**[【计算机二级Java语言】卷018](https://www.cnblogs.com/mjios/p/12469864.html)**

**选择题**

**公共知识**

【1】下列叙述中正确的是

〖A〗算法的复杂度包括时间复杂度与空间复杂度

〖B〗算法的复杂度是指算法控制结构的复杂程度

〖C〗算法的复杂度是指算法程序中指令的数量

〖D〗算法的复杂度是指算法所处理的数据量

算法复杂度是指算法在编写成可执行程序后, 运行时所需要的资源, 资源包括时间资源和内存资源。  
算法的复杂度包括时间复杂度与空间复杂度。  
算法的时间复杂度是指执行算法所需要的计算工作量; 算法的空间复杂度是指算法在执行过程中所需要的内存空间。

【2】设数据集合为D = { 1, 2, 3, 4, 5 } 。下列数据结构 B = (D, R)中为非线性结构的是

〖A〗R = { (2, 5), (5, 4), (3, 1), (4, 3) }

〖B〗R = { (1, 2), (2, 3), (3, 4), (4, 5) }

〖C〗R = { (1, 2), (2, 3), (4, 3), (3, 5) }

〖D〗R = { (5, 4), (4, 3), (3, 2), (2, 1) }

A项中, R = { (2, 5), (5, 4), (3, 1), (4, 3) }, 2为根结点, 元素顺序为2→5→4→3→1, 属于线性结构; 同理B项1为根结点, 元素顺序为1→2→3→4→5, D项5为跟结点, 元素顺序为5→4→3→2→1, 均为线性结构。  
C项中, 元素3有两个前件, 属于非线性结构。

【3】设栈的存储空间为S(1 : m), 初始状态为top = m + 1。经过一系列入栈与退栈操作后, top = m。现又在栈中退出一个元素后, 栈顶指针top值为

〖A〗0

〖B〗m - 1

〖C〗m + 1

〖D〗产生栈空错误

栈的顺序存储空间为S(1 : m), 初始状态top = m + 1, 所以这个栈是m在栈底, 1是开口向上的。  
经过一系列入栈与退栈操作后top = m, 则栈中有1个元素, 若现在又退出一个元素, 那么栈顶指针下移一位, 回到m + 1的位置。

【4】下列叙述中正确的是

〖A〗结点中具有两个指针域的链表一定是二叉链表

〖B〗结点中具有两个指针域的链表可以是线性结构, 也可以是非线性结构

〖C〗循环链表是循环队列的链式存储结构

〖D〗循环链表是非线性结构

结点中具有两个指针域的链表既可以是双向链表也可以是二叉链表, 双向链表是线性结构, 二叉链表属于非线性结构。  
循环链表是线性链表的一种形式, 属于线性结构, 采用链式存储结构, 而循环队列是队列的一种顺序存储结构。

【5】在长度为n的顺序表中查找一个元素, 假设需要查找的元素一定在表中, 并且元素出现在表中每个位置上的可能性是相同的, 则在平均情况下需要比较的次数为

〖A〗n / 4

〖B〗n

〖C〗3n / 4

〖D〗(n + 1) / 2

在顺序表中查找, 最好情况下第一个元素就是要查找的元素, 则比较次数为1; 在最坏情况下, 最后一个元素才是要找的元素, 则比较次数为n。  
则平均比较次数：(1 + 2+┉+n) / n = (n(n + 1) / 2) / n = (n + 1) / 2。

【6】下面属于系统软件的是

〖A〗UNIX系统

〖B〗ERP系统

〖C〗办公自动化系统

〖D〗学生成绩管理系统

软件按功能可以分为应用软件、系统软件和支撑软件(或工具软件)。  
系统软件是管理计算机的资源, 提高计算机使用效率并服务于其他程序的软件, 如操作系统, 编译程序, 汇编程序, 数据库管理系统和网络软件等。  
UNIX系统属于操作系统, 故属于系统软件。  
ERP系统、办公自动化系统、学生成绩管理系统属于应用软件。

