

# 全国计算机技术与软件专业技术资格（水平）考试

## 中级 软件设计师 2019 年 下半年 下午试卷 案例

（考试时间 150 分钟）

### 试题一 【说明】

某公司欲开发一款二手车物流系统，以有效提升物流成交效率。该系统的主要功能是：

(1) 订单管理:系统抓取线索，将车辆交易系统的交易信息抓取为线索。帮买顾问看到有买车线索后，会打电话询问买家是否需要物流，若需要，帮买顾问就将这个线索发起为订单并在系统中存储，然后系统帮助买家寻找物流商进行承运。

(2) 路线管理:帮买顾问对物流商的路线进行管理，存储的路线信息包括路线类型、物流商、起止地点。路线分为三种，即固定路线、包车路线、竞拍体系，其中固定路线和包车路线是合约制。包车路线的发车时间由公司自行管理，是订单的首选途径。

(3) 合约管理:帮买顾问根据公司与物流商确定的合约，对合约内容进行设置，合约信息包括物流商信息、路线起止城市、价格、有效期等。

(4) 寻找物流商:系统根据订单的类型(保卖车、全国购和普通二手车)、起止城市，需要的服务模式(买家接、送到买家等)进行自动派发或以竞拍体系方式选择合适的物流商。即:有新订单时，若为保卖车或全国购，则直接分配到竞拍体系中；否则，若符合固定路线和/或包车路线，系统自动分配给合约物流商，若不符合固定路线和包车路线，系统将订单信息分配到竞拍体系中。竞拍体系接收到订单后，将订单信息推送给有相关路线的物流商，物流商对订单进行竞拍出价，最优报价的物流商中标。最后，给承运的物流商发送物流消息，更新订单的物流信息，给车辆交易系统发送物流信息。

