**《专高三第18单元》练习手册**

编写人员：

编写时间：

专高三\_分布式架构\_第18单元

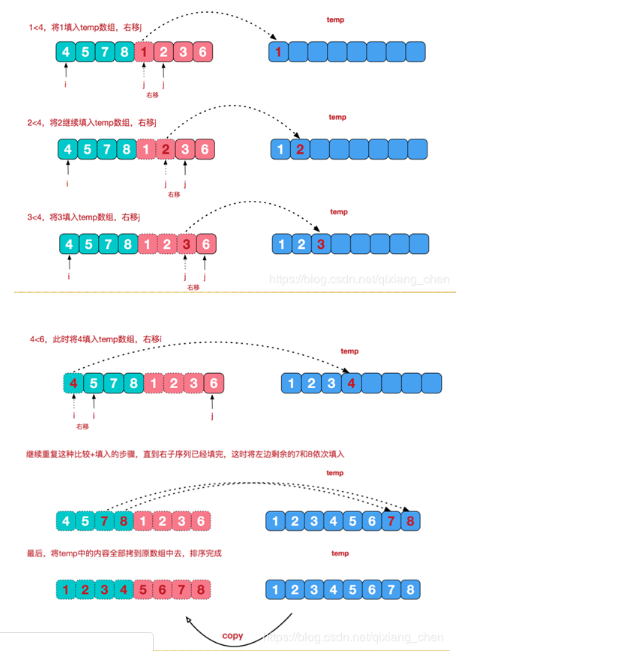
练习手册

一、判断题

1. (判断)Java语言是编译类型语言A. 正确B. 错误2. (判断)同一个JVM中同一个的类可以被加载多次A. 正确B. 错误3. (判断)long类型8个字节，float类型4个字节，long类型中最大数比float类型中最大的数要更大A. 正确B. 错误4. (判断)Java语言无法调用C语言编写的DLL动态链接库A. 正确B. 错误5. (判断)在Java语言中字符'A'占一个字节,汉字'中'占两个字节A. 正确B. 错误6. (判断)在Java语言中int和char类型可以相互转换A. 正确B. 错误7. (判断)在Java语言中可以集成父类的方法和属性，我们可以写一个类继承java.lang.StringA. 正确B. 错误8. (判断)Java语言中，有八种基本数据类型A. 正确B. 错误9. (判断)Java可以开发桌面程序，也可以开发服务器端程序A. 正确B. 错误10. (判断)Java类中启动方法可以是public static void main(String… arg)A. 正确B. 错误二、单选题1. (单选)下面代码Java变量定义正确的是 35. String #name="Jane Doe" 36. int %age = 24; 37. Double \_height = 123.5; 38. double ~temp = 37.5;A. 第35行B. 第36行C. 第37行D. 第38行2. (单选)public class TestString{ public static void main(String[] args) { String str = "420"; str += 42; System.out.println(str); } } 输出结果是：A. 42B. 420C. 462D. 420423. (单选)public class Test { public static void main(String[] args) { int x = 5; boolean b1 = true; boolean b2 = false; if((x==4) && !b2) System.out.println("1 "); System.out.println("2 "); if((b2=true) && b1) System.out.println("3 "); } } 输出结果是：A. 2B. 3C. 1 2D. 2 34. (单选)public void go(){ String o = ""; z: for(int x=0;i<3;x++) { for(int y=0;y<2;y++) { if(x == 1) break; if(x==2 && y==1) break z; o = o + x + y; } } System.out.println(o); } 输出结果是：A. 00B. 0001C. 000120D. 000120215. (单选)int x = 0; int y = 10; do{ y--; ++x; }while(x<5); System.out.println(x+","+y); 输出结果是：A. 5,6B. 5,5C. 6,5D. 6,66. (单选)public class Breaker { static String o = ""; public static void main(String[] args) { z: o = o + 2; for(int x=3;x<8;x++) { if(x == 4) break; if(x == 6) break z; o = o + x; } System.out.println(o); } } 输出结果是：A. 23B. 234C. 234567D. 编译错误7. (单选)public class Breaker2 { static String o = ""; public static void main(String[] args) { z: for(int x=2;x<7;x++) { if(x == 3) continue; if(x == 5) break z; o = o + x; } System.out.println(o); } } 输出结果是：A. 2B. 24C. 234D. 编译错误8. (单选)public class Spock { public static void main(String[] args) { Long tail = 2000L; Long distance = 1999L; Long story = 1000L; if((tail>distance)^((story\*2)==tail)) System.out.println("1"); if((distance+1 != tail)^((story\*2)==distance)) System.out.println("2"); } } 输出结果是：A. 1B. 2C. 12D. 无输出9. (单选)String[] elements = {"for","tea","too"}; String first = (elements.length>0)?elements[0]:null; 输出结果是：A. 编译错误B. 运行时抛出异常C. first赋值为nullD. first赋值为elements[0]10. (单选)import java.util.\*; public class Quest { public static void main(String[] args) { String[] colors={"blue","red","green","yellow","orange"}; Arrays.sort(colors); int s2 = Arrays.binarySearch(colors,"orange"); int s3 = Arrays.binarySearch(colors,"violet"); System.out.println(s2+" "+s3); } } 输出结果是：A. 2-1B. 2-4C. 2-5D. 编译错误11. (单选)class Alligator { public static void main(String[] args) { int[] x[] = {{1,2},{3,4,5},{6,7,8,9}}; int[][] y = x; System.out.println(y[2][1]); } } 输出结果是：A. 2B. 3C. 7D. 编译错误12. (单选)public class Barn { public static void main(String[] args) { new Barn().go("hi",1); new Barn().go("hi","world",2); } public void go(String... y,int x){ System.out.println(y[y.length-1]+" "); } } 输出结果是：A. hi hiB. hi worldC. world worldD. 编译错误13. (单选)class Person { String name = "No name"; public Person(String nm){name = nm;} } class Employee extends Person { String empId = "0000"; public Employee(String id){empId = id;} } class EmployeeTest { public static void main(String[] args) { Employee e = new Employee("4321"); System.out.println(e.empId); } } 输出结果是：A. 4321B. 0000C. 