【7】下面对软件测试描述错误的是

〖A〗严格执行测试计划, 排除测试的随意性

〖B〗随机地选取测试数据

〖C〗软件测试的目的是发现错误

〖D〗软件测试是保证软件质量的重要手段

软件测试的准则如下：①所有测试都应追溯到需求。

②严格执行测试计划, 排除测试的随意性。

③充分注意测试中的群集现象。

④程序员应避免检查自己的程序。

⑤穷举测试不可能。

⑥妥善保存测试计划、测试用例、出错统计和最终分析报告, 为维护提供方便。

【8】在关系数据库中, 描述全局数据逻辑结构的是

〖A〗概念模式

〖B〗用户模式

〖C〗内模式

〖D〗物理模式

数据库系统在其内部分为三级模式及两级映射, 三级模式即概念模式、内模式和外模式。

概念模式是数据库系统中全局数据逻辑结构的描述, 是全体用户公共数据视图。

外模式也称子模式或者用户模式, 是用户的数据视图, 也就是用户所能够看见和使用的局部数据的逻辑结构和特征的描述, 是与某一应用有关的数据的逻辑表示。

内模式又称物理模式, 是数据物理结构和存储方式的描述, 是数据在数据库内部的表示方式。

【9】下面选项中不是关系数据库基本特征的是

〖A〗不同的列应有不同的列名

〖B〗不同的列应有不同的数据类型

〖C〗与行的次序无关

〖D〗与列的次序无关

关系数据库中的二维表一般满足7个性质：①元组(行)个数是有限的——元组个数有限性。

②元组(行)均不相同——元组的唯一性。

③元组(行)的次序可以任意交换——元组的次序无关性。

④元组(行)的分量是不可分割的基本特征——元组分量的原子性。

⑤属性(列)名各不相同——属性名唯一性。

⑥属性(列)与次序无关, 可以任意交换——属性的次序无关性。

⑦属性(列)的分量具有与该属性相同的值域——分量值域的统一性。

【10】关系模型中的关系模式至少应是

〖A〗1NF

〖B〗2NF

〖C〗3NF

〖D〗BCNF

关系数据库中的关系是要满足一定要求的, 满足不同程度要求的为不同的范式。  
满足最低要求的叫第一范式, 简称1NF。  
在满足第一范式的基础上, 进一步满足更多要求规范则是第二范式。  
然后在满足第二范式的基础上, 还可以再满足第三范式, 以此类推。

对于关系模式, 若其中的每个属性都已不能再分为简单项, 则它属于第一范式。

若某个关系R为第一范式, 并且R中每一个非主属性完全依赖于R的某个候选键, 则称其为第二范式。  
第二范式消除了非主属性对主键的部分依赖。

如果关系R是第二范式, 并且每个非主属性都不传递依赖于R的候选键, 则称R为第三范式。  
(传递依赖：在关系模式中, 如果Y→X, X→A, 且X不决定Y和A不属于X, 那么Y→A是传递依赖。  
)比第三范式更高级的BCF范式, 它要求所有属性都不传递依赖于关系的任何候选键

**专业知识**

【11】下列对Java特性的描述中, 错误的是

〖A〗Java类的构造方法没有返回值, 因此必须声明为void

〖B〗Java通过接口实现多继承

〖C〗类是Java程序的基本单位

〖D〗动态性是Java语言的特性之一

java规定构造方法没有返回值, 也不能用void修饰, 选项A说法错误, 本题答案为A。

【12】下列方法的定义中, 符合Java命名约定的是

〖A〗void $someMethod() { }

〖B〗static int someMethod() { }

〖C〗String SomeMethod() { }

〖D〗float somemethod() { }

方法的命名遵循驼峰命名法及第一个单词应以小写字母作为开头, 后面的单词则用大写字母开头。  
选项B正确, 本题答案为B。

【13】下列选项中, 不属于位运算符的是

〖A〗>>

〖B〗<<

〖C〗^

〖D〗!

