

<http://course.pku.edu.cn/webapps/login/>



北大教学网
TEACHING AND LEARNING@PKU

建设互联互通的网络学习



首页

教学网在线

开放课程

精品课程

培训与服务

关于教学网

统计信息

每周二下午2点 电教403

“北大教学网体验活动”

欢迎同学们和助教参加

欢迎同学们和助教参加“北大教学网体验活动”



请在此登录

login

用户名



密码



访客登录

登录

通知公告

更多

最近更新课程

课程访问排行



北京大学



助教与分组

- 马家宽 (majk06@sei.pku.edu.cn)
 - ◆ 负责: 学号 \leq 00948046 的同学
- 郭金罡 (taguojg@gmail.com)
 - ◆ 负责: 00948047 \leq 学号 \leq 00948135
- 张 纯 (zhangchun08@sei.pku.edu.cn)
 - ◆ 负责: 00948138 \leq 学号 \leq 00948225
- 李 强 (liqiang08@sei.pku.edu.cn)
 - ◆ 负责: 学号 \geq 00948226





课程网站与作业

■ 上机时间:

- ◆ 每周五晚 19:00 – 21:00;

■ 上机地点:

- ◆ 计算机系机房（理科一号楼二层北侧）

■ 考勤要求:

- ◆ 不按要求上机视同旷课;

■ 座位安排:

- ◆ 每个同学使用一个固定机位
- ◆ 助教将统计每位同学的座位号码;



北京大学



《计算概论A》课程 程序设计部分

从现实问题到计算机程序

李 戈

北京大学 信息科学技术学院 软件研究所

2009年9月24日



北京大学

回顾——最初的程序

```
#include<iostream>
using namespace std;
int main( )
{
    int number[45] = {78, 56, 69, 31, 36, 67, 31, 47, 69, 34, 45, 74, 61, 82,
43, 41, 76, 79, 81, 66, 54, 50, 76, 51, 53, 28, 74, 39, 45, 61, 52, 41, 43, 75,
78, 84, 72, 51, 43, 64, 75, 81, 69, 55, 74};
    int max = 0;
    int i = 0;
    for(i = 0; i < 45; i++)
    {
        if(number[i] > max)
            max = number[i];
    }
    cout<<"The Maximal Number is:"<<max;
    return 0;
}
```



回顾——关于变量的定义和使用

■ 变量的定义及初始化

- ◆ 变量类型 变量名称;

- ◆ `int i;`

- ◆ `float result;`

变量的初始化:

- ◆ `int i = 0;`

相当于:

- ◆ `int i;`

- ◆ `i = 0;`

```
#include <iostream.h>
```

```
void main()
```

```
{
```

```
    int i;
```

```
    cout<<"请输入一个整数: ";
```

```
    cin>>i;
```

```
    cout<<"您输入的整数是: "<<i<<endl;
```

```
}
```



北京大学

新——字符型数据

■ 字符型变量

◆ `char a = 'a';`

◆ 用单引号括起来的单个字符。如 `'a'`, `'x'`, `'D'`, `'#'`

```
#include <iostream>
using namespace std;
int main( )
{
    char a = 'a';
    cout<<a<<endl;
    cout<<"This is the first line ! \n";
    cout<<'a'<<endl;
    return 0;
}
```





新——数组

■ 定义其他类型的数组并附初始值

- ◆ `int a[26];`

- ◆ `float a[26];`

- ◆ `bool a[26];`

- ◆ `char a[26] = {'a', 'b', 'c', ..., 'z'};`

■ 数组的使用

- ◆ 要使用第 n 个数: `a[n-1];` `a[n-1] = 1+2;`

- ◆ 注意: 对数组`a[n]`, 只能使用`a[0] – a[n-1]`;
`a[n]`不能用;



北京大學

数组的使用

```
#include<iostream>
using namespace std;
int main( )
{
    int number[45] = {78, 56, 69, 31, 36, 67, 31, 47, 69, 34, 45, 74, 61, 82,
        43, 41, 76, 79, 81, 66, 54, 50, 76, 51, 53, 28, 74, 39, 45, 61, 52, 41, 43, 75,
        78, 84, 72, 51, 43, 64, 75, 81, 69, 55, 74};
    int max = 0;
    int i = 0;
    for(i = 0; i < 45; i++)
    {
        if(number[i] > max)
            max = number[i];
    }
    cout<<"The Maximal Number is:"<<max;
    return 0;
}
```





回顾——if 语句

◆形式一

if (条件表达式)

 单一语句;

else

 单一语句;

最常用的条件表达式:

$A > B$, $A \geq B$

$A < B$, $A \leq B$

$A == B$

◆形式二

if (条件表达式)

{

 语句组;

}

else

{

 语句组;

