高级程序设计语言实验报告

——Bluefissure

目录

2014 年 12 月 23 日

常见问题

1. 使用 scan.next()以及 scan.nextInt()等方法后若想利用 scan.nextLine()读入字符串，会只读一个换行符。
2. 忽视+号的意思，多次在 println 方法中以单引号表示空格，与整数相加，结果表示成为了另一个整数。
3. 写实验九时还不知道如何实现 Comparable 接口，调用 Arrays.sort()对数组排序时各种报错。

解决方法

1. 使用 scan.next()以及 scan.nextInt()等方法后若想利用 scan.nextLine()读入字符串，先用 scan.nextLine() “吃掉”换行符。
2. 在与整数以+连接时，用双引号表示字符。
3. 查 Api。

具体代码

实验 2：输入输出小程序

a) 实验目的

本实验对应课本第二章知识，交互式应用程序：

- 掌握利用 Scanner 获取数据，println 输出结果的基本方法
- 掌握 Java 中的基本数据类型、变量、表达式以及数据类型转化等基本概念

b) 实验内容

1、编写一个程序，从键盘读入三位十进制数，以八进制的形式输出,要求程序有较友好的交换过程、源代码撰写较规范。

2、编写一个程序，从键盘读入一元二次方程 $ax+bx+c=0$ 的系数 a，b，c，计算输出其两个根。

c) 完成报告

```
package Experiment02;
import java.util.*;
public class Experiment02_1 {
```

```
    public static void main(String[] args)
    {
        Scanner in=new Scanner (System.in);
        System.out.println("Please input a decimal number:");
        int n=in.nextInt();
        String s="";
        while(n>0)
        {
            s=(char)(n%8+'0')+s;
            n/=8;
        }
        System.out.println("The octal number is:\n"+s);
    }
}
```

```
package Experiment02;
import java.util.Scanner;
public class Experiment02_2
{
    public static void main(String[] args)
    {
        Scanner scan = new Scanner(System.in);
        double m,x1,x2,a,b,c;
        System.out.println("Please input a,b,c:");
            a = scan.nextInt();
            b = scan.nextInt();
            c = scan.nextInt();
            m = (double)b * b - 4 * a * c;
        if(a == 0)
        {
            if(b == 0)
            {
                if(c == 0)
                    System.out.println("Infinite");
                else
                    System.out.println("No solution");
            }
            else
            {
                x1 = - c /(double)b+0;
                System.out.println(String.format("The answer is\n%.2f",x1));

            }
```

```
        }
        else if(m < 0)
            System.out.println("No solution");
        else if(m == 0)
        {
            x1 = -b / (double)(2 * a)+0;
            System.out.println(String.format("The answer is\n%.2f",x1));
        }
        else if(m>0){
            x1 = (-b + Math.sqrt(m)) / (2 * a)+0;
            x2 = (-b - Math.sqrt(m)) / (2 * a)+0;
            if(x1 > x2)
                System.out.println(String.format("The answer is\n%.2f",x2)
+" " + String.format("%.2f",x1));
            else
                System.out.println(String.format("The answer is\n%.2f",x1)
+" " + String.format("%.2f",x2));
        }
    }
}
```

