四川大学期末考试试题（闭卷）

（2016~2017 学年第 2 学期）　　A 卷

课程号：**311141040**　课程名称：**面向对象程序设计导论**　任课教师：＿＿＿＿＿＿

适用专业年级：**软件工程 2016 级**　学号：＿＿＿＿＿＿　姓名：＿＿＿＿＿＿

考生承诺

我已认真阅读并知晓《四川大学考场规则》和《四川大学本科学生考试违纪作弊处分规定（修订）》，郑重承诺：

1、 已按要求将考试禁止携带的文具用品或与考试有关的物品放置在指定地点；

2、 不带手机进入考场；

3、 考试期间遵守以上两项规定，若有违规行为，同意按照有关条款接受处理。

考生签名：

题　　号	一(40%)	二(20%)	三(40%)					
得　　分								
卷面总分			教师签名		阅卷时间			

注意事项：1. 请务必将本人所在学院、姓名、学号、任课教师姓名等信息准确填写在试题纸和添卷纸上；

2. **请将答案全部填写在本试题纸上；**

3. 考试结束，请将试题纸、添卷纸和草稿纸一并交给监考老师。

评阅教师	得分

一、单项选择题（本大题共 20 小题，每小题 2 分，共 40 分）

提示：在每小题列出的四个备选项中只有一个是符合题目要求的，请将其代码填写在下表中。错选、多选或未选均无分。

1	2	3	4	5	6	7	8	9	10
11	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**

1. Which of the following is the string concatenation operator in Java?

（A）　+

（B）　^

（C）　&

（D）　++

2. What is the value of the String S after the following line?

String S = (new String("arach")).substring(0,2) +(new String("nophobia")).substring(3);

（A）　"arachobia"

（B）　"arnophobia"

（C）　"arhobia"

（D） "rachobia"

3. According to the document entitled Code Conventions for the Java Programming Language, file suffixes used by Java software include which of the following?

I. .obj　II . .class　III. .h

（A） I and III only

（B） II and III only

（C） I and II only

（D） II only

4. Which of the following is true about RuntimeException and its subclasses?

（A） If a method throws a RuntimeException, the use of the try/catch block is optional.

（B） The FileIOException class is a subclass of RuntimeException.

（C） In general, handling of RuntimeException should be done at compile time.

（D） In general, RuntimeException must be caught with a try/catch block.

5. Which of the following is an application of the principle of inheritance:

（A） An object of class A has a reference to a class B object.

（B） Several methods have the same name, but have different signatures.

（C） Fields are usually declared private.

（D） All classes are ultimately derived from the super class called Object.

6. Which of the following statements about constructors is correct:

I. A constructor has the same name as the class name.

II. A constructor is responsible for the initialization of an object's fields.

III. A class can have several constructors.

IV. Constructor methods have no return type

（A） I and II　（B）I,II and IV　（C） II and III　（D）All of above

7. In the example code fragment shown below, the keyword abstract:

```
public abstract class Test // . . . more class code
```

（A） Is needed so that the class Test can implement abstract methods.

（B） Implies that no object of type Test can ever be created.

（C） Makes class Test independent of all other classes, in particular, it is not a subclass of the class Object.

（D） Ensures that only one object of type Test is ever created.

8. What are valid arguments to the instanceof operator?

（A） a class object and a class type

（B） any primitive type

（C） boolean type only

（D） class types only

9. The following statements compare and contrast array and ArraryList. Which is true?

（A） Both ArrayList and array objects automatically keep track of their capacity and the number of elements actually in use.

（B） ArrayList can expand its storage space as needed; an array cannot change the initial size of the array.

（C） The array type extends from Object. The ArrayList type extends from array.

（D） The elements in an ArrayList can be primitives (int, float, double etc.) or reference-to values. An array can store only reference-to values.

10. When would you use a private constructor?

（A） When you get bored with public

（B） If you want to disallow instantiation of that class from outside that class

（C） If you want to protect your class's members from outside modification

（D） Never, it's not allowed

11. When writing data to a file using a FileOutputStream, at what point is the data actually written to the file?

I. Immediately after the write function is called

II. When the data buffer is full

III. When the close function is called

（A） I only

（B） III only

（C） II and III

（D） II only

12. Suppose the class Undergraduate extends the class Student which extends the class Person. Given the following variable declaration:

```
Person p = new Person();  Student s = new Student();
Undergraduate ug = new Undergraduate();
```

Which of the following assignments are legal?

I. p = ug;

II. p = new Undergraduate();

III. ug = new Student();

IV. ug = p;

V. s = new Person();

（A） III and IV

（B） I and IV

（C） I and II

（D） II, III and V

13.Given the function below, what is the value of g(3)?

```
private int g( int num ) {
    if( num <= 1 ) {
        return 1;
    } else {
        return 3*g( num-1 ) + g( num-2 );
    }
}
```

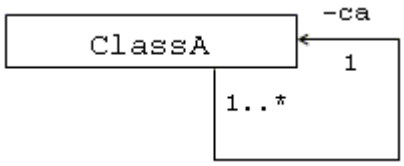
（A） This recursion is incorrect in some way.