}



北京大学



回顾——if语句（例）

```
#include<iostream>
using namespace std;
int main( )
{
    int age;
    cout<<"Input your age:"<<endl;
    cin>>age;
    if(age >= 18)
        cout<<"You are a man/woman!"<<endl;
    else
        cout<<"You are a boy/girl!"<<endl;
}
```



回顾——for语句（例）

```
#include<iostream>
using namespace std;
int main()
{
    int i = 0, sum = 0;
    for(i=0; i<=99; i++)
    {
        if(i % 2==0)
            sum = sum + i;
    }
    cout<<"The total is "<<sum<<endl;
    return 0;
}
```

- 计算并输出前N(设N为99)个自然数中所有偶数之和。



回顾——输入/输出

■ 输出流对象如何从程序中输出信息？

◆ `cout<<"Webcome to Beijing!"<<endl;`

■ 输入流对象如何向程序中输入信息？

◆ `int x; cin>> x;`

```
#include<iostream>
using namespace std;
int main() {
    int a,b,c;
    cin>>a>>b>>c;
    cout<<"a = "<<a<<"b = "<<b<<"c = "<<c<<endl;
    return 0;
}
```




回顾——学会利用函数

- 计算 $\sqrt{2^3 + 4^2}$
- 用到的数学函数 `sqrt (x)``pow(x,n);`
`#include <iostream>`
`#include <cmath>`
`using namespace std;`
`int main()`
`{`
`int a = 2, b = 3, c = 4, d = 2;`
`int nPow1 = pow(a, b), nPow2 = pow(4, 2);`
`float fResult = sqrt(nPow1+nPow2);`
`cout << fResult;`
`return 0;`
`}`
- 输出：4.89898



自定义函数

```
#include <iostream>
using namespace std;
int add(int x, int y);
int main( )
```

//这是函数的声明;

```
{
```

```
    int a = 30, b = 40, c = 0;
```

```
    cout<<"a="<<a<<",b="<<b<<endl;
```

```
    c = add(a, b);
```

//这是函数的调用;

```
    cout<<"a + b = "<<c<<endl;
```

```
    return 0;
```

```
}
```

```
int add(int x, int y)
```

//这是一个函数的定义;

```
{
```

```
    int z=x;
```

```
    x=y;
```

```
    y=z;
```

```
    return z;
```

```
}
```





面对问题，如何编写程序？



清华大学



探求程序的本质

■ 软件

- ◆ 软件是现实世界在计算机系统映射;
- ◆ 软件 = 程序 + 文档
 - 程序是对计算任务的处理对象和处理规则的描述。

■ 探求程序的本质

- ◆ 程序是现实世界的解决方案在计算机系统映射;



清华大学

软件开发的最终目的

■ 程序的开发

- ◆ 对所要处理的问题域进行正确的理解
- ◆ 把这种正确的理解正确的描述出来



北京大学



没有解决方案就没有程序！

■ 现实世界中的解决方案

- ◆ 为解决一个问题而采取的方法和步骤

■ 计算机系统上的算法

- ◆ 适用于计算机系统的解决某个问题的方法和步骤

■ 写程序“三部曲”

- (1) 先思考出现实世界中问题的解决方案；
- (2) 将现实世界的解决方案映射为计算机系统的算法；
- (3) 用高级程序设计语言精确地描述算法；



北京大学



现实问题

解决方案

计算机程序



北京大学



示例1 鸡兔同笼问题

■ 问题描述

- ◆ 一个笼子里面关了鸡和兔子（鸡有2只脚，兔子有4只脚，没有例外）。已经知道了笼子里面脚的总数 a ，问笼子里面至少有多少只动物，至多有多少只动物？





程序举例

■ 输入数据

- ◆ 第1行是测试数据的组数 n ，后面跟着 n 行输入。每组测试数据占1行，包括一个正整数 a (a 小于32768)。

■ 输出要求

- ◆ n 行，每行输出对应一个输入。输出是两个正整数，第一个是最少的动物数，第二个是最多的动物数，两个正整数用空格分开。如果没有满足要求的情况出现，则输出2个0。





程序举例

■ 输入样例

2

3

20

■ 输出样例

0 0

5 10



北京大学



解决方案

■ 问题抽取:

- ◆ 给一个脚数, 判定至少多少只, 至多有多少只动物?