实验 3：预定义类与自定义类

a) 实验目的

本实验对应课本第三、第四章知识

- 掌握字符串、数学类等主要预定义类的使用方法
- 掌握自定义类的一般方法

b) 实验内容

编写一个程序，实现如下功能：

自定义一个银行账户类，包括用户名，账号，余额等属性以及存钱、取钱、加利息、查询余额等方法。

模拟两个用户的各 8 次交易，包括创建账户，取钱，存钱，查询余额，加息。创建账户所需的信息由键盘输入；存钱、取钱的数额由随机数模拟，随机数上限由键盘输入。

每次账户交易在屏幕打印交易后的账户信息

c) 完成报告

```
package Experiment03;
import java.util.*;
class BankAccount
{
    Scanner in= new Scanner(System.in);
    String name="",password="";
    double interest;
    int ID=0;
    double balance=0;
```

```
    void setName(String s){name=s;}
    void setID(int i){ID=i;}
    void setPassword(String s){password=s;}
    void deposit(double money){balance+=money;}
    void Withdraw(double money){balance-=money;}
    void Plusinterest(double m){interest+=m;}
    void Information()
    {
        String s="";
        Integer id=ID;
        for(int i=1;i<=8-id.toString().length();i++)
            s+="0";
        System.out.println("Your information:");
        System.out.println("\tName:"+name);
        System.out.println("\tID:"+s+ID);
        System.out.println("\tBalance:"+balance);
        System.out.println("\tinterest:"+interest);
        in.nextLine();
    }
}
public class Experiment03_1 {
    public static void main(String[] args) {
        Scanner in= new Scanner(System.in);
        Random rand=new Random();
        int tot=0;
        BankAccount []cus=new BankAccount[1001];
        System.out.println("Welcome to the ZYP's BANK System.");
        in.nextLine();
        while(true)
        {
            System.out.println("Please select what you want to do:");
            System.out.println("1.Rigister");
            System.out.println("2.Login");
            System.out.println("3.Exit");
            int i=in.nextInt();
            in.nextLine();
            if(i==3) break;
            switch(i)
            {
                case 1:
                {
                    tot++;
                    cus[tot]=new BankAccount();
                    System.out.println("Please input your name:");
```

```
                    cus[tot].setName(in.nextLine());
                    cus[tot].setID(tot);
                    String p="null",pp="";
                    do
                    {
                        System.out.println("Please input your password:");
                        p=in.nextLine();
                        System.out.println("Please varify your password:");
                        pp=in.nextLine();
                        if(p.compareTo(pp)!=0)
                            System.out.println("You entered two different
passwords.");
                    }while(p.compareTo(pp)!=0);

                    cus[tot].setPassword(p);

                    System.out.println("Registered Successfully!");

                    cus[tot].Information();
                    break;
                }

                case 2:
                {
                    System.out.println("Please input your ID:");
                    int k=in.nextInt();in.nextLine();
                    if(k>tot||k<=0)
                    {
                        System.out.println("Wrong ID!");
                        in.nextLine();
                        break;
                    }
                    System.out.println("Please input your Password:");
                    String pwd=in.nextLine();
                    if(pwd.compareTo(cus[k].password)!=0)
                    {
                        System.out.println("Wrong Password!");
                        in.nextLine();
                        break;
                    }
                    System.out.println("Login Successfully!");
                    in.nextLine();

                    while(true)
```

```
                    {
                        System.out.println("Please select what you want to
do:");
                        System.out.println("1.deposit Money");
                        System.out.println("2.Whithdraw Mongey");
                        System.out.println("3.Inquire Balance");
                        System.out.println("4.Plus interest");
                        System.out.println("5.Logout");
                        int j=in.nextInt();
                        if(j==5) break;

                        switch (j)
                        {
                            case 1:
                            {
                                System.out.println("Please input the max
money you want to save:");
                                double lim=in.nextDouble();
                                double money=rand.nextDouble()*lim;
                                cus[k].deposit(money);
                                break;
                            }
                            case 2:
                            {
                                System.out.println("Please input the max
money you want to withdraw:");
                                double lim=in.nextDouble();
                                double money=rand.nextDouble()*lim;
                                if(cus[k].balance-money>=0)
                                    cus[k].balance-=money;
                                else
                                {
                                    System.out.println("You do not have
enough balance!");
                                    in.nextLine();

                                }
                                break;
                            }
                            case 3:
                            {
                                System.out.println("Your balance
is:"+cus[k].balance);
                                in.nextLine();in.nextLine();
```

```
                            break;
                        }
                        case 4:
                        {
                            System.out.println("The interest will be
added between between 0 and 1 randomly.");
                            double inte=rand.nextDouble();
                            cus[k].interest+=inte;
                            in.nextLine();
                            break;
                        }
                    }
                    if(j!=3)
                        cus[k].Information();
                }

                break;
            }

        }
    }
}
```

实验 4：分支循环初步

a)实验目的

尝试阅读理解简单的分支、循环程序

b)实验内容：

1、编写程序，实现 WSAD”方向控制。功能描述：编写一个程序，当键盘输入为“WSAD”这四个按键中的其中一个（大小写皆可），输出相对应的方向。比如用户输入“W”，输出“左”。

