

电子科技大学 2024-2025 学年第 1 学期期末考试 A 卷

参考答案及评分细则

考试科目: 软件工程与实践 考试形式: 闭卷 考试日期: 2024 年 12 月 19 日

本试卷由 4 部分构成, 共 10 页。考试时长: 120 分钟

成绩构成比例: 平时成绩 50 %, 期末成绩 50 %

题号	一	二	三	四	五	六	七	八	合计
得分									

得 分

一、程序阅读题 (每小题 5 分, 共 15 分)

程序阅读后, 在每题后答题框内填写。

1、阅读程序给出运行后的输出。(5 分)

```
public class ArrayDemo {  
    public static void main(String[] args) {  
        int[][] a = { { 1, 2 }, { 3, 4, 5 }, { 6, 7, 8, 9 } };  
        System.out.println("数组 a 一维长度: " + a.length);  
        System.out.println("数组 a[1] 的长度: " + a[1].length);  
        int[][] b = new int[3][];  
        b[0] = new int[2];  
        b[1] = new int[3];  
        b[2] = new int[4];  
        for (int i = 0; i < b.length; i++) {  
            for (int j = 0; j < b[i].length; j = j + 2) {  
                b[i][j] = a[i][j] + i + j;  
            }  
        }  
        for (int[] b_array : b) {  
            for (int element : b_array) {  
                System.out.print(element + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

答案：

数组 a 一维长度：3 (1 分)
数组 a[1] 的长度：3 (1 分)
1 0 (1 分)
4 0 8 (1 分)
8 0 12 0 (1 分)

2、阅读程序给出运行后的输出。(5 分)

```
abstract class Animal {  
    String name;  
    public Animal(String name) {  
        this.name = name;  
        System.out.println("Animal 的构造器被调用: " + name);  
    }  
    public abstract void sound();  
    public void description() {  
        System.out.println("这是一只名为 " + name + " 的动物。");  
    }  
    public static void info() {  
        System.out.println("Animal 类: 关于动物的一般信息。");  
    }  
}  
class Dog extends Animal {  
    int age;  
    public Dog(String name, int age) {  
        super(name);  
        this.age = age;  
        System.out.println("Dog 的构造器被调用: " + age);  
    }  
    public void sound() {  
        System.out.println("sound() 方法被调用");  
    }  
    public void description() {  
        System.out.println("这是一只名为 " + name + " 的狗，它的年龄为 " + age  
+ " 岁。");  
    }  
    public static void info() {  
        System.out.println("Dog 类: 关于狗的具体信息。");  
    }  
}  
  
public class Main1 {  
    public static void main(String[] args) {  
        Animal a = new Dog("赛虎", 5);  
    }  
}
```

```
    a.info();
    a.sound();
    a.description();
}
}
```

答案 (每行 1 分):

Animal 的构造器被调用: 赛虎
Dog 的构造器被调用: 5
Animal 类: 关于动物的一般信息。
sound() 方法被调用
这是一只名为 赛虎 的狗, 它的年龄为 5 岁。

3、阅读程序给出运行后的输出。(5 分)

```
class Task implements Runnable {
    private static int count = 5;

    @Override
    public synchronized void run() {
        while (count > 0) {
            System.out.println(Thread.currentThread().getName() + ": Task " + count);
            count--;
            try {
                Thread.sleep(100);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}

public class UniqueThreadTest {
    public static void main(String[] args) {
        Task task = new Task();
        Thread t1 = new Thread(task, "Thread-1");
        Thread t2 = new Thread(task, "Thread-2");
        Thread t3 = new Thread(task, "Thread-3");

        t1.start();
        try {
            t1.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

```

t2.start();
try {
    t2.join();
} catch (InterruptedException e) {
    e.printStackTrace();
}

t3.start();
try {
    t3.join();
} catch (InterruptedException e) {
    e.printStackTrace();
}
}
}

```

答案：

Thread-1: Task 5 (1 分)
 Thread-1: Task 4 (1 分)
 Thread-1: Task 3 (1 分)
 Thread-1: Task 2 (1 分)
 Thread-1: Task 1 (1 分)