（B） 43

（C） 4

（D） 13

14. Consider the following UML class diagram.



The diagram describes a

（A） class without methods

（B） relationship between a subclass and a superclass

（C） one-to-one relationship

（D） self-containing class

15.When a subclass defines an instance method with the same return type and signature as a method in its parent, the parent's method is said to be

（A） hidden

（B） Overridden

（C） private

（D） overloaded

16.Consider the Java program below.

```
public class Arr{
    public static void main(String[] args) {
```

```
        int[] a = {1, 2, 3};
        System.out.println(a[1]);
        System.out.println(a[3]);
    }
}
```

Which of the following is true about the result of executing the program?

（A） The number 2 is printed and a run-time exception terminates execution.

（B） The number 2 is printed and there is no abnormal termination.

（C） The number 1 is printed and there is no abnormal termination.

（D） The number 3 is printed and a run-time exception terminates execution.

17.A design pattern is typically used to

（A） ensure that code executes at optimal speed during runtime

（B） reduce the number of classes in the design of a program

（C） allow the use of object-orientated concepts in a language that is not object-oriented

（D） describe a practical solution to a common design problem

18.Which of the following describes the Behavioral pattern correctly?

（A） It provide a way to create objects while hiding the creation logic, rather than instantiating objects directly using new opreator.

（B） It concern class and object composition. Concept of inheritance is used to compose interfaces and define ways to compose objects to obtain new functionalities.

（C） It are specifically concerned with communication between objects.

（D） It specifically concerned with the presentation tier.

19.Which of the following is correct about Singleton design pattern.

（A） This type of design pattern comes under creational pattern.

（B） This pattern involves a single class which is responsible to create an object while making sure that only single object gets created.

（C） Singleton class provides a way to access its only object which can be accessed directly without need to instantiate the object of the class.

（D） All of the above.

20.A class which implements the ActionListener interface must implement which method?

（A） void handle(ActionEvent e)

（B） void actionPerformed(ActionEvent e)

（C） void eventDispatched(AWTEvent e)

（D） String getActionCommand(ActionEvent e)

评阅教师	得分

二、设计模式（共 **20** 分）。

1. Which design pattern is applied in the code snippet below?
2. Draw UML diagram for this pattern.
3. Revise the code if we need 2 more objects of class PrintSpooler.

```
public class PrintSpooler {
    private static final PrintSpooler instance = new PrintSpooler();
    private PrintSpooler() {}
    public static PrintSpooler getInstance() {
       return instance;
    }
}
```

评阅教师	得分

三、编程题（共 40 分）。

Write a working Java program that manages monthly billing information for a cell phone company's customers. In crafting your solution, you can use the *Input* class that is supplied (the *JavaDoc* API is shown below).

Overview of the Requirements

All customers are charged a base rate for their phone plan, but they will face additional charges based on their choice of plan. There are two types of plans:

- *Per Minute Plan*: In addition to the *base rate*, customers will pay a per-minute charge of 25 cents.
- *Flat Rate Plan*: In addition to the *base rate*, customers will pay an extra premium fee which can vary for each customer.

Details: Hierarchy of Phone *Classes*

There are 3 class: *Phone*, *PerMinutePhone* and *FlatRatePhone*.

The *Phone* Class

The *Phone* class will be *abstract*. It will have the following instance variables all of which must be *private*:

- *phoneNumber*: holds the phone number in a *String* object. There is no need for any validation of this field during input.
- *baseRate*: holds a floating-point number. This value is entered during keyboard input.
- *taxAmt*: holds the calculated tax. This value is calculated when all customer bills are displayed.

The *Phone* class will also hold a constant for the HST tax (which is now going to be 13%).

The *PerMinutePhone* and *FlatRatePhone* Classes

Both *PerMinutePhone* and *FlatRatePhone* are subclasses of *Phone*. The *PerMinutePhone* class has the following *private* instance variable:

- *minutes*: holds a floating-point number which represents the total number of minutes used during that month.

The *FlatRatePhone* class has the following *private* instance variable:

- *flatRatePremium*: holds the additional cost of having a flat-rate plan.

The Class Methods

The organization of methods is largely up to you, with the following requirements. You must implement overridden methods (virtual methods) to enter values from the keyboard, and to calculate the before-tax bill total. The tax calculation can be left as an exclusive responsibility of the *Phone* class.

Details: Customer *Class*

The *Customer* class will hold the customer's name in a *String* object, and have a reference to a *Phone* object. It will require suitable methods to perform:

- input
- processing / calculations
- display of output

Details: PhoneCompany *Class*

The *PhoneCompany* class is used to create an object, then the *addCustomer()* and *calcBills()* methods are called as a result of a user's menu choice.

The *addCustomer()* method will create a new *Customer* object and create the correct type of *Phone* object based on the user's choice. The *PhoneCompany* class must ensure that the addition of new customers will fit in the collection that is used to manage all the added customers.

The *calcBills()* method will determine the billing total for each customer, then display the results.

Methods Available in Input Class.

Details: TestPhoneCompany *Class*

Implement the TestPhoneCompany class. The following screen capture demonstrates a sample run of the program. You can see that two customers have been added, then the bills are displayed for those two customers. Of course, more customers could have been added.

Class *Input* Method Summary	
static int	**getInt**(String prompt) Retrieves an *int* from the keyboard.
static int	**getInt**(String prompt, int low, int high) Retrieves an *int* from the keyboard which is guaranteed to be between the specified ranges (inclusive).
static int	**getDouble**(String prompt, double low, double high) Retrieves a *double* from the keyboard which is guaranteed to be between the specified ranges (inclusive).
static String	**getString**(String prompt) Retrieves a *String* from the keyboard where the text is delimited by the <*Enter*> key.