b[3][1];

long long n2=x\*b[1][2]+y\*b[2][2]+z\*b[3][2];

long long n3=x\*b[1][3]+y\*b[2][3]+z\*b[3][3];

if(det==0)

{

cout<<"NONE"<<endl;

return 0;

}

else if(det<0)

{

n1=-n1;

n2=-n2;

n3=-n3;

det=-det;

}

if(n1<0||n2<0||n3<0)

{

cout<<"NONE"<<endl;

return 0;

}

long long gg=gcd(n1,gcd(n2,gcd(n3,det)));

cout<<n1/gg<<' '<<n2/gg<<' '<<n3/gg<<' '<<det/gg<<endl;

return 0;

}

本题的C参考代码如下：

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#define INF 214748326

double a[10][10];

double b[10][10];

double c[10];

int min=INF;

int ansx,ansy,ansz,ansk;

int fun(double x){

if ((int)(x+0.5)>(int)x) return (int)x+1;

return (int)x;

}

int main(){

int i,j,k,k1,x,y,z;

double t,tot;

for(i=1;i<4;i++)

scanf("%lf",&a[i][4]);

for(i=1;i<4;i++)

for(j=1;j<4;j++)

scanf("%lf",&a[j][i]);

memcpy(b,a,sizeof(a));

for (k=1;k<=10000;k++){

for (i=1;i<=3;i++)

a[i][4]\*=k;

for (i=1;i<3;i++){

for (j=i+1;j<=3;j++){

t=1;

if (a[j][i]!=0) t=a[i][i]/a[j][i];

a[j][i]=0;

for (k1=i+1;k1<=4;k1++){

a[j][k1]=t\*a[j][k1]-a[i][k1];

}

}

}

memset(c,0,sizeof(c));

c[3]=a[3][4]/a[3][3];

for (i=2;i>0;i--){

tot=0;

for (j=i+1;j<=3;j++){

tot+=a[i][j]\*c[j];

}

c[i]=(a[i][4]-tot)/a[i][i];

}

x=fun(c[1]);

y=fun(c[2]);

z=fun(c[3]);

if (b[1][1]\*x+b[1][2]\*y+b[1][3]\*z==b[1][4]\*k

&& b[2][1]\*x+b[2][2]\*y+b[2][3]\*z==b[2][4]\*k

&& b[3][1]\*x+b[3][2]\*y+b[3][3]\*z==b[3][4]\*k){

if (min>x+y+z){

ansx=x;

ansy=y;

ansz=z;

ansk=k;

min=z+y+z;

}

}

memcpy(a,b,sizeof(a));

}

if (ansx==0 && ansy==0 && ansz==0)

printf("NONE");

else

printf("%d %d %d %d\n",ansx,ansy,ansz,ansk);

system("pause");

return 0;

}

本题的Java参考代码如下：

import java.util.Scanner;

public class Main {

static long[][] g = new long[10][10];

public static void main(String[] args) {

long a, b, c;

long A, B, C, D, E, F, x, y, z;

Scanner scanner = new Scanner(System.in);

String[] split = scanner.nextLine().trim().split(" ");

a = Integer.parseInt(split[0]);

b = Integer.parseInt(split[1]);

c = Integer.parseInt(split[2]);

for (int l = 1; l <= 3; l++) {

for (int m = 0; m < 3; m++) {

g[l][m+1] = scanner.nextInt();

}

}

A=a\*g[1][2]-b\*g[1][1];

B=a\*g[2][2]-b\*g[2][1];

C=a\*g[3][2]-b\*g[3][1];

D=b\*g[1][3]-c\*g[1][2];

E=b\*g[2][3]-c\*g[2][2];

F=b\*g[3][3]-c\*g[3][2];

z=B\*D-A\*E;

y=A\*F-D\*C;

x=C\*E-B\*F;

if(z<0){

z=-z;y=-y;x=-x;

}

if(z<0||y<0||z<0){

System.out.println("NONE");

return;

}

long temp=gcd(x, y);

temp=gcd(temp, z);

x=x/temp;y=y/temp;z=z/temp;

System.out.println(x+" "+y+" "+z+" "+(x\*g[1][1]+y\*g[2][1]+z\*g[3][1])/a);

}

static long gcd(long a, long b){

if(a==0)return b;

if(b==0)return a;

if(a%b==0){

return b;

}

return gcd(b, a%b);

}

}