得 分

二、编程题（共 20 分）

1、(20 分) 定义接口 Shape_Interface，有方法 double get_area() 用于计算图形面积，方法 double get_perimeter() 用于计算周长，方法 void display() 用于显示图形的名称、周长和面积。

编写一个抽象类 Shape 类来实现 Shape_Interface 接口，具有数据成员：String name 用于保存图形名称，构造函数的参数是图形名称；

定义 Shape 的子类：圆(Circle)和矩形(Rectangle)。圆有属性半径，矩形有属性长和宽，都是 double 类型，圆的构造函数参数为半径值，矩形的构造函数参数为长和宽。

定义主类 Test，在其 main 方法中创建半径为 3 的三角形和长宽分别为 4 和 5.1 的矩形类的对象，并赋给 Shape_Interface 的对象 a，使用对象 a 来输出三角形和矩形的名称、周长和面积。

答案：

```

interface Shape_Interface{ //接口定义 2 分
    double get_area();
    double get_perimeter();
    void display();
}

```

```
abstract class Shape implements Shape_Interface { //Shape 类定义 2 分
```

考场教室 座位号 _____
任课教师 _____ 答题
学号 _____ 学名 _____
学院 _____ 姓名 _____
密封线 _____ 密封

```
String name;
Shape(String n){
    name = n;
}
}

class Circle extends Shape //每个成员 1 分, 类定义 1 分, 共 6 分。
{
    private double radius;

    Circle(double r){
        super("三角型");
        radius=r;
    }
    public double get_perimeter(){
        return 3.14 * radius * 2;
    }
    public double get_area(){
        return 3.14 * radius * radius;
    }
    public void display(){
        System.out.println(name + "周长是:" + get_perimeter() + "面积是:" +
get_area());
    }
}

class Rectangle extends Shape implements Shape_Interface //每个方法成员 1 分, 属性成员共 1
分, 类定义 1 分, 共 6 分。
{
    private double x;
    private double y;

    Rectangle(double a, double b){
        super("矩形");
        x=a;
        y=b;
    }
    public double get_perimeter(){
        return 2*(x+y);
    }
    public double get_area(){
        return x*y;
    }
    public void display(){
    }
}
```

```

        System.out.println(name + "周长是:" + get_perimeter() + "面积是:" +
get_area());
    }
}

class Test
{
    static public void main(String argv[])
    {
        Shape_Interface a = new Circle(3.0);
        a.display();
        a = new Rectangle(4.0, 5.1);
        a.display();
    }
}

```

得 分	

三、简答题（每小题 5 分，共 15 分）

1、软件工程的三要素是什么？并各用一句话简述每个要素的含义。

答案：

软件工程的三要素包括过程、方法、工具。 (2 分)

过程贯穿软件开发的各个环节，在各环节之间建立里程碑。 (1 分)

软件工程方法是完成软件工程项目的技术手段。 (1 分)

工具为软件工程的过程和方法提供自动化或半自动化的工具支持。 (1 分)

2、简述根据软件项目的不同资源（需求、计划、人员、资金等）情况，选择适当软件过程模型的方法。

答案（需求、计划、人员、资金四方面每个方面至少描述 1 点得 1 分，覆盖完全得 1 分）：

1) 前期需求明确的情况下，尽量采用瀑布模型。

2) 需求不稳定的情况下，尽量采用增量模型。

3) 不确定因素很多，很多东西无法提前计划的情况下，尽量采用增量模型或螺旋模型。

4) 用户无系统使用经验，需求分析人员技能不足的情况下，尽量借助原型模型。

5) 资金和成本无法一次到位的情况下，可采用增量模型。

6) 对于完成多个独立功能开发的情况，每个功能内部都尽量遵循瀑布模型。

7) 全新系统的开发必须在总体设计完成后再开始增量或并行。

8) 编码人员经验较少的情况下，尽量不要采用敏捷模型。

9) 增量和原型可以综合使用，但每一次增量都必须有明确的交付和出口原则。

3、试列举三个面向规模的度量指标，并进行解释。

答案（答出三条得 3 分，解释正确 1 条得 1 分，满分 5 分）：

千行代码（KLOC）：这些代码指的是源代码，通过源代码的行数来直观度量一个软件程序有多大规模

生产率（PM）： $PM = L / E$, L 表示代码总量(单位：KLOC), E 表示软件工作量(单位：人月)