(5) 物流商注册:物流商账号的注册开通。

现采用结构化方法对二手车物流系统进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图。

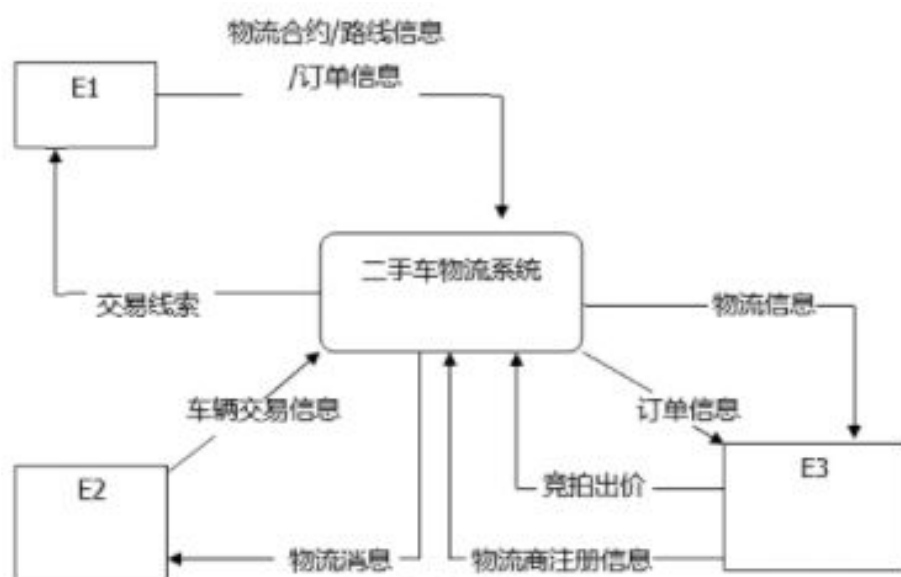


图 1-1 上下文数据流图

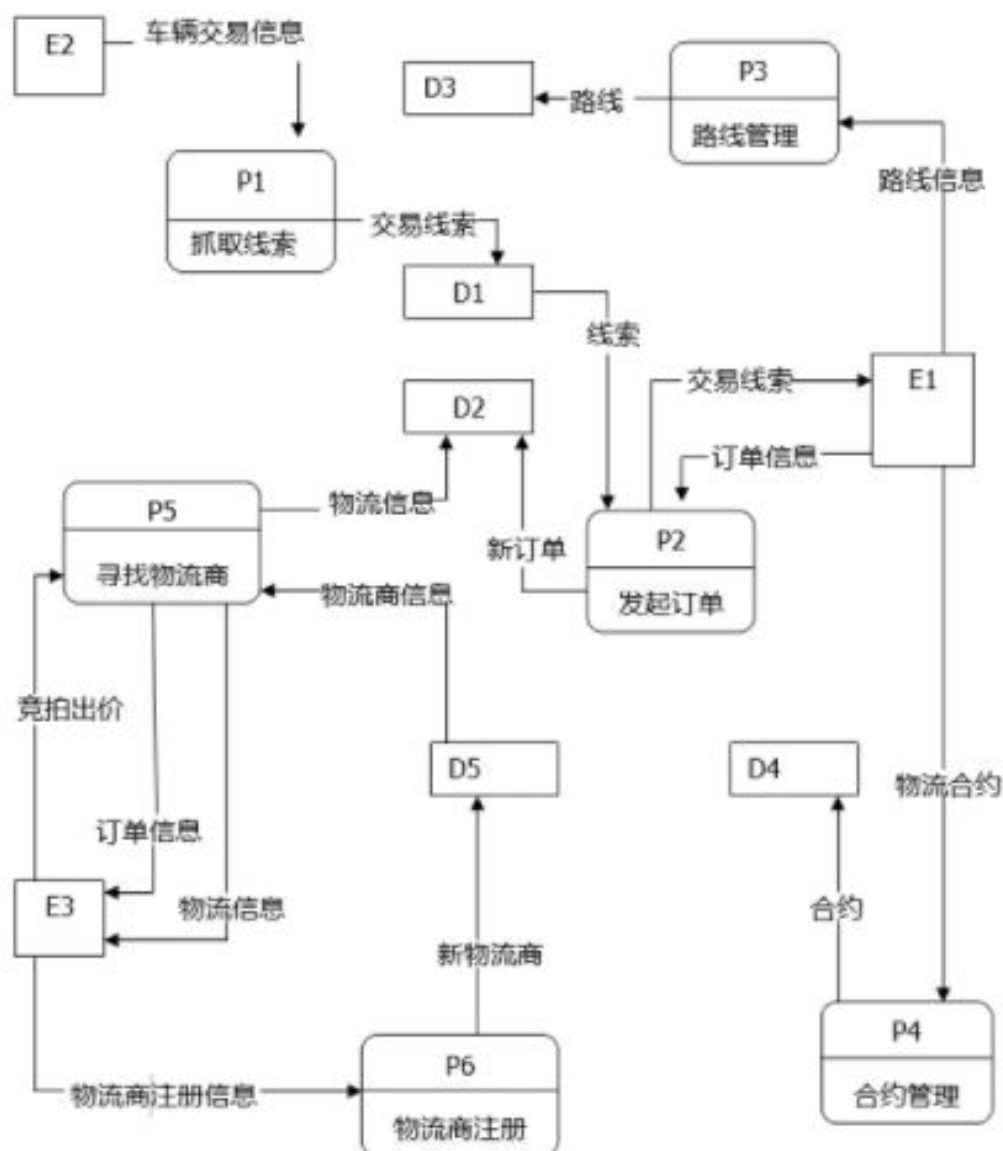


图 1-2 0层数据流图

**问题： 1.1**

(3 分)

使用说明中的词语，给出图 1-1 中的实体 E1~E3 的名称。

**问题： 1.2**

(5 分)

使用说明中的词语，给出图 1-2 中的数据存储 D1~D5 的名称。

**问题： 1.3**

(4 分)

根据说明和图中术语，补充图 1-2 中缺失的数据流及其起点和终点。

**问题： 1.4**

(3 分)

根据说明，采用结构化语言对“P5：寻找物流商”的加工逻辑进行描述。

**试题二 【说明】**

公司拟开发新入职员工的技能培训管理系统以便使新员工快速胜任新岗位。

1. 部门信息包括:部门号、名称、部门负责人、电话等，部门号唯一标识部门关系中的每一个元组，一个部门有多个员工，但一名员工只属于一个部门，每个部门只有一名负责人，负责部门工作。