选项D是逻辑运算符, 不是位运算符。  
本题答案为D。

【14】下列代码段执行后, t的值为

复制代码

Java

|  |  |
| --- | --- |
|  | int a = 5, b = 6, c = 8, t = 0; |
|  | t = a > b ? (a > c ? a : c) : b; |

〖A〗0

〖B〗5

〖C〗6

〖D〗8

本题考查条件运算符的应用, 表达式a > c返回值为false, 那么结果为c的值, a > b的返回值也为false, 结果为b的值。  
选项C正确, 本题答案为C。

【15】使表达式 x <= 0 || x > 100的值为false的选项是

〖A〗x = -5

〖B〗x = 0

〖C〗x = 50

〖D〗x = 150

要使两边的表达式为false的只有当x大于0小于100的时候才能成立, 选项C正确, 本题答案为C.

【16】下列代码段执行后, f的值为

复制代码

Java

|  |  |
| --- | --- |
|  | int x = 8, f; |
|  | f = 0; |
|  | for (int i = 2; i < x; i++) { |
|  | if (x % i != 0) |
|  | continue; |
|  | f = f + i; |
|  | } |

〖A〗0

〖B〗6

〖C〗7

〖D〗21

Java中关键字continue、break和return 的区别：continue:跳出本次循环继续下一次循环

break:跳出循环体, 继续执行循环外的函数体

return :跳出整个函数体, 函数体后面的部分不再执行

本题的运行结果为6, 选项B正确, 本题答案为B。

【17】下列代码段执行后, a的值为

复制代码

Java

|  |  |
| --- | --- |
|  | int a = 2, b = 8, c = 5; |
|  | if (a < b) |
|  | a = b; |
|  | if (a < c) |
|  | a = c; |

〖A〗2

〖B〗8

〖C〗5

〖D〗0

表达式a < b返回值为true, 执行a = b表达式。  
分支结构中如果多个条件同时满足, 那么只执行最前面的表达式, 后面表达式不执行。  
选项B正确, 本题答案为B。

【18】下列代码段执行后, x, y的值分别是

复制代码

Java

|  |  |
| --- | --- |
|  | int x = 1, y = 5; |
|  | do { |
|  | x++; |
|  | y--; |
|  | } while (x <= y); |

〖A〗2 4

〖B〗4 2

〖C〗3 3

〖D〗4 3

o...while 循环是 while 循环的变体。  
在检查while()条件是否为真之前, 该循环首先会执行一次do { } 之内的语句, 然后在while()内检查条件是否为真, 如果条件为真的话, 就会重复do...while这个循环, 直至while()为假。

i++:先引用后增加++i:先增加后引用

i++:先在i所在的表达式中使用i的当前值, 后让i加1++i:让i先加1, 然后在i所在的表达式中使用i的新值

选项B正确, 本题答案为B。

【19】下列代码段执行后, n的值为

复制代码

Java

|  |  |
| --- | --- |
|  | int n = 0; |
|  | for (int a = 1; a <= 9; a++) |
|  | for (int b = 0; b <= 9; b++) |
|  | if ((a \* 10 + b) % 10 == 0) |
|  | n++; |

〖A〗9

〖B〗10

〖C〗90

〖D〗100

n++:先在n所在的表达式中使用n的当前值, 后让n加1, 当程序执行第10次的时候停止, n++先等于9后再自身加1, 选项A正确, 本题答案为A。

【20】下列代码段执行后的结果是

复制代码

Java

|  |  |
| --- | --- |
|  | int[] a = new int[5]; |
|  | a[0] = 1; |
|  | for (int i = 1; i < 5; i++) |
|  | a[i] = a[i - 1] \* 2; |
|  | for (int i = 0; i < 5; i++) |
|  | System.out.print(a[i++] + " "); |

〖A〗1 2 4 8 16

〖B〗1 4 16

〖C〗1 2 3 4 5

〖D〗1 2 3

第一个for循环通过下标将所有的值都添加到数组中去, 数组中的值为[1, 2, 4, 8, 16], 第二个for循环再运算过程中i每循环一次自身+2, 所以输出的是数组中下标为0, 2, 4的值, 选项B正确, 本题答案为B。

