厦门大学《Java 程序设计》课程试卷

软件 学院 软件工程 系 2013 年级 软件工程 专业

主考教师：王美红 试卷类型：（A 卷/B 卷）

1、 Single choice(10 points)

Identify the letter of the choice that best completes the statement or answers the question.

(1) What is displayed by System.out.println("1" + new Integer(2) + 3)?

A. 123 B. 15

C. 6 D. Compile error

(2) Which one of these lists contains only Java programming language keywords?

A. class, if, void, long, Int, continue

B. instanceof, break, finally, default, throws

C. try, virtual, throw, final, volatile, transient

D. byte, break, assert, switch, include

(3) Which method must exist in every Java application?

A. Main B. paint

C. init D. begin

(4) Which will legally declare, construct, and initialize an array?

A. int [] myList = {"1", "2", "3"};

B. int [] myList = (5, 8, 2);

C. int myList [] [] = {4,9,7,0};

D. int myList [] = {4, 3, 7};

(5) When you try to compile MyClass, the java compiler gives an error message

MyClass is not abstract and does not override abstract method <some method> in java.util.Comparator

Which of the following is <some method> in the error message?

A. equals(myClass)

B. compareTo(myClass)

C. compare(myClass, myClass)

D. toString()

(6) Consider the following code segment:

```
List<String> list = new ArrayList<String>();
list.add("[");      list.add("A");      list.add("]");
System.out.println(list);
ListIterator it = list.listIterator();
while(it.hasNext())
{
        if ("[".equals(it.next()) || "]".equals(it.next()))
                it.remove();
        else
                it.add("*");
}
System.out.println(list);
```

The first output line is

[[, A,]]

What is the second output line?

A. [A]

B. [A, B]

C. [B, A]

D. ClassCaseException

(7) Which of the following best describes the set of all pairs of values for boolean variables *a* and *b*, such that

(!a && b) == !(a || b)

evaluates to true?

A. Empty set

B. Only one pair: a == true, b == false

C. Two pairs in which a == true

D. Two pairs in which a != b

E. All four possible combinations of values

(8) Which of these cannot be used as a case label in a switch statement :

A. case ‘x’:

B. case x: // assume x is a final int variable

C. case 5:

D. case 2.3:

(9) The _____ loads Java bytecode to the memory.

A. java

B. bytecode verifier

C. JVM’s class loader

D. java compiler

(10) The statement System.out.printf("%.2f", 1234.567) outputs _____

A. 1234.567

B. 1234.5

C. 1234.57

D. 1234

2、 Multiple choice(10 points)

Identify the letters of the choices that best complete the statement or answer the question.

(1) Which three are methods of the Object class?

A. notify()　　B. notifyAll()

C. wait(long msecs)　　D. sleep(long msecs)

(2) public interface Foo

```
{
    int k = 4; /* Line 3 */
}
```

Which three piece of codes are equivalent(等价地) to line 3?

A. final int k = 4;　　B. public int k = 4;

C. static int k = 4;　　D. abstract int k = 4;

E. volatile int k = 4;　　F. protected int k = 4;

(3) Which are not valid declarations of a boolean?

A. boolean b1 = 0; B. boolean b2 = 'false';

C. boolean b3 = false; D. boolean b5 = no;

(4) Which of the following expression cannot yield an integer between 0 and 100, inclusive?

A. (int)(Math.random() * 100)

B. (int)(Math.random() * 100) + 1

C. (int)(Math.random() * 100 + 1)

D. (int)(Math.random() * 101)

(5) For any object obj, a call obj.getClass().getName() returns the name of the obj's class.

Suppose

System.out.println(new X() + "+" + new Y());

displays

X+Y

Which of the following implementations would produce that result?

A. Class X has a method **public String toString() { return "X"; }**

and class Y has a method **public String toString() { return "Y"; }**

B. Both class X and class Y extend class M that has a method **public String toString() { return getClass().getname(); }**

C. Both class X and class Y extend an abstract class M that has methods **public abstract String getName();**

public String toString() { return getname(); }

3、 True/False(10 points)

Indicate whether the sentence or statement is true or false.

(1) java.util packages is automatically loaded, so you don't need to import it.

(2) The JVM performs automatic garbage collection to reclaim the memory occupied（占有） by objects that are no longer used.

