

实验

3

数组和方法应用

3.1 核心知识

- (1) 数组是一组相关类型的变量集合,并且这些变量可以按照统一的方式进行操作。
- (2) 数组的下标从 0 开始,数组长度大小减 1 结束。超出下标范围,程序会抛出 `ArrayIndexOutOfBoundsException` 异常。
- (3) 方法就是定义在类中的具有特定功能的一段独立小程序,用来完成某个功能操作。

3.2 实验目的

- (1) 理解数组的含义,掌握数组的创建和使用方法。
- (2) 掌握数组的遍历方式,以及如何避免产生 `ArrayIndexOutOfBoundsException` 异常。
- (3) 掌握方法的定义与使用,能够抽取方法来增强代码的可读性和扩展。



3.3 实验导例

用户从键盘输入一个数字,创建指定容量大小的数组,再依次输入相应个数的半径值,最后输出每个半径对应圆的面积。

编写程序 `ArrayCircle.java`。使用 `Scanner` 接收用户输入的数字,创建指定容量大小的数组。使用 `for` 循环相应次数,依次让用户输入半径,并将值存入该数组中。遍历数组,计算出每一个圆的面积。程序运行结果如图 3-1 所示。

```
请输入要计算的个数: 3
请输入第1个半径: 2
请输入第2个半径: 3
请输入第3个半径: 4
输入完毕, 开始计算.....
圆的半径是2.0, 面积计算为12.56
圆的半径是3.0, 面积计算为28.26
圆的半径是4.0, 面积计算为50.24
```

图 3-1 计算圆的面积

参考代码清单 ArrayCircle.java

```
package exper3;
import java.util.Scanner;
public class ArrayCircle{
    static Scanner scanner=new Scanner(System.in);
    public static void main(String[] args){
        System.out.print("请输入要计算面积的圆的个数: ");
        int count=scanner.nextInt();
        double[] arr=getArr(count);
        System.out.println("输入完毕,开始计算.....");
        print(arr);
    }
    public static void print(double[] arr) {
        for (int i=0; i<arr.length; i++) {
            System.out.println("圆的半径是 "+arr[i] +", 面积计算为" +arr[i] *
                arr[i] * 3.14);
        }
    }
    public static double[] getArr(int count) {
        double[] arr=new double[count];
        for (int i=0; i<count; i++){
            System.out.print("请输入第" + (i +1) + "个半径: ");
            double r=scanner.nextDouble();
            arr[i]=r;
        }
        return arr;
    }
}
```

3.4

实验内容

(1) 编写一个程序,接收用户输入的 10 个数字,按照从小到大的顺序采用冒泡排序法对数组进行排序。此处排序算法可更改为其他排序算法。

① 编写程序 Sort.java。创建一个长度为 10 的 int 类型数组,通过键盘接收用户输入的 10 个数字。遍历数组,采用冒泡排序法对数组进行排序,并输出排序后的数组。





② 程序运行结果如图 3-2 所示。

(2) 编写一个方法 reverse(), 该方法的作用是将数组中的元素进行翻转。

① 编写程序 ReverseArr.java。编写一个方法 reverse(), 接收参数为数组。使用临时数组将数组中的元素翻转, 并返回新的数组和输出翻转后的数组。

② 程序运行结果如图 3-3 所示。

```

请输入10个数字, 用空格隔开
3 5 6 8 9 4 1 2 10 7
1 2 3 4 5 6 7 8 9 10
  
```

图 3-2 数组排序

```

数组翻转前:
1 2 3 4 5 6
数组翻转后:
6 5 4 3 2 1
  
```

图 3-3 数组翻转

3.5 参考代码

参考代码清单 Sort.java

```

package exper3;
import java.util.Scanner;
public class Sort {
    public static void main(String[] args) {
        int[] arr=buildArray();
        bubbleSort(arr);
        print(arr);
    }
    public static int[] buildArray(){
        System.out.println("请输入 10 个数字,用空格隔开");
        Scanner scanner=new Scanner(System.in);
        int[] arr=new int[10];
        for (int i=0; i<10; i++) {arr[i]=scanner.nextInt();}
        return arr;
    }
    public static void bubbleSort(int[] arr) {
        for (int i=0; i<arr.length -1; i++) {
            for (int j=0; j<arr.length -1 -i; j++) {
                if (arr[j]>arr[j +1]) {
                    int temp=arr[j];
                    arr[j]=arr[j +1];
                    arr[j +1]=temp;
                }
            }
        }
    }
    public static void print(int[] arr) {
        for (int i=0; i<arr.length; i++){
            System.out.print(arr[i] + " ");
        }
    }
}
  
```

参考代码清单 ReverseArray.java

```
package exper3;
public class ReverseArray {
    public static void main(String[] args) {
        int[] arr={1,2,3,4,5,6};
        System.out.println("数组 翻转前: ");
        print(arr);
        int[] reverse=reverse(arr);
        System.out.println();
        System.out.println("数组 翻转后: ");
        print(reverse);
    }
    public static void print(int[] arr) {
        for (int i=0; i<arr.length; i++){
            System.out.print(arr[i] + " ");
        }
    }
    public static int[] reverse(int[] arr) {
        int[] tmp=new int[arr.length];
        for (int i=0; i<arr.length; i++){
            tmp[i]=arr[arr.length -1 -i];
        }
        return tmp;
    }
}
```

3.6

扩展知识



事实上,for 循环还存在另一种写法,称为“增强 for 循环”,表示为 foreach,其语法如下:

```
for(数据类型 变量名: 数组) {}
```

这种 foreach 写法等价于普通的 for 循环,但是却可以避免操作不当带来的数组下标越界问题,因此增强 for 循环在实际开发中使用较为广泛。实验导例中参考代码清单 ArrayCircle.java 可以修改为以下写法。

参考代码清单 ArrayCircleEnhance.java

```
package exper3;
import java.util.Scanner;
public class ArrayCircleEnhance {
    static Scanner scanner=new Scanner(System.in);
    public static void main(String[] args){
        System.out.print("请输入要计算的个数: ");
        int count=scanner.nextInt();
        double[] arr=getArray(count);
    }
}
```

```
        System.out.println("输入完毕,开始计算……");
        print(arr);
    }
    public static void print(double[] arr) {
        for (double item: arr) { //增强 for 循环
            System.out.println("圆的半径是" + item + ",面积计算为" + item * item * 3.14);
        }
    }
    public static double[] getArray(int count) {
        double[] arr=new double[count];
        for (int i=0; i<count; i++){
            System.out.print("请输入第" + (i +1) + "个半径: ");
            double r=scanner.nextDouble();
            arr[i]=r;
        }
        return arr;
    }
}
```