【21】一个类用于表示抽象概念而不能创建实例时, 该类称为

〖A〗具体类

〖B〗抽象类

〖C〗父类

〖D〗子类

普通类是一个完善的功能类, 可以直接产生实例化对象, 并且在普通类中可以包含有构造方法、普通方法、static方法、常量和变量等内容。  
而抽象类是指在普通类的结构里面增加抽象方法的组成部分。  
那么什么叫抽象方法呢?在所有的普通方法上面都会有一个"{}", 这个表示方法体, 有方法体的方法一定可以被对象直接使用。  
而抽象方法, 是指没有方法体的方法, 同时抽象方法还必须使用关键字abstract做修饰。  
而拥有抽象方法的类就是抽象类, 抽象类要使用abstract关键字声明, 选项B正确, 本题答案为B。

【22】内部类可直接访问

〖A〗同一个程序的变量

〖B〗同一个包的变量

〖C〗定义它的外部类的变量

〖D〗其他包的变量

内部类访问特点：1, 内部类可以直接访问外部类中的成员。  
2, 外部类要访问内部类, 必须建立内部类的对象。  
一般用于类的设计。  
分析事物时, 发现该事物描述中还有事物, 而且这个事物还在访问被描述事物的内容。  
这时就是还有的事物定义成内部类来描述。  
内部类能直接访问外部类中成员, 是因为内部类持有了外部类的引用, 即外部类名.this。  
访问内部类成员 则要显示写出内部类.this内部类也可以存放在局部位置上, 但是内部类在局部位置上只能访问局部中被final修饰的局部变量。  
选项C正确, 本题答案为C。

【23】下列程序段的运行结果是

复制代码

Java

|  |  |
| --- | --- |
|  | String str1 = "Good"; |
|  | String str2 = "morning"; |
|  | String str3 = str1.toLowerCase() + str2.toUpperCase(); |
|  | System.out.println(str3); |

〖A〗Goodmorning

〖B〗goodMORNING

〖C〗gOODmORNING

〖D〗goodmorning

toLowerCase()将字符串转小写, toUpperCase()转大写。  
选项B正确, 本题答案为B。

【24】下列程序的运行结果是

复制代码

Java

|  |  |
| --- | --- |
|  | public class Test { |
|  | static String s; |
|  | public static void main(String[] args) { |
|  | System.out.println(" s=" + s); |
|  | } |
|  | } |

〖A〗s = s

〖B〗s=

〖C〗s = null

〖D〗"s="

String类型的默认值为null, 选项C正确, 本题答案为C。

【25】下列程序段的运行结果是

复制代码

Java

|  |  |
| --- | --- |
|  | int[][] a = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 } }; |
|  | int[][] b = { { 2, 3, 4 }, { 5, 6, 7 }, { 8, 9, 10 } }; |
|  | int[][] c = new int[3][3]; |
|  | int i, j, k, dim = 3; |
|  | for (i = 0; i < dim; i++) |
|  | for (j = 0; j < dim; j++) { |
|  | c[i][j] = 0; |
|  | for (k = 0; k < dim; k++) |
|  | c[i][j] += a[i][k] \* b[k][j]; |
|  | } |
|  | System.out.print(c[0][0] + " " + c[2][2]); |

〖A〗36 174

〖B〗36 90

〖C〗2 174

〖D〗2 90

二维数组的每一个元素是一个一维数组定义格式数据类型[][]数组名=new 数据类型[二维数组的长度/包含的一维数组的个数][每个一维数组的长度]; int[][] arr = new int[3][5]; ---定义了一个整型的二维数组, 其中包含3个一维数组, 每个一维数组可以存储5个整数arr[0] ---下标为0的位置上的一维数组arr[1][3] ---如果要获取具体的元素需要两个下标数据类型[][]数组名=new 数据类型[二维数组的长度/包含的一维数组的个数][]; 选项A正确, 本题答案为A。