(3) The finally method is called by the garbage collector to perform termination housekeeping on an object just before the garbage collector reclaims（回收再利用） the object’s memory.

(4) Every class must have at least one constructor. If you do not provide any in a class’s

declaration, the compiler will not create any constructor for you.

(5) Operator **instanceof** determines if an object has the *has-a* relationship with a specific type.

(6) Polymorphism enables us to write programs that process objects that share the same superclass as if they're all objects of the superclass; this can simplify programming

(7) A new class of objects can be created quickly and conveniently by inheritance—then new class absorbs（吸收） the characteristics of an existing class.

(8) Swing GUI components are thread safe.

(9) Java allows synchronized methods and synchronized statements.

(10) Throws clause specifies the exceptions the method throws.

4、 Short answer(40 points)

(1) (5 points)Please explain the life cycle of a Thread.

(2) (6 points)Please explain the characteristics of java interface and abstract class.

(3) (5 points)What is checked exception? Please explain how to use it.

(4) (4 points)What is the difference between Set and List.

(5) (6 points)Illustrate(举例说明) how to make an object as a separate thread running

(6) (8 points)Illustrate how to communicate between server and client based on Socket.

(7) (6 points)Please explain how to add menus to a JFrame object.

5、 Completion (30 points)

(1) (4 points)What is the output of the following code segment?

```
Map m = new TreeMap();
m.put("La", "La");
m.put("La-La", "La");
m.put("La-La-La", "Ye-Ye");
Iterator it = m.keySet().iterator();
while (it.hasNext())
    System.out.println(m.get(it.next()) + " ");
```

_____（1）_____

(2) (2 points)Many of Java's networking classes are contained in package _____（2）_____

(3) (2 points)ExecutorService method _____（3）_____ends each thread in an ExecutorService as soon as it finishes executing its current Runnable, if any.

(4) (4 points) _____（4）_____and _____（5）_____ enable you to specify, with a single method declaration, a set of related methods, or with a single class declaration, a set of related types, respectively.

(5) (6 points)Write a simple generic version of method isEqualTo that compares its two arguments with the equals method and returns true if they're equal and false otherwise.

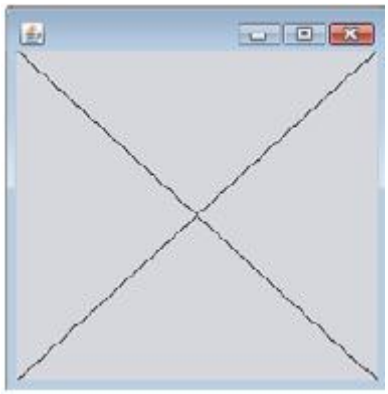
_____（6）_____

_____

(6) (6 points)Complete the following program section to show a message dialog when user pressed "ok" button.

```
JButton_ button = new JButton("ok");
button. _____（7）_____
_____;
```

(7) 6 (points) Complete the following program to show a frame like the following fig.



```
import java.awt.Graphics;
import javax.swing.JPanel;
public class DrawPanel extends JPanel
{
   // draws an X from the corners of the panel
   public void paintComponent( Graphics g )
   {     _____（8）_____
         _____
   } // end method paintComponent
```

```
} // end class DrawPanel

import javax.swing.JFrame;

public class DrawPanelTest
{
    public static void main( String[] args )
    {
        // create a panel that contains our drawing
        DrawPanel panel = new DrawPanel();
        // create a new frame to hold the panel
        JFrame application = new JFrame();
        // set the frame to exit when it is closed
        application.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE );
        application.add( panel ); // add the panel to the frame
        application.setSize( 250, 250 ); // set the size of the frame
        application.setVisible( true ); // make the frame visible
    } // end main
} // end class DrawPanelTest
```