每千行代码的平均成本（CKL）： $CKL = S / L$, S 为软件项目总开销，L 表示代码总量(单

位: KLOC)

代码出错率(EQRI): $EQRI = Ne / L$, Ne 表示代码出错的行数, L 表示代码总量(单位: KLOC)

文档与代码比(DI): $DI = Pd / L$, Pd 表示文档页数, L 表示代码总量(单位: KLOC)。

得 分

四、应用题 (每小题 10 分, 共 50 分)

1、某公司计划设计一个进销存系统, 主要用于管理商品的采购、销售和库存。经过调研, 该系统有两类主要用户, 即仓库管理员和销售人员。且系统主要功能为: 仓库管理员需要添加新商品及其初始库存; 仓库管理员需要更新商品库存信息, 并生成库存报表; 销售人员需要查询商品库存信息; 销售人员根据销售情况录入销售订单, 并且系统会检查库存是否足够并根据销售数据更新库存。

请分析上述的描述找出该系统的参与者与用例。

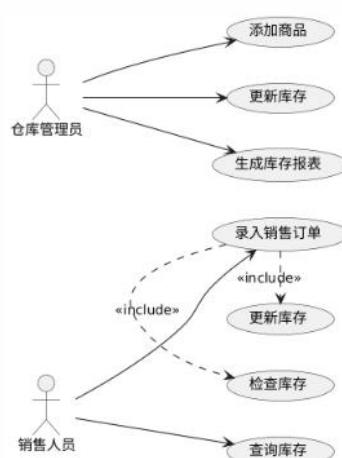
- (1) 用文字写出分析结果(参与者与用例)
- (2) 画出用例图

参与者

1. 仓库管理员: 负责管理商品信息和库存。
2. 销售人员: 负责销售订单的录入和库存查询。

用例

1. 添加商品, 添加新商品信息, 并指定初始库存。
2. 更新库存, 更新商品的库存信息。
3. 生成库存报表, 生成当前库存的详细报表。
4. 查询库存, 查询指定商品的库存信息。
5. 录入销售订单, 录入新的销售订单, 完成库存检查和更新。
6. 检查库存, 检查库存是否满足销售订单需求。
7. 更新库存, 根据销售订单更新库存信息。



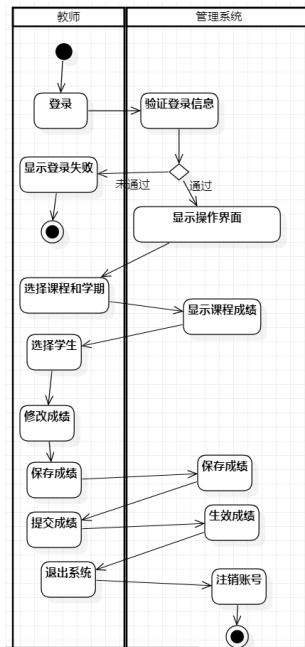
评分标准:

参与者每个 1 分, 共 2 分, 用例每个 1 分, 共 6 分, 包含类型用例每个 1 分, 共 2 分。

2、建立“教师修改学生成绩”用例的活动图。

- (1) 登录：教师登录时，系统验证教师输入的账号、密码、动态码等信息。如果验证未通过，则教师端显示登录失败，否则进入步骤(2)。
- (2) 系统显示教师操作界面。
- (3) 教师选择需要修改的课程和学期，系统显示教师选中的课程成绩。
- (4) 教师选择需要修改成绩的学生，修改学生分数，修改完成后保存学生成绩。
- (5) 教师点击“提交”按钮，系统将成绩修改生效。
- (6) 教师退出系统，系统注销教师账号。