【26】当方法遇到异常又不知如何处理时, 正确的做法是

〖A〗捕获异常

〖B〗匹配异常

〖C〗声明抛出异常

〖D〗嵌套异常

当方法遇到异常又不知如何处理时需要声明抛出异常, 由谁调用谁去处理该异常, 选项C正确, 本题答案为C。

【27】执行下列程序段, 当输入-1后的输出是

复制代码

Java

|  |  |
| --- | --- |
|  | byte studentAge = 0; |
|  | Scanner scnr = new Scanner(System.in); |
|  | try { |
|  | System.out.println("Input student Age: "); |
|  | studentAge = scnr.nextByte(); |
|  | if (studentAge < 0) |
|  | throw new Exception("Not Positive"); |
|  | System.out.println("Student Age: " + studentAge); |
|  | } catch (InputMismatchException exc) { |
|  | System.out.println("Input mismatch"); |
|  | } catch (Exception exc) { |
|  | System.out.println("Error:" + exc.getMessage()); |
|  | } |

〖A〗Error : Not Positive

〖B〗Input mismatch

〖C〗Student Age : -1

〖D〗Student Age : -1 Input mismatch

年龄不符合大于0则直接抛出自定义异常, 该异常继承Exection异常, catch处理该异常执行Exection exc中的语句, 错误提示信息为Not Positive, 选项A正确, 本题答案为A。

【28】Collection接口中, 进行集合批量删除的方法是

〖A〗containsAll()

〖B〗add()

〖C〗removeAll()

〖D〗addAll()

A)boolean containsAll(Collectionc)如果此 collection 包含指定 collection 中的所有元素, 则返回 true。

B)boolean add(E e)确保此 collection 包含指定的元素(可选操作)。

C)boolean removeAll(Collection<?>c)移除此 collection 中那些也包含在指定 collection 中 的所有元素。

D)boolean addAll(Collection<?extends E > c)将指定 collection 中的所有元素都添加到此 collection 中。

本题答案为C。

【29】同时实现了DataInput和DataOutput这两个接口的类是

〖A〗DataStream

〖B〗RandomAccessFile

〖C〗ObjectStream

〖D〗DataFile

DataOutput接口实现类有：DataOutputStream, FileCacheImageOutputStream, FileImageOutputStream, ImageOutputStreamImpl, MemoryCacheImageOutputStream, ObjectOutputStream, RandomAccessFile DataInput接口实现类有：DataInputStream, FileCacheImageInputStream, FileCacheImageOutputStream, FileImageInputStream, FileImageOutputStream, ImageInputStreamImpl, ImageOutputStreamImpl, MemoryCacheImageInputStream, MemoryCacheImageOutputStream, ObjectInputStream, RandomAccessFile

选项B正确, 本题答案为B。

【30】下列选项中不是字符输出流类的是

〖A〗CharArrayWriter

〖B〗BufferedWriter

〖C〗FileWriter

〖D〗InputWriter

字符输出流基类：Writer、FileWriter、CharArrayWriter、PipedWriter、StringWriter、BufferedWriter、OutputStreamWriter、FilterWriter、PrintWriter。  
输入流没有Writer选项D说法错误, 本题答案为D。

【31】为得到文件f的路径名, 应使用的语句是

〖A〗f.getPath();

〖B〗f.getParent();

〖C〗f.getParentFile();

〖D〗f.list();

A)f.getPath(); 将此抽象路径名转换为一个路径名字符串。

B)f.getParent();  
返回此抽象路径名父目录的路径名字符串;  
如果此 路径名没有指定父目录, 则返回 null。

C)f.getParentFile();  
返回此抽象路径名父目录的抽象路径名;  
如果 此路径名没有指定父目录, 则返回 null D)f.list();  
返回一个字符串数组, 这些字符串指定此抽象路径名表 示的目录中的文件和目录。