2、编写程序，实现 Hi - Lo 猜猜游戏程序。实验描述：从 1-1000 中随机选择一个数，反复让用户猜该数字是什么，直到用户猜对或用户退出为止。每猜一次告诉用户猜测的结果是对还是过大或是过小。使用一个标识值确定用户是否想退出。当用户猜对时报告其猜测的次数。每次游戏结束时询问用户是否想继续玩，直到用户选择结束。

c)完成报告

```
package Experiment04;
import java.util.*;
import java.io.*;

public class Experiment04_1 {
    public static void main(String[] args)
    {
        Scanner in= new Scanner(System.in);
        String str;
        while(in.hasNext())
        {
            str=in.nextLine();
            if(str.toCharArray()[0]=='W'||str.toCharArray()[0]=='w')
                System.out.println("Up");
            else if(str.toCharArray()[0]=='A'||str.toCharArray()[0]=='a')
                System.out.println("Left");
            else if(str.toCharArray()[0]=='S'||str.toCharArray()[0]=='s')
                System.out.println("Down");
            else if(str.toCharArray()[0]=='D'||str.toCharArray()[0]=='d')
                System.out.println("Right");
            else
            {
                System.out.println("Wrong Input!");
                break;
            }
        }
    }
}


package Experiment04;
import java.util.*;
public class Experiment04_2 {
    public static void main(String[] args)
    {
        Scanner in= new Scanner(System.in);
        Random rand=new Random();
        int num;
        int guess;
        boolean f=true;
        while(f)
        {
            num=rand.nextInt(1000)+1;
            guess=-1;
```

```
            int time=0;
            while(guess!=num)
            {
                System.out.println("Please guess a number:");
                guess=in.nextInt();
                time++;
                if(guess>num)
                    System.out.println("The number you guessed is larger
than the answer.");
                else if(guess<num)
                    System.out.println("The number you guessed is smaller
than the answer.");
                else if(guess==num)
                {
                    System.out.println("You won!!\nYou've guessed "+time+"
times.");
                    in.nextLine();
                }
            }
            System.out.println("Would you play again?\n1.Yes\n2.No\n");
            f=(in.nextInt()==1);
        }
    }
}
```

实验 5：四则运算器

a)实验目的

本实验对应课本第五、六章，控制流程：

- 学习、理解 Scanner 扩展应用
- 尝试阅读理解简单的分支、循环程序
- 更深入学习的利用 Debug 工具分析程序控制流程
- 简单的文本处理

b)实验内容：

阅读程序，完成下面要求

/**********

要求

1 Debug 模式运行分析该程序，写出该程序的作用
2 补充完整该程序，使得其可以正确计算 data.txt 中的所有运算，并友好的输出结果
3 拷贝 data.txt 的路径到 d 盘根目录下，如何使程序正确运行
4 修改该程序，使其支持 data2.txt 数据处理
5 (optional)修改程序，列举溢出、崩溃、文件未找到 java.io.FileNotFoundException 等运行时或运行后错误
6 （Optional）修改程序，使其支持 data3.txt
**********/

c)完成报告

```
package Experiment05;

import java.io.File;
import java.util.Scanner;
public class T02Scanner {
    public static void main(String args[]) throws Exception {

//      String filename = "D:\\data.txt";
        String filename = "data.txt";
        int op1,op2,result=0;
        String operator ="";
        // create a scanner from the data file
        Scanner scanner = new Scanner(new File(filename));

        // 重复从文件中读取数据
        while (scanner.hasNext()) {
            // retrieve each data element
            operator = scanner.next();
            op1 = scanner.nextInt();
            op2 = scanner.nextInt();

            if (operator.equals("+"))
                result = op1 + op2;
            else if (operator.equals("-"))
                result = op1 - op2;
            System.out.println(op1+" "+operator+" "+op2+" is " + result);
//          System.out.println("result is " + result);
        }

        scanner.close(); // also closes the File
    }
}
```

```
package Experiment05;

import java.io.File;
import java.util.*;

public class T03Scanner {
    public static void main(String args[]) throws Exception {

        String filename = "data3.txt";
        int op1, op2, result = 0, i;
        String operator = "";
        // create a scanner from the data file
        Scanner scanner = new Scanner(new File(filename));
        final int maxn = 10000;
        int a[] = new int[maxn];
        char op[] = new char[maxn];
        int lengtha = 0, lengthop = 0;
        while (scanner.hasNext()) {

            lengtha = 0;
            lengthop = 0;

            String strr, str;
            strr = scanner.nextLine();
            Scanner scan = new Scanner(strr);
            while (scan.hasNext()) {
                str = scan.next();
                if (Character.isDigit(str.charAt(0))) {
                    a[lengtha++] = Integer.parseInt(str);
                } else {
                    if (str.equals("+") || str.equals("-"))
                        op[lengthop++] = str.charAt(0);
                    else {
                        if (str.equals("*"))
                            a[lengtha - 1] *= scan.nextInt();
                        else if (str.equals("/"))
                            a[lengtha - 1] /= scan.nextInt();
                    }
                }

            }
            result = a[0];
            for (i = 0; i < lengthop; i++) {
                if (op[i] == ('+'))
```

```
                    result += a[i + 1];
                else
                    result -= a[i + 1];
            }

            System.out.println(result);
        }
        scanner.close(); // also closes the File
    }
}
```

