1、在 java 中重写方法应遵循规则的包括（）
A.访问修饰符的限制一定要大于被重写方法的访问修饰符

B.可以有不同的访问修饰符

C.参数列表必须完全与被重写的方法相同

D.必须具有不同的参数列表

答案：BC

解析：两同两小一大原则：

两同：方法名和参数列表相同

两小：返回值或声明异常比父类小（或相同）

一大：访问修饰符比父类的大（或相同）

2、Java 中的集合类包括 ArrayList 、 LinkedList 、 HashMap 等，下列关于集合类描述正确的是？（）

A.ArrayList 和 LinkedList 均实现了 List 接口

B.ArrayList 访问速度比 LinkedList 快

C.随机添加和删除元素时，ArrayList 的表现更加快速

D.HashMap 实现 Map 接口，它允许任何类型的键和值对象，并允许将 NULL 用作键或值

答案：ABD

解析：A、HashMap 实现了 Map 接口的，它的 Key 和 Value 都可以是 null，但是 Hashtable 种，Key 和 Value 都不能是 null。

B、ArrayList 与 LinkedList 都实现了 List 接口，继承了 AbstractList 类。

C、ArrayList 底层是动态数组是实现，随机位置添加和删除，都需要移动数组的数据，而 LinkedList 底层是双向链表，只需要修改 Node 节点的引用。

D、随机访问数组要比链表块。

3、将一组无序的正整数重新排列成有序序列,其方法有()

A.拓扑排序　　B.快速排序　　C.堆排序　　D.基数排序

答案:BCD

解析：由某个集合上的一个偏序得到该集合上的一个全序，这个操作称之为拓扑排序。
其它三个都是可以对无序的正整数排序。
4、Java1.8 版本之前的前提，Java 特性中,abstract class 和 interface 有什么区别
A.抽象类可以有构造方法，接口中不能有构造方法

B.抽象类中可以有普通成员变量，接口中没有普通成员变量

C.抽象类中不可以包含静态方法，接口中可以包含静态方法

D.一个类可以实现多个接口，但只能继承一个抽象类。

答案：ABD

解析：接口和抽象类的区别：
1.接口的方法默认为 public abstract ,接口中的变量默认为 public static final，在 java8 之前所有的方法不能有实现
抽象类中可以有非抽象方法
2.一个类可以实现多个接口，但只能继承一个抽象类
3.一个类实现接口，要实现该接口的所有抽象方法。
4.接口不能被实例化，但可以声明，但是必须引用一个实现该接口的对象。
抽象类可以有构造方法，但是不能被直接通过 new 进行实例化。但可以通过子类继承，实例化子类的时候抽象类也会被实例化。
这其实用到了多态，向上转型。父类引用指向子类对象。
5.从设计层面来说，抽象类是对类的抽象，是一种模板设计，接口是行为的抽象，是一种行为的规范。

5、关于数据结构，下面叙述中正确的是（）

A.直接选择排序是一种稳定的排序方法

B.哈弗曼树带权路径长度最短的树，路径上权值较大的结点离根较近

C.拓扑排序是指结点值得有序排序

D.当从一个最小堆中删除一个元素时，需要把堆尾元素填补到堆顶位置，然后再按条件把它逐层向下调整到合适位置

答案：BD

解析:不稳定的排序：快 些 选 堆（快速排序，希尔排序，选择排序，堆排序）拓扑排序是结点的逻辑排序。

6、（）完成对数据库数据的建表与更新

A.DCL　　B.DDL　　C.DML　　D.DQL

答案：BC

解析：DML（data manipulation language），数据操作语言，如增删该查
DDL（data definition language），数据定义语言，如建表删表，修改表字段（改变表结构）
DCL（data control language），数据控制语言，如权限授权
DQL（data query language），数据查询语言
7、Java 中线程的几种状态是()。

答案：新建状态、就绪状态、运行状态、阻塞状态及死亡状态。

8、Java 克隆方式有哪几种？

答案：浅克隆，通常只是对克隆的实例进行复制，但里面的其他子对象，都是共用的。
深克隆：克隆的时候会复制它的子对象的引用，里面所有的变量和子对象都是又额外拷贝了一份。
9、简述 Java 内存模型？

答案：Java 虚拟机规范中将 Java 运行时数据分为六种。
1.程序计数器:是一个数据结构,用于保存当前正常执行的程序的内存地址。Java 虚拟机的多线程就是通过线程轮流切换并分配处理器时间来实现的,为了线程切换后能恢复到正确的位置，每条线程都需要一个独立的程序计数器，互不影响，该区域为“线程私有”。
2.Java 虚拟机栈：线程私有的，与线程生命周期相同，用于存储局部变量表，操作栈，方法返回值。局部变量表放着基本数据类型，还有对象的引用。
3.本地方法栈：跟虚拟机栈很像，不过它是为虚拟机使用到的 Native 方法服务。
4.Java 堆：所有线程共享的一块内存区域，对象实例几乎都在这分配内存。
5.方法区：各个线程共享的区域，储存虚拟机加载的类信息，常量，静态变量，编译后的代码。
6.运行时常量池：代表运行时每个 class 文件中的常量表。包括几种常量：编译时的数字常量、方法或者域的引用。
10、两个乒乓球队进行比赛，各出三人。甲队为 a,b,c 三人，乙队为 x,y,z 三人。已抽签决定比赛名单。有人向队员打听比赛的名单。a 说他不和 x 比，c 说他不和 x,z 比，请编程序找出三队赛手的名单

解析：public class Prog {

```
    public static void main(String[] args) {
        String[] team1 = {"a","b","c"};
        String[] team2 = {"x","y","z"};
        for(int i=0; i<3; i++){
            for(int j=0; j<3; j++){
                if(i == 0 && j == 0)//a 说他不和 x 比
                    continue;
                else if(i == 2 && (j == 0 || j == 2))
                    continue;//c 说他不和 x,z 比
                else{
                    System.out.println(team1[i] + "<-->" + team2[j]);
                }
            }
        }
    }
}
```