本题答案为A。

【32】下列代码采用缓冲的方式实现了将一个文件的内容复制到另一个文件的功能。下划线处应填入的代码是

复制代码

Java

|  |  |
| --- | --- |
|  | import java.io.\*; |
|  | public class test { |
|  | public static void main(String[] args) { |
|  | try { |
|  | FileInputStream is = new FileInputStream("D:\\from.txt"); |
|  | FileOutputStream os = new FileOutputStream("D:\\to.txt"); |
|  | BufferedInputStream bi = new BufferedInputStream(is); |
|  | BufferedOutputStream bo = new \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(os); |
|  | byte[] data = new \_\_\_\_\_\_\_\_\_\_\_\_\_[1]; |
|  | while (bi.read(data) != -1) { |
|  | bo.write(data); |
|  | } |
|  | bo.flush(); |
|  | bi.close(); |
|  | bo.close(); |
|  | } catch (ArrayIndexOutOfBoundsException e) { |
|  | e.printStackTrace(); |
|  | } catch (IOException e) { |
|  | e.printStackTrace(); |
|  | } |
|  | } |
|  | } |

〖A〗BufferedOutputStream, byte

〖B〗BufferedInputStream, byte

〖C〗BufferedInputStream, char

〖D〗BufferedOutputStream, string

BufferedOutputStream 的实例化类为其本身, 复制文本是靠字节byte传输, 选项A正确, 本题答案为A。

【33】与AWT相比, Swing的按钮JButton

〖A〗只能显示文字

〖B〗只能显示图标

〖C〗可以同时显示文字和图标

〖D〗不能同时显示文字和图标

Button是AWT中组件, 此中组件信赖于本地操作系统的GUI, 缺乏平台独立性。  
其外观与本地操作系统有关。  
JButton是Swing中组件, 是用纯JAVA语言编写的, 不信赖于本地操作系统的GUI。  
Swing中组件可以跨平台运行。  
其外观与本地操作系统无关, 可以同时显示文字和图标。  
选项C正确, 本题答案为C。

【34】下列选项中默认布局管理器不是BorderLayout 的是

〖A〗Window

〖B〗Panel

〖C〗Frame

〖D〗Dialog

FlowLayout是Panel和Applet的默认布局管理器。  
构件在容器中的放置规律是从上到下、从左到右进行。  
BorderLayout是Window、Frame和Dialog默认布局管理器。  
BorderLayout布局管理器把容器分成5个区域：North、South、East、West和Center, 每个区域只能放置一个构件。  
选项B正确, 本题答案为B。

【35】有如下程序

复制代码

Java

<img border = "0" alt = "" src = "http://res.weilaijiaoyu.cn/CourseFile/JAVA/PC/Images/ja\_19\_3.png">

〖A〗JOptionPane.showMessageDialog(null, "今天是星期二吗", "请确认", JOptionPane.INFORMATION\_MESSAGE);

〖B〗JOptionPane.showMessageDialog(null, "请确认", "今天是星期二吗", JOptionPane.INFORMATION\_MESSAGE);

〖C〗JOptionPane.showMessageDialog(null, "今天是星期二吗", "请确认", JOptionPane.WARNING\_MESSAGE);

〖D〗JOptionPane.showMessageDialog(null, "请确认", "今天是星期二吗", JOptionPane.WARNING\_MESSAGE);

语法：JOptionPane.showMessageDialog(null, "在对话框内显示的描述性的文字", "标题条文字串", JOptionPane.INFORMATION\_MESSAGE); JOptionPane.WARNING\_MESSAGE使用这个在描述文字前是 图标 JOptionPane.INFORMATION\_MESSAGE才是对应图中图标

选项A正确, 本题答案为A。

【36】下列说法中正确的是

〖A〗线程中包括线程

〖B〗Java线程模型由代码和数据组成

〖C〗进程是轻型的线程

〖D〗线程必须在进程中运行

A 进程中包括线程; B Java中的线程模型包含3部分：一个虚拟的CPU, 该CPU执行的代码, 代码所操作的数据; C线程具有许多传统进程所具有的特征, 故又称为轻型线程或进程元; 而把传统的进程称为重型进程。  
选项D正确, 本题答案为D。

【37】为了使下列程序正常运行并且输出"1 2 3 4 5 ", 在下划线处应填入的是

复制代码

Java

|  |  |
| --- | --- |
|  | public class Test1 \_\_\_\_\_\_\_\_\_\_\_\_ { |
|  | private int x = 0; |
|  | boolean flag = true; |
|  | public static void main(String[] args) { |
|  | Test1 r = new Test1(); |
|  | Thread t1 = new Thread(r); |
|  | t1.\_\_\_\_\_\_\_(); |
|  | } |
|  | public void run() { |
|  | while (flag) { |
|  | x++; |
|  | System.out.print(x + " "); |
|  | if (x >= 5) |
|  | flag = false; |
|  | } |
|  | } |
|  | } |