实验 6：文本处理

a)实验目的

本实验对应课本第五章控制流程以及第三章，Java 预定义类

- 综合运用控制流程
- 学习使用 String，Random 等类。

b)实验内容：

阅读程序，完成下面要求

```
/************
 * 字符串及随机数
 * 下面程序作用是统计字符串中字符's'所占比例
 * 1、调试程序，使输出正确结果
 * 2、参考 T02Scanner，修改程序，支持从文本文件中统计字符's'所占比例
 * 3、利用随机采样的方法，统计该文本文件中's'所占比例（例如，随机选择 10000 个字符，计算选中's'的比例）
 * 4、搜集 10 个英文文件，分别统计's"z'的出现频率
 * 5、尝试总结采集样本数与结果准确度间的关系
************/
```

c)完成报告

```
package Experiment06;
import java.util.Random;
import java.util.Scanner;
import java.text.Format;
import java.text.NumberFormat;
import java.io.*;
public class T03StringProc {
```

```
    public static void main(String[] args) throws Exception {
        String filename = "data1.txt";
        Scanner scanner = new Scanner(new File(filename));
        String str=scanner.nextLine();
        int count = 0;
        for (int i = 0; i< str.length(); i++){
            if (str.charAt(i) == 's'){
                count++;
            }
        }
        System.out.println("percentage of 's' is " +
(double)count/str.length());
        int tot=0;
        count = 0;
        final int MAX=100;
        Random rand=new Random();
        for (int i = 0; tot<=MAX; i=(int)(rand.nextInt(str.length()))){
            tot++;
            if (str.charAt(i) == 's'){
                count++;
            }
        }
        System.out.println("percentage of 's' is " + (double)count/tot);

    }
}


package Experiment06;
import java.util.Random;
import java.util.Scanner;
import java.text.Format;
import java.text.NumberFormat;
import java.io.*;
public class T03StringProc_4 {
    public static void main(String[] args) throws Exception {
    for(int u=1;u<=10;u++)
    {
        String filename = "data"+u+".txt";
        Scanner scanner = new Scanner(new File(filename));
        String str=scanner.nextLine();
        int count = 0;
        for (int i = 0; i< str.length(); i++){
            if (str.charAt(i) == 's'){
```

```
                count++;
            }
        }
        System.out.println("In "+filename+" percentage of 's' is " +
(double)count/str.length());
        int tot=0;
        count = 0;
        final int MAX=10000;
        Random rand=new Random();
        for (int i = 0; tot<=MAX; i=(int)(rand.nextInt(str.length()))){
            tot++;
            if (str.charAt(i) == 's'){
                count++;
            }
        }
        System.out.println("In "+filename+" percentage of 's' is about " +
(double)count/tot+"\n");
         count = 0;
        for (int i = 0; i< str.length(); i++){
            if (str.charAt(i) == 'z'){
                count++;
            }
        }
        System.out.println("In "+filename+" percentage of 'z' is " +
(double)count/str.length());
         tot=0;
        count = 0;
        for (int i = 0; tot<=MAX; i=(int)(rand.nextInt(str.length()))){
            tot++;
            if (str.charAt(i) == 'z'){
                count++;
            }
        }
        System.out.println("In "+filename+" percentage of 'z' is about " +
(double)count/tot+"\n");
    }
    }
}
```