运行时抛出异常D. 编译错误14. (单选)class Mud { //将下面代码放到这里, System.out.println("hi"); } } 一共有多少选项可以正常编译 public static void main(String...a){ public static void main(String.\* a){ public static void main(String... a){ public static void main(String[]... a){ public static void main(String...[] a){A. 0B. 1C. 2D. 315. (单选)class Atom { Atom(){System.out.print("atom ");} } class Rock extends Atom { Rock(String type){System.out.print(type);} } public class Mountain extends Rock { Mountain(){ super("granite "); new Rock("granite "); } public static void main(String[] a){new Mountain();} } 输出结果是：A. 编译错误B. atom graniteC. granite graniteD. atom granite atom granite16. (单选)public class Blip { protected int blipvert(int x){return 0;} } class Vert extends Blip { //下面代码放在这里 } 1. public int blipvert(int x){return 0;} 2. private int blipvert(int x){return 0;} 3. private int blipvert(long x){return 0;} 4. protected long blipvert(int x){return 0;} 5. protected int blipvert(long x){return 0;} 6. protected long blipvert(long x){return 0;} 7. protected long blipvert(int x,int y){return 0;} 一共有多少选项放在上面可以正常编译A. 0B. 2C. 4D. 517. (单选)1.class Super 2.{ 3. private int a; 4. protected Super(int a){this.a = a;} 5.} 6.class Sub extends Super 7.{ 8. public Sub(int a){super(a);} 9. public Sub(){this.a = 5;} 10.} 上面代码如下修改可以编译通过A. 修改第2行为public int a;B. 修改第2行为protected int a;C. 修改第13行为public Sub(){this(5);}D. 修改第13行为public Sub(){super(a);}18. (单选)class Batman { int squares = 81; public static void main(String[] args){ new Batman().go(); } void go(){ incr(++squares); System.out.println(squares); } void incr(int squares){squares+=10;} } 输出结果是：A. 81B. 82C. 91D. 9219. (单选)class Pass { public static void main(String[] args) { int x = 5; Pass p = new Pass(); p.doStuff(x); System.out.println(" main x = "+x); } void doStuff(int x){ System.out.println(" doStuff x = "+x++); } } 输出结果是：A. 编译错误B. 运行时抛出异常C. doStuff x=6 main x=6D. doStuff x=5 main x=520. (单选)class ClassA { public int numberOfInstances; protected ClassA(int numberOfInstances){ this.numberOfInstances=numberOfInstances; } } class ExtendedA extends ClassA { private ExtendedA(int numberOfInstances) { super(numberOfInstances); } public static void main(String[] args){ ExtendedA ext = new ExtendedA(420); System.out.println(ext.numberOfInstances); } } 输出结果是：A. 420B. 运行时抛出异常C. 所有构造方法必须声明为publicD. 构造方法不能声明为私有21. (单选)class One { void foo(){} } class Two extends One { //下面代码放在这里 } 一共有多少选项可以放在上面可以正常编译 1.int foo(){} 2.void foo(){} 3.public void foo(){} 4.private void foo(){} 5.protected void foo(){}A. 0B. 1C. 3D. 222. (单选)class X { X(){System.out.print(1);} X(int x){ this(); System.out.print(2); } } public class Y extends X { Y(){ super(6); System.out.print(3); } Y(int y){ this(); System.out.println(4); } public static void main(String[] a){new Y(5);} } 输出结果是：A. 13B. 134C. 1234D. 2134三、多选题1. (多选)public class Hi { void m1(){} protected void m2(){} } class Lois extends Hi { //　下面哪些代码可以放在这里 } 输出结果是：A. public void m1(){}B. protected void m1(){}C. private void m1(){}D. void m2(){}2. (多选)class One { public One foo(){return this;} } class Two extends One { public One foo(){return this;} } class Three extends Two { //下面哪些代码可以放在这里 } A.public void foo(){} B.public int