一、

ABADC ACDCC

二、

ABC

ABC

ABD

ABC

AB

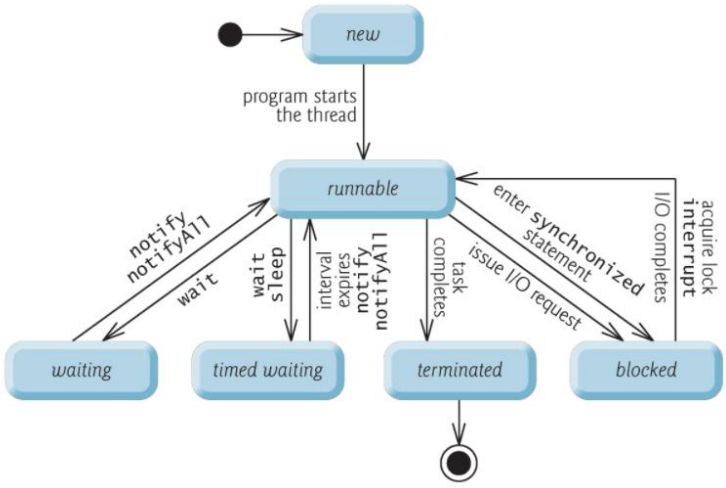
三、

FTFFF TTFTT

四、

(1) (5 points)Please explain the life cycle of a Thread.

答：线程在其生命周期中经历不同的阶段。以下是生命周期的各个阶段。



(2) (6 points)Please explain the characteristics of java interface and abstract class.

答：（1）**接口是抽象方法的集合**。如果一个类实现了某个接口，那么它就继承了这个接口的抽象方法。这就像契约模式，如果实现了这个接口，那么就必须确保使用这些方法。接口只是一种形式，接口自身不能做任何事情。

（2）如果一个类有 abstract 方法，这个类一定是抽象类。可以有自己的变量。抽象类用来捕捉子类的通用特性，不能被实例化，只能被用作子类的超类。抽象类是被用来创建继承层级里子类的模板。

(3) (5 points)What is checked exception? Please explain how to use it.

答：Exception 的子类，但非 RuntimeException。① 由不在程序控制范围内的条件引起。② 编译器检查每个方法调用和方法声明，以确定该方法是否抛出 checked exceptions。如果是，编译器将验证选中的异常是否被 catch 或在 throws 子句中声明。如果未满足 catch 或 declare 要求，编译器将发出一条错误消息，指示必须捕获或声明异常。

(4) (4 points)What is the difference between Set and List.

答：Set 一个集合类，不包含重复项，派生 HashSet、TreeSet。List 有序的集合类，可以包含

重复的元素，派生出 ArrayList，LinkedList，Vector。

(5) (6 points)Illustrate(举例说明) how to make an object as a separate thread running

答：
比如说我们想在游戏里面让不同角色并行攻击。首先创建类 Battle，实现 Runnable 接口。启动的时候，创建一个 Battle 对象 battle1，然后再根据该 battle 对象创建一个线程对象，借助线程对象的 start()方法，启动一个新的线程。
在创建 Thread 对象的时候，把 battle1 作为构造方法的参数传递进去，这个线程启动的时候，就会去执行 battle1.run()方法了。

```
Battle battle1 = new Battle(gareen,teemo);
new Thread(battle1).start();
```

(6) (8 points)Illustrate how to communicate between server and client based on Socket.

答：（1）服务器端首先创建 ServerSocket 对象 server。然后用 server.accept()创建一个 Socket 对象 connection 等待客户端连接，没有连接则处于阻塞状态。接连后进行通信。之后进入处理阶段，最后用 close()关闭流结束。（2）客户端首先创建 Socket 对象 connection。然后进行输入输出，流通信。最后用 close()关闭流结束。

1、ServerSocket server = new ServerSocket(portNumber, queueLength); // 创建ServerSocket对象（端口号和长度）
2、**Socket** connection = server.**accept()**; **// 服务端等待客户端连接**，没连接则处于阻塞状态
3、Socket.getOutputStream(); Socket.getInputStream(); **// 进行通信**，输入输出流，OutputStream类、InputStream类、读写操
// to get the **OutputStream and InputStream** objects that enable that server to communicate with the client by sending and receiving bytes.
4、//the processing phase. **// 处理阶段**
5、Socket.close(); **//关闭流和连接**

- 客户端

Socket connection = new Socket(serverAddress, port); //创建Socket对象
Socket.getOutputStream(); Socket.getInputStream(); //OutputStream类、InputStream类，**输入输出流通信**
Socket.close(); **//关闭流**

（7）略
五、（1）
La
La
Ye-Ye
（2）java.net（3）shutdown() （4）泛型类、泛型方法
（5）

```
//修饰符 <代表泛型的变量> 返回值类型 方法名（参数）{}
 public static <T> boolean isEqualTo(T arg1, T arg2) {
        return arg1.equals(arg2);
 }
```

（6）（7）略