2. 员工信息包括员工号、姓名、部门号、岗位、基本工资、电话、家庭住址等，其中员工号是唯一标示员工关系中的每一个元组。岗位有新入职员工，培训师、部门负责人等不同岗位设置不同的基本工资，新入职员工要选择多门课程进行培训，并通过考试取得课程成绩，一名培训师可以讲授多门课程、一门课程可由多名培训师讲授。

3. 课程信息包括课程号，课程名称、学时等；其中课程号唯一标识课程关系的每一个元组。

关系模式设计

部门(部门号，部门名，部门负责人，电话)

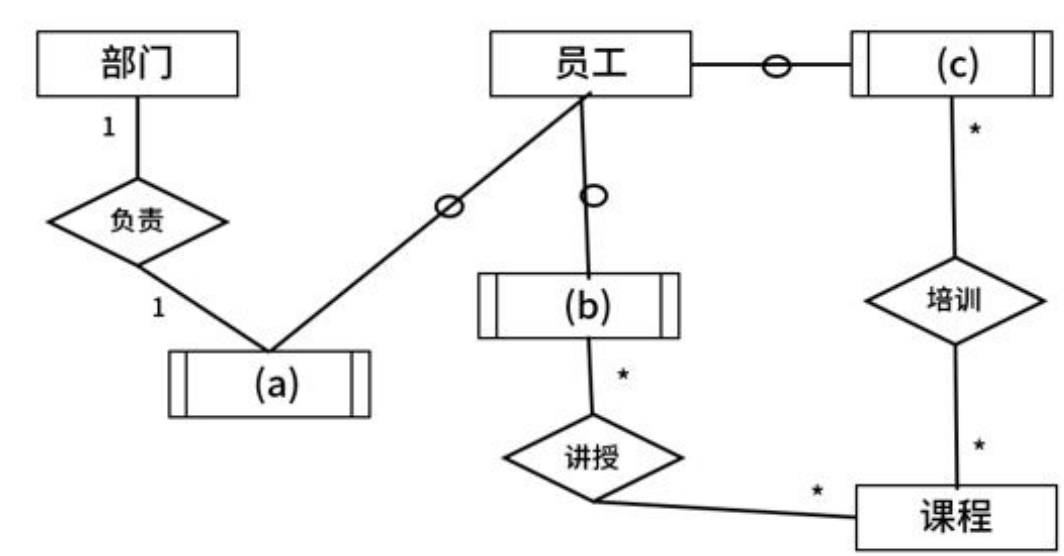
员工(员工号，姓名，部门号，d，电话，家庭住址)

课程(e，课程名称，学时)

讲授(课程号,培训师,培训地点)

培训(课程号,(f) )

根据需求阶段收集的信息，设计的实体联系图如图 2-1 所示



**问题： 2.1**

(1) 补充图 2 -1 中的空 (a) - (c)

(2) 图 2-1 中是否存在缺失联系，若存在，则说明所缺失的联系和联系类型

**问题： 2.2**

根据题意，将关系模式中的空 (d) - (f) 补充完整

**问题： 2.3**

员工关系模式的主键为 (g) ， 外键为 (h) ， 讲授关系模式的主键为 (i) ， 外键为 (j)

**问题： 2.4**

员工关系是否存在传递依赖?用 100 字以内的文字说明理由。

### 试题三 【说明】

某牙科诊所拟开发一套信息系统， 用于管理病人的基本信息和就诊信息。 诊所工作人员包括： 医护人员 (DentalStaff)、 接待员 (Receptionist) 和办公人员 (OfficeStaff) 等。 系统主要功能需求描述如下：