〖A〗implements Runnable, start

〖B〗implements Runnable, run

〖C〗extends Thread, start

〖D〗extends Thread, run

实现Runnable, 线程代码存在接口的子类的run方法。

线程开始是调用start()方法, 选项A正确, 本题答案为A。

【38】用synchronized关键字保护的对象共享数据, 其修饰符必须是

〖A〗public

〖B〗default

〖C〗private

〖D〗protected

1． 定义private 的instance变量+它的 get方法, 而不要定义public / protected的instance变量。  
如果将变量定义为public, 对象在外界可以绕过同步方法的控制而直接取得它, 并改动它。  
这也是JavaBean的标准实现方式之一。  
2． 如果instance变量是一个对象, 如数组或ArrayList什么的, 那上述方法仍然不安全, 因为当外界对象通过get方法拿到这个instance对象的引用后, 又将其指向另一个对象, 那么这个private变量也就变了, 岂不是很危险。  
这个时候就需要将get方法也加上synchronized同步, 并且, 只返回这个private对象的clone()――这样, 调用端得到的就是对象副本的引用了。  
选项C正确, 本题答案为C。

【39】如果用户定义的Applet类中包含main()方法, 则下列陈述中正确的是

〖A〗该Applet程序将出现编译错误

〖B〗main()方法不能是静态的(static)

〖C〗该Applet程序可以通过编译, 但运行时将出错

〖D〗该Applet程序可以通过编译, 并且能够正常运行

本题考查Applet。  
Applet运行时, 首先由浏览器调用init()方法, 通知该Applet已经被加载到浏览器中, 使Applet执行一些基本的初始化。  
该方法不是必须的, 可以没有, Applet可以正常编译和运行。  
选项D正确, 本题答案为D。

【40】下列代码包括一个HTML文件和一个Applet程序。为了使HTML文件在浏览器中加载时显示字符串"Happy new year!", 在下划线处应填入的代码是

复制代码

Java

|  |  |
| --- | --- |
|  | <HTML><BODY><APPLET CODE = "AppletTest6.class" WIDTH = 150 HEIGHT = 25></APPLET></BODY></HTML > import java.awt.\*; |
|  | import \_\_\_\_\_\_\_\_\_\_.\*; |
|  | public class \_\_\_\_\_\_\_\_\_ extends JApplet { |
|  | JLabel label; |
|  | public void init() { |
|  | Container contentPane = getContentPane(); |
|  | label = new JLabel(); |
|  | contentPane.add(label); |
|  | label.setText(" Happy new year!"); |
|  | } |
|  | } |

〖A〗java.swing, AppletTest6

〖B〗javax.swing, AppletTest6

〖C〗java.awt.swing, NewYearApplet

〖D〗javax.swing, GreetingApplet

swing在javax包中, 由HTML里面可知是AppletTest6类 选项B正确, 本题答案为B。

**编程题**

[编程题代码下载](https://github.com/CoderMJLee/NCRE)

【41】在考生文件夹中存有文件名为Java\_1.java的文件, 该程序是不完整的, 请  
在注释行"//\*\*\*\*\*\*\*\*\*\*Found\*\*\*\*\*\*\*\*\*\*"下一行语句的下划线地方填入  
正确内容, 然后删除下划线, 请勿删除注释行或改动其他已有语句内容。存  
盘时文件必须存放在考生文件夹下, 不得改变原有文件的文件名。  
程序中定义了三个整型的一维数组a、b、c, 请将a数组中的对应项与b数组  
中的对应项相加后赋给数组c的对应项(例如：c[i] = a[i] + b[i]), 最后计  
算输出数组c中所有项之和。程序运行结果如下：  
s = 64

