

矩阵乘法

杨琦 西安交通大学计算机教学实验中心



【例】矩阵乘法

用户输入 $A_{M\times N}$, $B_{N\times K}$ 两个矩阵的元素,计算它们的乘积并输出。其中M,N,K也由用户输入,它们均不超过20。



【问题分析】输入矩阵的元素只要根据行数、列数,使用循环即可。设C是矩阵的乘积,则矩阵乘法的运算公式是

$$c_{ij} = \sum_{k=1}^{N} a_{ik} \times b_{kj}, \qquad i = 1,...,M, j = 1,...,K$$



【算法描述】



$$c_{ij} = \sum_{k=1}^{N} a_{ik} \times b_{kj}, \qquad i = 1,...,M, j = 1,...,K$$



【源程序1】

```
#include <iostream>
using namespace std;
int main(){
    const int M=20,N=20,K=20;
    double A[M][N],B[N][K],C[M][K];
    int M1,N1,N2,K1; //矩阵的实际维数
    int i,j,k;
                         //循环变量
    cout<<"请输入第1个矩阵的维数M N"<<endl;
                         //输入第1个矩阵的行数和列数
    cin>>M1>>N1:
    cout<<"请按行输入第1个矩阵的元素"<<endl:
```





【源程序2】

```
for(i=0;i<M1;i++) { //输入第1个矩阵的元素,按行
   for(j=0;j<N1;j++) //第i行
         cin>>A[i][j];
 cout<<"请输入第2个矩阵的维数N K"<<endl;
                      //输入第2个矩阵的行数和列数
 cin>>N2>>K1;
 while(N2!=N1) {
cout<<"第2个矩阵的行数应等于第1个矩阵的列数,请重输"<<endl;
   cin>>N2>>K1:
```

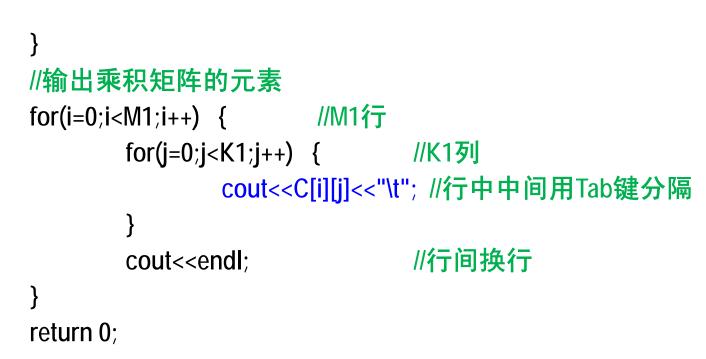


【源程序3】

```
cout<<"请按行输入第2个矩阵的元素"<<endl;
  for(i=0;i<N1;i++) { //输入第2个矩阵的元素, 按行
   for(j=0;j<K1;j++)
                      //第i行
     cin>>B[i][j];
                      //第M1个行
  for(i=0;i<M1;i++) {
   for(j=0;j<K1;j++) { //每行的K1列
     C[i][j]=0; //i行j列元素赋初值0
     for(k=0;k<N1;k++) //计算i行j列元素的值
         C[i][j]=C[i][j]+A[i][k]*B[k][j];
```



【源程序4】







【运行结果】

请输入第1个矩阵的维数M N 23

请按行输入第1个矩阵的元 素

2 3 1

7 5 3

请输入第2个矩阵的维数NK

3 2

请按行输入第2个矩 阵的元素

<u>23</u>

1 2

4 5

46 31



【问题扩展】

①这是一个正确的程序,但不是一个好的程序。请同学们讨论这个程序有哪些不足?如何处理?



②关于矩阵的典型运算还有两个矩阵的和、差,求矩阵的逆矩阵、转置矩阵,矩阵元素的最大值、最小值,各行的最大、最小值,对角线的乘积、和等等,总之是通过下标操作数组的元素。



【总结】

①二维数组的定义
<类型><数组名>[<行数>][<列数>];

- ②二维数组的使用 数组名[i][j], i,j称为下标,从0开始,表示第i+1行j+1列的 元素。下标不能越界。

③二维数组的存储 按行存储