foo(){return 3;} C.public Two foo(){return this;} D.public One foo(){return this;}A. public void foo(){}B. public int foo(){return 3;}C. public Two foo(){return this;}D. public One foo(){return this;}3. (多选)class Employee { String name;double baseSalary; public Employee(String name,double baseSalary){ this.name = name; this.baseSalary = baseSalary; } } public class SalesPerson extends Employee { double commission; public SalesPerson(String name, double baseSalary,double commission) { //下面哪些代码可以放在这里编译运行 } }A. super(name,baseSalary);B. this.commission = commission;C. super(); this.commission = commission;D. super(name,baseSalary); this.commission = commission;4. (多选)public abstract class Shape { private int x; private int y; public abstract void draw(); public void setAnchor(int x,int y) { this.x = x; this.y = y; } } 下面定义正确的是：A. public class Circle implements Shape { private int radius; }B. public abstract class Circle extends Shape { private int radius; }C. public class Circle extends Shape { private int radius; public void draw(); }D. public class Circle extends Shape { private int radius; public void draw(){}; }5. (多选)class ClassA{} class ClassB extends ClassA{} class ClassC extends ClassA{} ClassA p0 = new ClassA(); ClassB p1 = new ClassB(); ClassC p2 = new ClassC(); ClassA p3 = new ClassB(); ClassA p4 = new ClassC(); 下面哪几个赋值语句是合法的A. p0=p1B. p1=p2C. p1 = (ClassB)p3D. p2 = (ClassC)p46. (多选)class Thingy{Meter m = new Meter();} class Component{ void go(){System.out.print("c");}} class meter extends Component { void go(){System.out.print("m");} } class DeluxeThingy extends Thingy { public static void main(String[] args) { DeluxeThingy dt = new DeluxeThingy(); dt.m.go(); Thingy t = new DeluxeThingy(); t.m.go(); } } 下面描述正确的是：A. 输出mmB. 输出mcC. DeluxeThingy has-a ComponentD. 以上都不对7. (多选)abstract public class Employee { protected abstract double getSaleAmount(); public double getCommision() { return getSalesAmount()\*0.15; } } class Sales extends Employee { //下面哪些代码放在这里可以正确运行 }A. double getSalesAmount(){return 1230.45;}B. public double getSalesAmount(){return 1230.45;}C. private double getSalesAmount(){return 1230.45;}D. protected double getSalesAmount(){return 1230.45;}8. (多选)public interface Status { /\*下面哪些修饰符可以放在这里\*/ int MY\_VALUE=10; }A. finalB. staticC. nativeD. public9. (多选)下面哪些类正确实现java.lang.Runnable和java.lang.Cloneable接口A. public class Session implements Runnable,Cloneable { public void run(); public Object clone(); }B. public class Session extends Runnable,Cloneable { public void run(){...} public Object clone(){...} }C. public class Session implements Runnable,Cloneable { public void run(){...} public Object clone(){...} }D. public abstract class Session implements Runnable,Cloneable { public void run(){...} public Object clone(){...} }10. (多选)10.interface Foo{ 11. int bar(); 12.} 13. 14.public class Beta{ 15. 16. class A implements Foo{ 17. public int bar(){return 1;} 18. } 19. 20. public int fubar(Foo foo){return foo.bar();} 21. 22. public void testFoo(){ 23. 24. Class A implements Foo{ 25. public int bar(){return 2;} 26. } 27. 28. System.out.println(fubar(new A())); 29. } 30. 31. public static void main(String[] argv){ 32. new Beta().testFoo(); 33. } 34.} 下面哪些语句是正确的A. 编译错误B. 代码可以编译，运行输出2C. 如果第16,17,18行删除，输出2D. 如果第24,25,26行删除，输出1

四、技能题

1. **案例题目:** 使用Java实现归并排序算法



**将两个排序好的数组，合并排序，参考上图实现算法**