**矩阵相乘**

问题描述

　　小明最近在为线性代数而头疼，线性代数确实很抽象（也很无聊），可惜他的老师正在讲这矩阵乘法这一段内容。  
　　当然，小明上课打瞌睡也没问题，但线性代数的习题可是很可怕的。  
　　小明希望你来帮他完成这个任务。  
  
　　现在给你一个ai行aj列的矩阵和一个bi行bj列的矩阵，  
　　要你求出他们相乘的积（当然也是矩阵）。  
　　(输入数据保证aj=bi,不需要判断)

输入格式

　　输入文件共有ai+bi+2行，并且输入的所有数为整数（long long范围内）。  
　　第1行：ai 和 aj  
　　第2~ai+2行：矩阵a的所有元素  
　　第ai+3行：bi 和 bj  
　　第ai+3~ai+bi+3行：矩阵b的所有元素

输出格式

　　输出矩阵a和矩阵b的积（矩阵c)  
　　（ai行bj列）

样例输入

2 2  
12 23  
45 56  
2 2  
78 89  
45 56

样例输出

1971 2356  
6030 7141

C++代码：

# include <stdio.h>

const int Max = 100;

long long a[Max][Max];

long long b[Max][Max];

int main()

{

int a1, a2, b1, b2, i, j, k, temp;

scanf("%d %d", &a1, &a2);

for(i = 0; i < a1; i++){

for(j = 0; j < a2; j++){

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

}

}

scanf("%d %d", &b1, &b2);

for(i = 0; i < b1; i++){

for(j = 0; j < b2; j++){

scanf("%ld", &b[i][j]);

}

}

for(i = 0; i < a1; i++){

for(j = 0; j < b2; j++){

temp = 0;

for(k = 0; k < a2; k++){

temp += a[i][k] \* b[k][j];

}

printf("%d ", temp);

}

printf("\n");

}

return 0;

}

C代码：

#include <stdio.h>

long long a[300][300];

long long b[300][300];

long long c[300][300];

int main()

{

int i,j,k,bn,bm,an,am;

scanf("%d%d",&an,&am);

for(i=0;i<an;i++)

for(j=0;j<am;j++)

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

scanf("%d%d",&bn,&bm);

for(i=0;i<bn;i++)

for(j=0;j<bm;j++)

scanf("%ld",&b[i][j]);

for(i=0;i<an;i++)

for(j=0;j<bm;j++)

{

c[i][j]=0;

for(k=0;k<bn;k++)

c[i][j]+=a[i][k]\*b[k][j];

}

for(i=0;i<an;i++)

{

for(j=0;j<bm;j++)

printf("%ld ",c[i][j]);

printf("\n");

}

return 0;

}

JAVA代码：

import java.util.\*;

public class Main{

//矩阵相乘

public static long[][] Mul(long[][] a, long[][] b) {

//确保矩阵a的列数和b的行数相等

if(a[0].length != b.length) {

return null;

}

//用来存放结果的矩阵，axb的结果为a的行数和b的列数

long[][] result = new long[a.length][b[0].length];

//对a的每行进行遍历

for(int i=0; i<a.length; i++) {

//对b的每列进行遍历

for(int j=0;j<b[0].length; j++) {

//c为每一个点的值

long c = 0;

//第i行j列的值为a的第i行上的n个数和b的第j列上的n个数对应相乘之和，其中n为a的列数，也是b的行数，a的列数和b的行数相等

for(int k=0; k<a[0].length; k++) {

c += (a[i][k]\*b[k][j]);

}

result[i][j] = c;

}

}

return result;

}

public static void main(String[]args){

long a[][];

long b[][];

String a\_str[];

Scanner scan=new Scanner(System.in);

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

int ai=Integer.parseInt(str[1]);

int aj=Integer.parseInt(str[0]);

a=new long [aj][ai];

for(int i=0;i<aj;i++){

a\_str=scan.nextLine().trim().split(" ");

int ia=0;

for(String s:a\_str){

a[i][ia]=Integer.parseInt(s);

ia++;

}

}

str=scan.nextLine().trim().split(" ");

int bi=Integer.parseInt(str[1]);

int bj=Integer.parseInt(str[0]);

b=new long [bj][bi];

for(int i=0;i<bj;i++){

a\_str=scan.nextLine().trim().split(" ");

int ia=0;

for(String s:a\_str){

b[i][ia]=Integer.parseInt(s);

ia++;

}

}

long c[][]=Mul(a,b);

for(long tmp[]:c){

for(long i:tmp)

System.out.print(i+" ");

System.out.println();

}

}

}