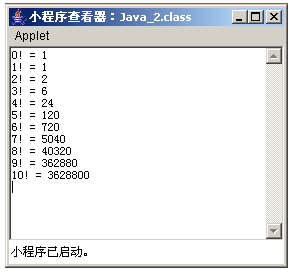
复制代码

Java

1.class 2.length 3.c[j] 4.out

本题考生对数组的掌握  
填空1:在java中创建类必须用关键字class  
填空2:获取数组的长度用length属性  
填空3:程序要求数组c中所有元素的和, 所以必须获取数组c中的每一项值  
填空4:输出语句为System.out.println()

【42】在考生文件夹中存有文件名为Java\_2.java 和Java\_2.html文件,  
Java\_2.java是不完整的, 请在注释行"//\*\*\*\*\*\*\*\*\*\*Found\*\*\*\*\*\*\*\*\*\*"  
下一行语句的下划线地方填入正确内容, 然后删除下划线, 请勿删除注释行或改动其他已有语句内容。  
存盘时文件必须存放在考生文件夹下, 不得改变原有文件的文件名。  
本题的要求是：在Applet中显示0至10的阶乘。请完善Java\_2.java文件, 并进行调试(在命令行中使用appletviewer Java\_2.html, 或者使用集成开发环境), 使运行结果为：



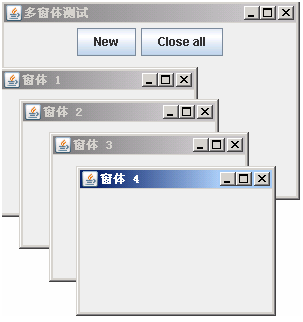
复制代码

Java

1.Applet 2.add 3.factorial(i)4.number

本题考查考生对Applet应用程序的运用  
填空1 : Applet是采用Java编程语言编写的小应用程序, 该程序可以包含在 HTML(标准通用标记语言的一个应用)页中, 与在页中包含图像的方式大致相同。  
本程序需要继承该类  
填空2:需要将TextArea组件添加到Applet容器的一端, add()方法提供此操作  
填空3:在输出语句中需要调用处理阶乘算法的方法, factorial()方法提供算法结果  
填空4:阶乘的定义为：n != 1×2×3×...×n。  
阶乘亦可以递归方式定义：0 != 1, n != (n - 1) !×n。

【43】在考生文件夹中存有文件名为Java\_3.java的文件, 该程序是不完整的, 请  
在注释行"//\*\*\*\*\*\*\*\*\*\*Found\*\*\*\*\*\*\*\*\*\*"下一行语句的下划线地方填入  
正确内容, 然后删除下划线, 请勿删除注释行或改动其他已有语句内容。存  
盘时文件必须存放在考生文件夹下, 不得改变原有文件的文件名。  
本题的要求是：  
程序中用Swing编写了一个窗体, 窗体内含有两个按钮"New"和"Close all"。  
当"New"按钮被按下后, 就生成一个新的窗体。当"Close all"按钮被按下  
后, 就关闭所有现在已经打开的窗体。程序运行如下图所示。



复制代码

Java

1.javax 2.extends 3.newButton 4.f 5.ActionListener

本题考查考生对窗口构件的应用  
填空1:所有swing构件中的类都是java扩展包中的类, java的扩展包为javax  
填空2 : JPanel 是 Java图形用户界面(GUI)工具包swing中的面板容器类, 包含在javax.swing 包中, 是一种轻量级容器, 可以加入到JFrame窗体中, 本类需要继承该类才能实现相关效果  
填空3:将指定的构件添加到容器中, add()方法提供此操作, 本程序需将按钮构件添加到容器中, newButton为新建的按钮需添加进去  
填空4:程序要求将窗口显示出来, CloseFrame类继承JFrame类, 提供setVisible()方法  
填空5 : ActionListener用于接收操作事件的侦听器接口。  
对处理操作事件感兴趣的类可以实现此接口, 而使用该类创建的对象可使用组件的 addActionListener 方法向该组件注册。  
在发生操作事件时, 调用该对象的 actionPerformed 方法。  
本程序关闭窗口需要实现该类