1. 记录病人基本信息(Maintainpatientinfo)。初次就诊的病人，由接待员将病人基本信息录入系统。病人基本信息包括病人姓名、身份证号、出生日期、性别、首次就诊时间和最后一次就诊时间等。每位病人与其医保信息(MedicalInsurance)关联。
2. 记录就诊信息(Recordofficevisitiinfo)。病人在诊所的每一次就诊，由接待员将就诊信息(OfficeVisit)录入系统。就诊信息包括就诊时间、就诊费用、支付代码、病人支付费用和医保支付费用等。
3. 记录治疗信息(Recorddentalprocedure)。病人在就诊时，可能需要接受多项治疗，每项治疗(Procedure)可能由多位医护人员为其服务。治疗信息包括：治疗项目名称、治疗项目描述、治疗的牙齿和费用等。治疗信息由每位参与治疗的医护人员分别向系统中录入。
4. 打印发票(Printinvoices)。发票(Invoice)由办公人员打印。发票分为两种：给医保机构的发票(InsuranceInvoice)和给病人的发票(PatientInvoice)。两种发票内容相同，只是支付的费用不同。当收到治疗费用后，办公人员在系统中更新支付状态(Enterpayment)。
5. 记录医护人员信息(Maintain dental staffinfo)。办公人员将医护人员信息录入系统。医护人员信息包括姓名、职位、身份证号、家庭住址和联系电话等。
6. 医护人员可以查询并打印其参与的治疗项目相关信息(Searchandprintprocedureinfo)。