实验 7：日历输出程序 Java 控制流程

a)实验目的

更深入的理解 Java 控制流程

b)实验内容：

基本功能：输入一个月份，给出 2013 年这个月的日历，日历要求每行显示 7 列，对应星期一到星期日；
扩展功能：输入一个月份，同时输出该月起始的两个月的日历，要求两个月的日历水平排列而非上下排列。

c)完成报告

```
package Experiment07;
import java.util.*;
public class Experiment07_1 {

    static int Zeller(int y,int m,int d)
    {

        int c=y/100;
        y%=100;
        int w=c/4-2*c+y+y/4+(13*(m+1)/5)+d-1;
        return w%7;
    }
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.print("Please input the month and year:");
        int month=scan.nextInt();
        int year=scan.nextInt();

        int maxd=0;
        boolean f=false;
        if(year%400==0||(year%100!=0&&year%4==0))
                    f=true;
        switch(month)
        {
        case 1:case 3:case 5:case 7:case 8:case 10:case 12:
            {maxd=31;break;}
        case 4:case 6:case 9:case 11:
            {maxd=30;break;}
        case 2:
            {maxd=f?29:28;break;}
        }
```

```
        System.out.println("Sun.\tMon.\tTues.\tWed.\tThurs.\tFri.\tSat.");
        int begin=Zeller(year,month,1);
        int p=1;
        for(int j=1;j<=begin;j++){
            System.out.print("\t");p++;}

        for(int i=1;i<=maxd;i++)
        {

            System.out.print(i);
            if(p++%7==0)
                System.out.println();
            else System.out.print("\t");
        }
    }
}
```

```
package Experiment07;
import java.util.*;
public class Experiment07_2{

    static int Zeller(int y,int m,int d)
    {

        int c=y/100;
        y%=100;
        int w=c/4-2*c+y+y/4+(13*(m+1)/5)+d-1;
        return w%7;
    }
    static int maxdays(int month,int year)
    {
        int maxd=0;
        boolean f=false;
        if(year%400==0||(year%100!=0&&year%4==0))
        {
                    f=true;
        }
        switch(month)
        {
        case 1:case 3:case 5:case 7:case 8:case 10:case 12:
            {maxd=31;break;}
        case 4:case 6:case 9:case 11:
```

```
            {maxd=30;break;}
        case 2:
            {maxd=f?29:28;break;}
        }
        return maxd;
    }
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.print("Please input the month and year:");
        int month=scan.nextInt();
        int year=scan.nextInt();

        int maxd=0,maxdp;
        maxd=maxdays(month,year);
        if(month<12)
            maxdp=maxdays(month+1,year);
        else
            maxdp=maxdays(1,year+1);
    System.out.print("Sun.\tMon.\tTues.\tWed.\tThurs.\tFri.\tSat.\t\t");
        System.out.println("Sun.\tMon.\tTues.\tWed.\tThurs.\tFri.\tSat.");
        int begin=Zeller(year,month,1);
        int beginp=(begin+maxd)%7;
        int i,j,p=1,pp=1,d1=1,d2=1;
        for(i=1;i<=6;i++)
        {
            for(j=1;j<=14;j++)
            {
                if(j<=7)
                {
                    int temp=j+(i-1)*7;
                    if(temp<=begin)
                        System.out.print("\t");
                    else
                    {
                        if(p<=maxd)
                            System.out.print(p+"\t");
                        else
                            System.out.print("\t");
                        p++;
                    }
                    if(j==7) System.out.print("\t");
                }
                else
                {
```

```
                int temp=(j-7)+(i-1)*7;
                if(temp<=beginp)
                    System.out.print("\t");
                else
                {
                    if(pp<=maxdp)
                    System.out.print(pp+"\t");
                    else
                        System.out.print("\t");
                    pp++;
                }
                if(j==14)
                    System.out.println();
            }
        }
    }
}
```

