中北大学软件学院

**实验报告**

|  |  |
| --- | --- |
| 专 业： | 软件工程 |
| 课程名称： | 面向对象程序设计 |
| 班 级： | 22130403 |
| 学 号： | 2213040340 |
| 姓 名： | 张恒瑜 |
| 辅导教师： | 李华玲 |

2021年9月制

成绩：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验时间 | 2023年 9 月 24 日 21 时至 23时 | 学时数 | 4学时 | |
| 1.实验名称  基本数据类型和条件分支语句 | | | | |
| 2.实验目的  （1）熟悉Java的基本数据类型的定义和使用 ；  （2）掌握从控制台输入数据和输出数据的方法；  （3）学会条件流程控制语句的使用方法；  （4）初步学会用面向对象的思维解决简单的问题； | | | | |
| 3.训练知识点集群  Java基本数据类型、输入数据和输出数据的方法、条件流程控制语句 | | | | |
| 4.实验内容  （1）闰年计算：根据输入的年份，判断该年份是否为闰年。  （2）水仙花数：按用户输入的数值，判断数值是否为“水仙花数”。  （3）课程融合之“三元组”：  设计实现抽象数据类型“三元组 （Triplet）” 。每个三元组由任意三个实数的序列构成，基本操作包括：创建一个三元组，取三元组的任意一个分量，置三元组的任意一个分量，求三元组的最大分量、最小分量，显示三元组，销毁三元组等。  要求：用类封装需要定义的数据类型Triplet，完成所有基本操作的Java语言实现与调用，并写测试程序。   1. 小组自主创作实验：围绕实验目的，题目自拟。 | | | | |
| 5.实验源代码  (1)闰年计算   1. 水仙花数 2. 三元组   （1）//闰年（leapyear）判断：year % 4 == 0 && year % 100 ！= 0  import java.util.Scanner;  class Leapyear {  int year;  public void JudgeLeapyear () {  Scanner sc = new Scanner(System.in);  System.out.println("请输入一个年份：");  year = sc.nextInt();  if(year % 4 == 0 && year % 100 != 0) {  System.out.println("是的朋友，这是一个闰年！");  }  else {  System.out.println("oh抱歉我的上帝，这不是一个闰年！");  }  }  }  public class TestLeapyear {  public static void main (String[] args) {  Leapyear yea = new Leapyear();  yea.JudgeLeapyear();  }  }  （2）//水仙花是：abc = a^3 + b^3 + c^3  //水仙花： Daffodil  import java.util.Scanner;  class Daffodil {  int number;  public void JudgeDaffodilNum () {  Scanner sc = new Scanner(System.in);  System.out.println("请输入一个三位数：");  number = sc.nextInt();  if(number == Math.pow(number%10 , 3) + Math.pow((number%100)/10 , 3) + Math.pow(number/100 , 3)) {  System.out.println("这是一个水仙花数！");  }  else {  System.out.println("抱歉，这不是一个水仙花数！");  }  }  }  public class TestDaffodil {  public static void main (String[] args) {    Daffodil da = new Daffodil();  da.JudgeDaffodilNum();    }  }  （3）  import java.util.Scanner;  class Triplet {  float e1;  float e2;  float e3;  public void initTriplet() {  Scanner sc = new Scanner(System.in);  System.out.println("请输入三元组的三个数:");  e1 = sc.nextFloat();  e2 = sc.nextFloat();  e3 = sc.nextFloat();    }  public void SearchTriplet() {  Scanner sc1 = new Scanner(System.in);  System.out.println("请输入查找的三元组中的第几个数：");  int num = sc1.nextInt();  if(num == 1) System.out.println(e1);  if(num == 2) System.out.println(e2);  if(num == 3) System.out.println(e3);  }  public void ChangeTriplet() {  Scanner sc2 = new Scanner(System.in);  System.out.println("你要修改三元组中第几个数：");  int num = sc2.nextInt();  if(num == 1){  e1 = sc2.nextFloat();  System.out.println("修改后的数为：" + e1);  }  if(num == 2){  e2 = sc2.nextFloat();  System.out.println("修改后的数为：" + e2);  }  if(num == 3){  e3 = sc2.nextFloat();  System.out.println("修改后的数为：" + e3);  }  }  public float MaxTriplet() {  float max = e1;  if(e2 > e1) {  max = e2;  }  if(e3 > e2) {  max = e3;  }  return max;  }  public float MinTriplet() {  float min = e1;  if(e2 < e1) min = e2;  if(e3 < e2) min = e3;  return min;  }  public void ShowTriplet() {  System.out.println("当前三元组是:" + "[" + e1 + "," + e2 + "," + e3 +c "]") ;  }  public void DeleteTriplet() {  }  }  public class TestTriplet {  public static void main(String[] args) {  Scanner sc3 = new Scanner(System.in);  Triplet tri = new Triplet();  int choose;  while (true) {  System.out.println("\*\*\*\*\*\*请输入你想进行的操作\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*1.创建一个三元组\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*2.查找三元组的数\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*3.修改三元组的数\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*4.找三元组最大值\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*5.找三元组最小值\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*6.列目前的三元组\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*7.删除目前三元组\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*\*\*\*\*8.退出程序\*\*\*\*\*\*\*\*\*\*\*");  System.out.println("\*\*\*\*\*\*\*请输入你想进行的操作\*\*\*\*\*\*\*");  choose = sc3.nextInt();    switch (choose) {  case 1 -> {  tri.initTriplet();  break;  }  case 2 -> {  tri.SearchTriplet();  break;  }  case 3 -> {  tri.ChangeTriplet();  break;  }  case 4 -> {  tri.MaxTriplet();  System.out.println("三元组的最大值是：" + tri.MaxTriplet());  break;  }  case 5 -> {  tri.MinTriplet();  System.out.println("三元组的最小值是：" + tri.MinTriplet());  break;  }  case 6 -> {  tri.ShowTriplet();  break;  }  case 7 -> {  tri.DeleteTriplet();  break;  }  case 8 -> {  return;  }  default -> {  System.out.println("输入错误请重新输入！");  break;  }  }  }  }  }   1. 实验结论及心得   **在C语言的基础上，做这些就显得游刃有余，只需要注意java中一些独特的语句类型表示和函数类型表示即可，再接再厉！** | | | |
|  | | | |