根据上述描述，建立一个标识泳道的活动图（教师和管理系统）。



评分标准：泳道绘制正确得 2 分，开始、结束标识每个 1 分，步骤绘制正确 1 个得 1 分，共 6 分。

3、某成绩管理系统学生成绩录入的流程，如下：

- 1) 教师输入账号密码，用户数据库进行信息验证，若账号密码错误，反馈错误信息；
- 2) 教师选择需要录入成绩的课程，若课程不存在，则创建课程；
- 3) 教师输入学生的成绩信息，并提交给系统；
- 4) 系统将信息保存到数据库中，并反馈给教师相应的成功或失败信息。

画出该用例的顺序图。

答案：

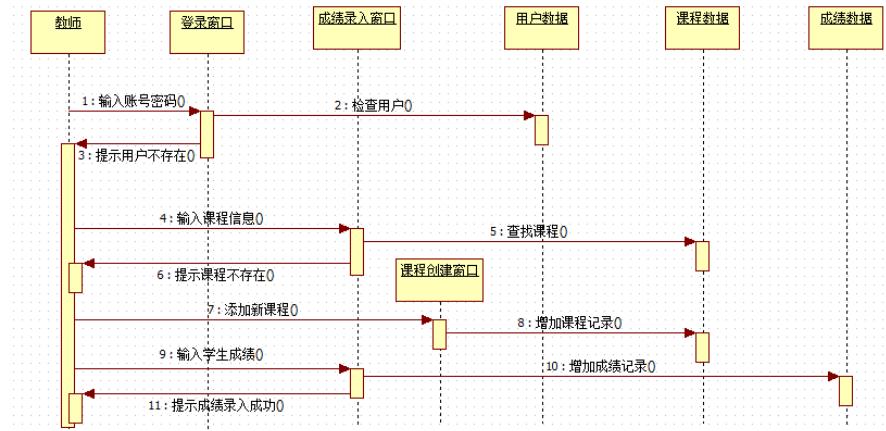
整体绘图规范 2 分，顺序图流程正确性 8 分。（错 1 个扣 1 分）

无.....效

题.....答

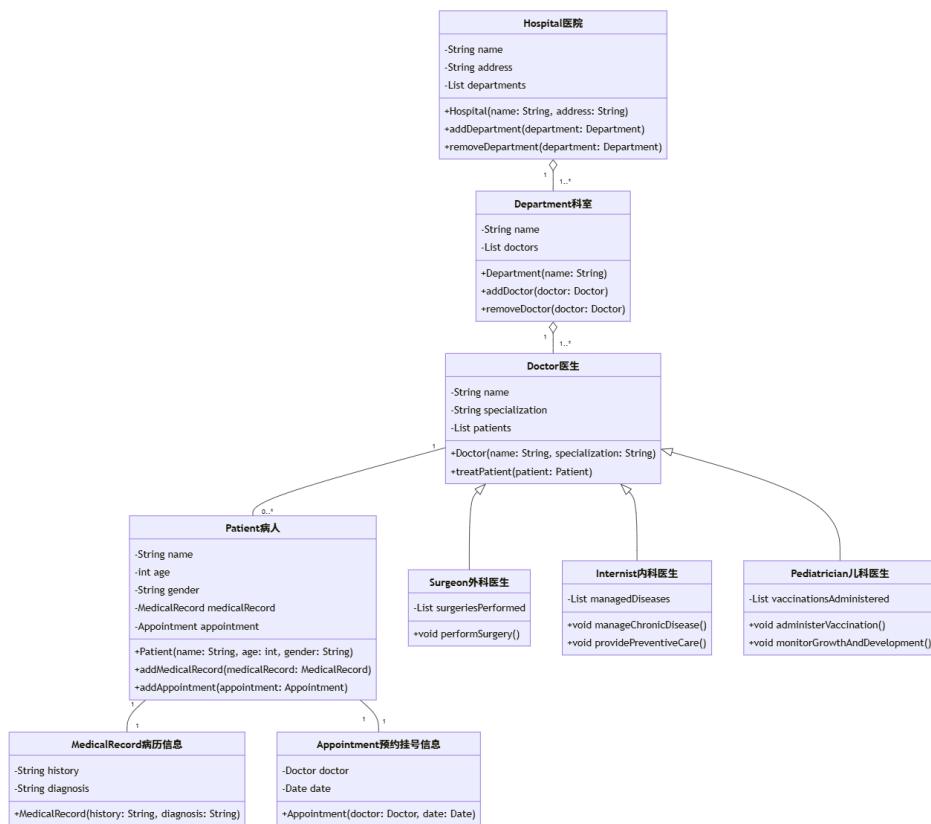
内.....以

线.....封.....密



4、某医院需要对病人进行管理，包括病人的基本信息（如姓名、年龄、性别等）、病历信息（如病史、诊断结果等）以及预约挂号信息。医院有多种科室，如内科、外科、儿科等，每个科室有多个医生，每个医生可以接诊多个病人，但每个病人在同一时间只能被一个医生接诊。现要为这家医院开发一个病人管理系统，设计系统的类图，并体现接口实现、父子类继承以及聚合关系。

答案：一个类中若属性、方法完整得 1 分，最高 6 分；标明继承(泛化)关系和聚合关系，每个 1 分，最高 4 分。



5、根据下面给出的规格说明，利用等价类划分的方法，给出足够的测试用例。“一个程序读入三个整数。把此三个数值看成是一个三角形的三个边。这个程序要打印出信息，说明这个三角形是三边不等的、是等腰的、还是等边的。”

判断标准：等价类划分正确，三行，第一行 2 分，第二行和第三行各 1 分，共 4 分；测试用例，共 6 个输出，每个输出对应的输入，完全正确，给 1 分，共 6 分。

答案：

设三角形的 3 条边分别为 A, B, C。如果它们能够构成三角形的 3 条边，必须满足： $A > 0, B > 0, C > 0$ ，且 $A+B > C, B+C > A, A+C > B$ 。

如果是等腰的，还要判断 $A=B$, 或 $B=C$, 或 $A=C$ 。