实验 8 自定义 ArrayList

a)实验目的

本部分对应课本第八章，帮助学生深入理解数组使用方法

b)实验内容：

利用数组实现 Java ArrayList 类的基本功能，要求实现方法包括：

Add, Insert, Delete, Find

编写一个测试应用，测试该自定义类的正确性

c)完成报告

```
package Experiment08;
import java.util.*;
class Array
{
    int maxn=1001,length=0;
    int []a=new int[maxn];
    void clear()
    {
        for(int i=0;i<=1000;i++)
            a[i]=0;
    }
```

```
    void add(int num){a[length++]=num;}
    void insert(int dex,int num)
    {
        for(int i=length-1;i>dex;i--)
            a[i+1]=a[i];
        a[dex+1]=num;
        length++;
    }
    void delete(int dex)
    {
        for(int i=dex;i<length;i++)
            a[i]=a[i+1];
        a[length--]=0;
    }
    int find(int num)
    {
        for(int i=0;i<length;i++)
            if(a[i]==num) return i;
        return -1;
    }
    void print()
    {
        for(int i=0;i<length;i++)
            System.out.print(a[i]+" ");
        Scanner in=new Scanner(System.in);
        System.out.print("\n");
    }
}
public class Experiment08_1 {
    public static void main(String[] args)
    {
        Array arr=new Array();
        Scanner in=new Scanner(System.in);
        System.out.println("Please input N:");
        int n=in.nextInt();
        System.out.println("Please input N numbers to add:");
        for(int i=1;i<=n;i++)
        {
            arr.add(in.nextInt());
        }
        arr.print();
        System.out.println("Please input the number to add:");
        arr.add(in.nextInt());
        arr.print();
```

```
        int m;
        System.out.println("Please input the dex and num:");
        n=in.nextInt();m=in.nextInt();
        arr.insert(n, m);
        arr.print();
        System.out.println("Please input the dex to delete:");
        n=in.nextInt();
        arr.delete(n);
        arr.print();
        System.out.println("Please input the numer you want to find:");
        m=in.nextInt();
        System.out.println(arr.find(m));
    }
}
```

实验 9 图形家族-继承与多态

a)实验目的

本实验涉及的知识点主要为 继承、多态、排序，对应课本 9、10 章知识点

b)实验内容：

1、矩形、正方形、椭圆、圆形、六边形、正六边形都是形状，请以形状（Shape）为最顶层的类，设计出一个层次化的类结构，至少能够对每个形状命名，并求面积、周长
2、写一个程序，创建若干形状，存储到数组中，实现排序算法，可以根据面积、周长对创建的图形实现排序。

c)完成报告

```
package Experiment09;
import java.util.*;
abstract class Shape implements Comparable<Shape>
{
    private int comtype=0;
    protected double circum,area;
    protected String name;
    public Shape()
    {
        name=null;
        circum=area=0;
    }
    public Shape(String name)
    {
```

```
        this.name=name;
    }
    public void changecomtype()
    {
        this.comtype=(comtype==0)?1:0;
    }
    public abstract double getCircum();
    public abstract double getArea();
    public int compareTo(Shape s)
    {
        if(comtype==0)
        {
            return (this.area>s.area)?1:((this.area<s.area)?-1:0);
        }
        else
        {
            return (this.circum>s.circum)?1:((this.circum<s.circum)?-1:0);
        }
    }
}
class Ellipse extends Shape
{
    double a,b;
    public Ellipse(){};
    public Ellipse(double a,double b)
        {this.a=a;this.b=b;}
    public double getCircum()
        {circum=2*Math.PI*b+4*(a-b);return circum;}
    public double getArea()
        {area=Math.PI*a*b;return area;}
}
class Circle extends Ellipse
{
    double r;
    public Circle(double r)
    {
        a=r;b=r;
        this.r=r;
    }
}

class Rectangle extends Shape
{
    double l,w;
```

```
    public Rectangle(){};
    public Rectangle(double l,double w)
    {
        this.l=l;this.w=w;
    }
    public double getCircum()
        {circum=2*(l+w);return circum;}
    public double getArea()
        {area=l*w;return area;}
}

class Square extends Rectangle
{
    double a;
    public Square(){};
    public Square(double a)
    {
        l=a;w=a;
    }
}
class Hexagon extends Shape
{
    double r1,r2,r3,r4,r5,r6;
    public Hexagon(){};
    public void Hexagen(double a,double b,double c, double d,double
e,double f)
    {
        r1=a;r2=b;r3=c;r4=d;r5=e;r6=f;
    }
    public double getCircum()
        {circum=r1+r2+r3+r4+r5+r6;return circum;}
    public double getArea()
        {area=47;return area;}
}
class rHexagon extends Hexagon
{
    double r;
    public rHexagon(){};
    public void rHexagen(double r)
    {
        r1=r2=r3=r4=r5=r6=r;
        this.r=r;
    }
    public double getCircum()
```

```
        {circum=6*r;return circum;}
    public double getArea()
        {area=0.5*3*Math.sqrt(3)*r*r;return area;}
}
public class Experiment09_1 {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner in = new Scanner(System.in);
        Shape[] s=new Shape[100];
        s[0]=new Ellipse(3,2);
        s[1]=new Square(3);
        s[2]=new Rectangle(2,3);
        s[3]=new Circle(2);
        for(int i=0;i<=3;i++)
        {
            s[i].getCircum();
            s[i].getArea();
        }
        System.out.println("The areas and circums:");
        for(int i=0;i<=3;i++)
        {
            System.out.printf("%.2f\t%.2f\n",s[i].area,s[i].circum);
        }
        System.out.println();
        Arrays.sort(s,0,4);
        System.out.println("Sort areas:");
        for(int i=0;i<=3;i++)
        {
            System.out.printf("%.2f\n",s[i].area);
            s[i].changecomtype();
        }
        System.out.println();
        Arrays.sort(s,0,4);
        System.out.println("Sort circums:");
        for(int i=0;i<=3;i++)
        {
            System.out.printf("%.2f\n",s[i].circum);
        }
    }
}
```