■ 解决方法:

- ◆ 若脚数为奇数, 错
- ◆ 若脚数为偶数,
 - 则动物最多为 $\text{脚数}/2$
 - 最少为 $\text{脚数}/4$ 或 $\text{脚数}/4 + 1$





解决方案 整理

- 接收输入的“组数”
- 循环（组数）接收输入的“脚数”

对于每个输入的脚步数

- ◆ 如果是奇数，错
- ◆ 如果是偶数，
 - 最多的动物数为 脚数/2
 - 最少的动物数
 - ◆ 如果脚数能被4整除则 最少：脚数/4
 - ◆ 如果脚数不能被4整除则 最少：脚数/4 + 1



```

#include <iostream>
using namespace std;
int main( )
{
    int nCases, i, nFeet;
    //nCases 表示输入测试数据的组数, nFeet 表示输入的脚步数。
    cin>>nCases;
    for(i = 0; i < nCases; i++){
        cin>>nFeet;
        if(nFeet %2 != 0) // 如果有奇数只脚, 则输入不正确,
            // 因为不论2 只还是4 只, 都是偶数
            cout<<"0 0"<<endl;
        else if (nFeet%4 != 0)
            //若要动物数目最少, 使动物尽量有4 只脚
            //若要动物数目最多, 使动物尽量有2 只脚
            cout<<nFeet / 4 + 1<<" "<<nFeet / 2<<endl;
        else
            cout<<nFeet / 4<<" "<<nFeet / 2<<endl;
    }
    return 0;
}

```



示例2 百元买百鸡问题

■ 问题描述

- ◆ 假定小鸡每只 5 角，公鸡每只 2 元，母鸡每只 3 元。现在有 100 元钱要求买 100 只鸡，编程列出所有可能的购鸡方案。





解决方案

- 设母鸡、公鸡、小鸡各为 x 、 y 、 z 只，根据题意列方程：

$$\begin{cases} x+y+z=100 \\ 3x+2y+0.5z=100 \end{cases}$$

- 穷举法（枚举法）

- ◆ 将可能出现的各种情况一一测试，判断是否满足条件；





解决方案整理

■ 最简单的办法:

- ◆ 循环: 对于每个 $X \leq 33$

- 循环: 对于每个 $y \leq 50$

- ◆ 循环: 对于每个 $z \leq 100$

- * 如果 $x + y + z = 100$

- 并且 $3x + 2y + 0.5z = 100$

- 则 打印输出;



北京大学

```
#include <iostream>
using namespace std;
int main()
{
    int x,y,z;
    cout<<"\t 母鸡\t\t 公鸡\t\t 小鸡"<<endl;
    for(x=0;x<=33;x++)
        for(y=0;y<=50;y++)
            for(z = 0; z<=100; z++)
                {
                    if((x+y+z) ==100)
                        if((3*x+2*y+0.5*z)==100)
                            cout<<"\t"<<x<<"\t\t"<<y<<"\t\t"<<z<<endl;
                }
    return 0;
}
```



稍作简化的解决方案

- 由于 $x + y + z = 100$, 且 $x \leq 33$, $y \leq 50$, $z < 100$;
- 最简单的办法:
 - ◆ 循环: 对于每个 $x \leq 33$
 - 循环: 对于每个 $y \leq 50$
 - ◆ 对于每个 $z = 100 - x - y$
 - * 如果 $3x + 2y + 0.5z = 100$
则 打印输出;



```
#include <iostream>
using namespace std;
int main()
{
    int x, y, z;
    cout<<"\t 母鸡\t\t 公鸡\t\t 小鸡"<<endl;
    for(x=0; x<=33; x++)
        for(y=0; y<=50; y++)
        {
            z=100-x-y;
            if((3*x+2*y+0.5*z)==100)
                cout<<"\t"<<x<<"\t\t"<<y<<"\t\t"<<z<<endl;
        }
    return 0;
}
```



示例3 数字统计

■ 问题:

- ◆ 输入1000个0 ~ 50的整数，请你统计每个数在输入数列中出现的次数。

■ 解:

- ◆ 你的解决方案?





数字统计

■ 方法一:

```
for (i = 1; i <= 1000; i++)  
{  
    cin >> num;  
    for (j = 0; j < 50; j++)  
    {  
        if (num==j)  
            count[j]++;  
    }  
}
```





http://poj.grids.cn



Online Judge	Problem Set	Authors	Online Contests	User
Web Board Home Page F.A.Qs Announcement	Problems Submit Problem Online Status Prob.ID: <input type="text"/> <input type="button" value="Go"/>	Register Update your info Authors ranklist <input type="text"/> <input type="button" value="Search"/>	Current Contest Past Contests Scheduled Contests Award Contest	User ID: <input type="text"/> Password: <input type="password"/> <input type="button" value="login"/> Register

Welcome To PKU JudgeOnline

Problem Set is the place where you can find large amount of problems from different programming contests. Online Judge System allows you to test your solution for every problem.

First of all, read carefully [Frequently Asked Questions](#).

Then, choose a [problem](#), solve it and [submit](#) your solution.

If you want to publish your problems or setup your own online contest, just [write us](#).



Designer & Developer: Xu Pengcheng Ying Fuchen Xie Di



《程序设计导引及在线实践》



好好想想，有没有问题？

谢谢！



清华大学