如果是等边的，则需判断是否 $A=B$, 且 $B=C$, 且 $A=C$ 。

划分等价类如下：

输入条件	有效等价类	无效等价类
是否为三角形的三条边	(A>0), (1)	(A≤0), (7)
	(B>0), (2)	(B≤0), (8)
	(C>0), (3)	(C≤0), (9)
	(A+B>C), (4)	(A+B≤C), (10)
	(B+C>A), (5)	(B+C≤A), (11)
	(A+C>B), (6)	(A+C≤B), (12)
是否为等腰三角形	(A=B), (13)	(A≠B) and (B≠C)
	(B=C), (14)	and (C≠A) (16)
	(C=A), (15)	
是否为等边三角形	(A=B) and (B=C) and (C=A)	(A≠B), (18)
		(B≠C), (19)
	(17)	(C≠A), (20)

设计测试用例如下：

序号	【A, B, C】	覆盖等价类	输出
1	【3, 4, 5】	(1), (2), (3), (4), (5), (6)	一般三角形
2	【0, 1, 2】	(7)	不能构成三角

2

3	【1, 0, 2】	(8)	形
4	【1, 2, 0】	(9)	
5	【1, 2, 3】	(10)	
6	【1, 3, 2】	(11)	
7	【3, 1, 2】	(12)	
8	【3, 3, 4】	(1), (2), (3), (4), (5), (6), (13)	
9	【3, 4, 4】	(1), (2), (3), (4), (5), (6), (14)	等腰三角形
10	【3, 4, 3】	(1), (2), (3), (4), (5), (6), (15)	
11	【3, 4, 5】	(1), (2), (3), (4), (5), (6), (16)	非等腰三角形
12	【3, 3, 3】	(1), (2), (3), (4), (5), (6), (17)	等边三角形
13	【3, 4, 4】	(1), (2), (3), (4), (5), (6), (14), (18)	
14	【3, 4, 3】	(1), (2), (3), (4), (5), (6), (15), (19)	非等边三角形
15	【3, 3, 4】	(1), (2), (3), (4), (5), (6), (13), (20)	