实验 10 链表操作

a)实验目的

（1） 熟悉链表的原理和基本算法。

（2） 加深对引用的理解和使用。

b)实验内容：

阅读课本 Magazine Collection 例子

在 MagazineList 类中添加方法，

void sort()//按照书名对链表排序

void mergeSort(MagazineList another)//检查两个链表是否都为排序链表，如果是，归并；如果不是，直接返回。

c)完成报告

```
package Experiment10;

public class MagazineList {
    private MagazineNode list;
    public MagazineList()
    {
        list=null;
    }
    public void add(Magazine mag)
    {
        MagazineNode node=new MagazineNode(mag);
        MagazineNode current;
        if(list==null)
            list=node;
        else
        {
            current=list;
            while(current.next!=null)
                current=current.next;
            current.next=node;
        }
    }

    void sort()
    {
```

```
        MagazineNode current=list;
        while(current.next!=null)
        {

    while(current.next!=null&&current.magazine.toString().compareTo(current
.next.magazine.toString())<=0)
                    current=current.next;
            if(current.next==null) break;
            MagazineNode insert=current.next;
            if(current.next.next!=null)
                current.next=insert.next;
            else
                current.next=null;
            MagazineNode pointer=list;

    if(pointer.magazine.toString().compareTo(insert.magazine.toString())>=0
)
            {
                insert.next=pointer;
                list=insert;
                current=list;
            }
            else
            {

    while(pointer.next.magazine.toString().compareTo(insert.magazine.toStri
ng())<=0)
                    pointer=pointer.next;
                insert.next=pointer.next;
                pointer.next=insert;
            }
//          System.out.println(this);
        }
    }

    void mergeSort(MagazineList another)
    {
        MagazineNode current=list,current2=another.list;
        while(current.next!=null)
        {

    if(current.magazine.toString().compareTo(current.next.magazine.toString
())>0)
                return;
```

```
            current=current.next;
        }
        while(current2.next!=null)
        {

    if(current2.magazine.toString().compareTo(current2.next.magazine.toStri
ng())>0)
                return;
            current2=current2.next;
        }
        current=list;
        current2=another.list;
        MagazineList result=new MagazineList();
        while(current!=null||current2!=null)
        {
            if(current==null||current2==null)
            {
                while(current!=null)
                {
                    result.add(current.magazine);
                    current=current.next;
                }
                while(current2!=null)
                {
                    result.add(current2.magazine);
                    current2=current2.next;
                }
            }
            else
if(current.magazine.toString().compareTo(current2.magazine.toString())<=0)
            {
                result.add(current.magazine);
                current=current.next;
            }
            else
            {
                result.add(current2.magazine);
                current2=current2.next;
            }
        }
        this.list=result.list;
    }

    void Reverse()
```

```
    {
        MagazineNode current=list,temp;
        list=null;
        while(current!=null)
        {
            temp=current.next;
            current.next=list;
            list=current;
            current=temp;



        }
    }

    public String toString()
    {
        String result="";
        MagazineNode current=list;
        while(current!=null)
        {
            result+=current.magazine + "\n";
            current=current.next;
        }
        return result;
    }
}
```