

《计算机编程》课程上机作业范例程序

张 勇

上海财经大学信息管理与工程学院

第四次上机作业范例程序

1、打印数字金字塔

```
#include <iostream>
using namespace std;
int main(){
    int number;
    while(true){
        cout<<"请输入金字塔的层数(<10):";
        cin>>number;
        if(number>0 && number<10){
            for(int i=number-1;i>=0;i--){
                for(int j=0;j<number-i-1;j++) cout<<' ';
                for(int j=0;j<i+1;j++) cout<<j+1;
                for(int j=i;j>0;j--) cout<<j;
                cout<<endl;
            }
            for(int i=1;i<number;i++){
                for(int j=0;j<number-i-1;j++) cout<<' ';
                for(int j=0;j<i+1;j++) cout<<j+1;
                for(int j=i;j>0;j--) cout<<j;
                cout<<endl;
            }
        }
        else
            return 0;
    }
}
```

2、打印杨辉三角形

```
#include <iostream>
#include <iomanip>
using namespace std;

const int N = 7;    // 行数
const int Gap = 3;  // 间隔的空格数
int main(){
```

```
int i,j,k;
int a[N][N];    // 定义表示杨辉三角形的二维数组

for(i = 0; i<N; i++)
{
    a[i][0] = 1; a[i][i] = 1;  // 每行的第一个和最后一个元素值为 1
}

for(i = 2; i<N; i++)    // 从第二行开始，中间的元素值
    for(j = 1; j<i; j++) // 等于前一行两个元素值之和
        a[i][j] = a[i-1][j] + a[i-1][j-1];

for(i = 0; i<N; i++)    // 输出所有元素
{
    for(k = 0; k <= N-i; k++)
        cout<<setw(Gap)<<" ";
    for(j = 0; j <= i; j++)
        cout<<setw(2*Gap)<<a[i][j];
    cout<<endl;
}
return 0;
}
```

3、字符串排序（有两个参考答案）

// 利用字符数组实现

```
#include <iostream>
#include <string>
using namespace std;
const int  NUM = 5;

int main( )
{
    char str[NUM][50];
    char tempstr[50];
    cout<<"Please input original strings:"<<endl
        <<"-----"<<endl;
    for(int i = 0; i<NUM; i++)
        cin>>str[i];
    for(int i = 0; i<NUM-1; i++) {
        int k = i;
        for(int j = i+1; j<NUM; j++)
            if(strcmp(str[k],str[j])>0)
                k = j;
        if(k!=i){
            strcpy(tempstr,str[i]);
```

```

        strcpy(str[i],str[k]);
        strcpy(str[k],tempstr);
    }
}
cout<<endl<<"Sorted strings:"<<endl
    <<"-----"<<endl;
for(int i = 0; i<NUM; i++)
    cout<<str[i]<<endl;
return 0;
}

```

// 利用字符串类实现

```

#include <iostream>
#include <string>
using namespace std;
const int NUM = 5;
int main( ){
    string str[NUM];
    string tempstr;
    cout<<"Please input original strings:"<<endl
        <<"-----"<<endl;
    for(int i = 0; i<NUM; i++)
        cin>>str[i];
    for(int i = 0; i<NUM-1; i++){
        int k = i;
        for(int j = i+1; j<NUM; j++)
            if(str[k]>str[j])
                k = j;
        if(k!=i)
        {
            tempstr = str[i];
            str[i] = str[k];
            str[k] = tempstr;
        }
    }
    cout<<endl<<"Sorted strings:"<<endl
        <<"-----"<<endl;
    for(int i = 0; i<NUM; i++)
        cout<<str[i]<<endl;

    return 0;
}

```

4. 寻找整数矩阵最大值

```
#include <iostream>
using namespace std;
int main( ){
    int a[3][4]={ {56,12,23,56},{37,2,56,46},{-12,-34,56,8}};
    int maxvalue = a[0][0] -1;
    int maxvaluenum = 0;
    int position[12][2] = {0};

    for (int i = 0; i<=2; i++)
        for (int j = 0; j<=3; j++)
            if (a[i][j]>=maxvalue) {
                maxvaluenum = a[i][j] == maxvalue? ++maxvaluenum:1;
                maxvalue = a[i][j];
                position[maxvaluenum-1][0] = i;
                position[maxvaluenum-1][1] = j;
            }

    for(int i = 0;i<maxvaluenum; i++)
        cout<<"max="<<maxvalue<<".row="<<position[i][0]
        <<".column="<<position[i][1]<<endl;
    return 0;
}
```

5. 报数离队问题（有两个参考答案）

```
// 二层循环的实现
#include <iostream>
using namespace std;
const int PersonNum = 20;
const int ModNum = 3;

int main(){
    bool isInGroup[PersonNum];
    int personInGroup = PersonNum;
    for(int i = 0; i<PersonNum; i++)
        isInGroup[i] = true;
    int counter = 0;
    while(personInGroup>0){
        for(int j = 0; j<PersonNum; j++){
            if(!isInGroup[j]) continue;
            if(!(++counter%ModNum)){
                isInGroup[j] = false;
                personInGroup--;
                cout<<PersonNum - personInGroup<<") 编号"<<j
            }
        }
    }
}
```

```

        <<"的人离开圈子！"<<endl;
    }
}
}
return 0;
}

// 单层循环的实现
#include <iostream>
#include <iomanip>
using namespace std;
const int PersonNum = 20;
const int ModNum = 3;

int main(){
    int next[PersonNum]; // 定义next数组，next[i]表示
                        // 报数时编号i的人的下一个人的编号
    int personInGroup = PersonNum;
    for(int i = 0; i<PersonNum; i++) // 给next数组赋初值
        next[i] = (i + 1) % PersonNum;
    int counter = 0, currentNo = 0, previousNo = 19;
    while(personInGroup>0){
        if(!(++counter%ModNum)){
            personInGroup--;
            cout<<setw(3)<<PersonNum - personInGroup<<" 编号"
                <<currentNo<<"的人离开圈子！"<<endl;
            next[previousNo] = next[currentNo];
        }
        else
            previousNo = currentNo;
        currentNo = next[currentNo];
    }

    return 0;
}

```