现采用面向对象方法开发该系统，得到如图 3-1 所示的用例图和 3-2 所示的初始类图。

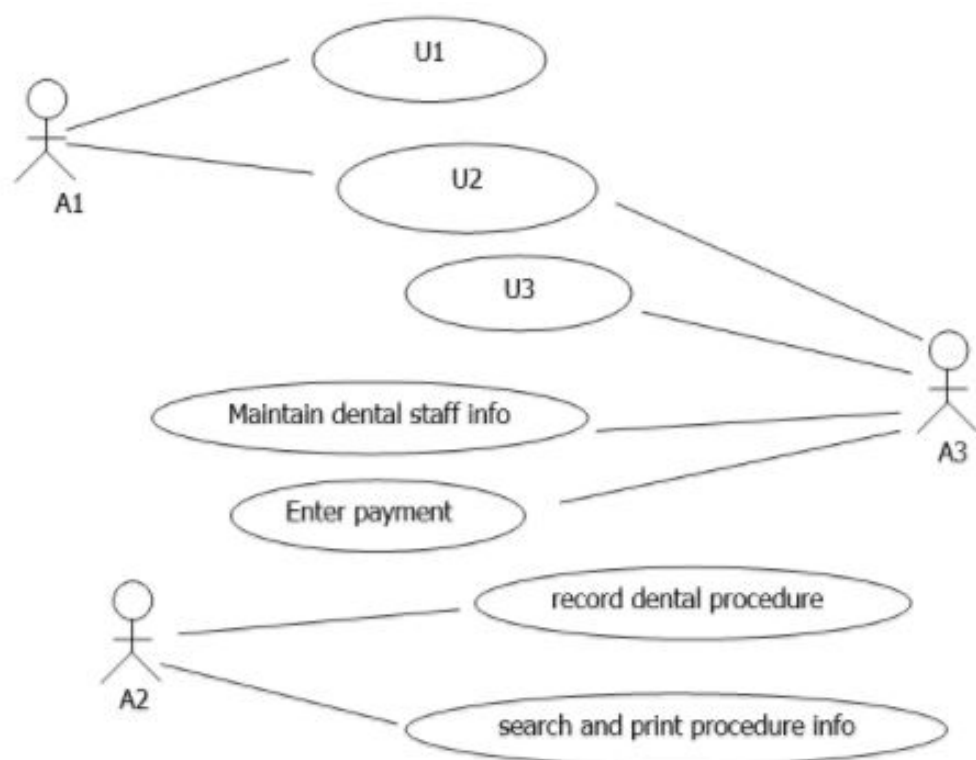


图 3-1 用例图

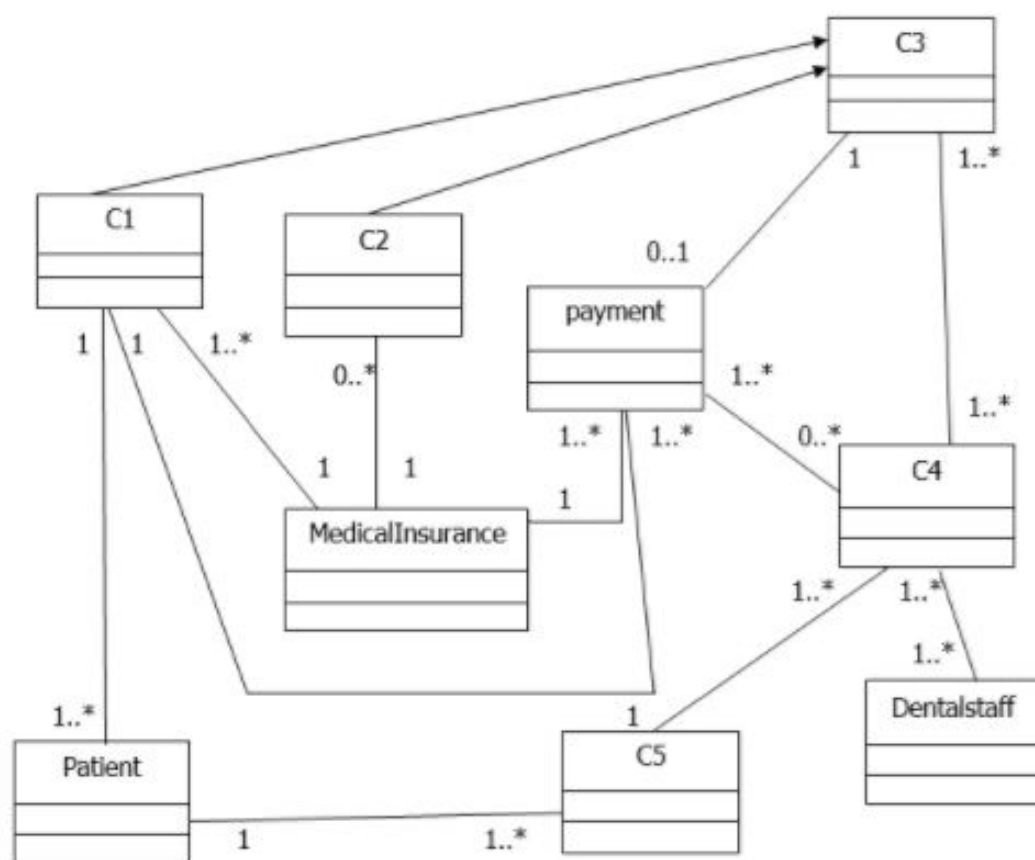


图 3-2 初始类图

**问题： 3.1**

(6 分)

根据说明中的描述，给出图 3-1 中 A1~A3 所对应的参与者名称和 U1~U3 所对应的用例名称。

**问题： 3.2**

(5 分)

根据说明中的描述，给出图 3-2 中 C1~C5 所对应的类名。

**问题： 3.3**

(4 分)

根据说明中的描述，给出图 3-2 中类 C4、C5、Patient 和 DentalStaff 的必要属性。

**试题四 【说明】**

0-1 背包问题定义为:给定  $i$  个物品的价值  $v[1\cdots i]$ 、小重量  $w[1\cdots i]$  和背包容量  $T$ ，每个物品装到背包里或者不装到背包里。求最优的装包方案，使得所得到的价值最大。

0-1 背包问题具有最优子结构性质。定义  $c[i][T]$  为最优装包方案所获得的最大价值，则可以得到如下所示的递归式。

**【c 代码】**

下面是算法的 C 语言实现。

(1) 常量和变量说明

T: 背包容量

$v[]$ : 价值数组

$w[]$ : 重量数组

$c[i][j]$ :  $c[i][j]$  表示前  $i$  个物品在背包容量为  $j$  的情况下最优装包方案所能获得的最大价值

(2) C 程序



本人将不方便阅读的图片梳理成文字

```
#include
#include
#define N6
#define maxT 1000
int c[N][maxT]= {
    0
}
;
int Memoized_Knapsack(int v[N],int w[N],int T) {
    int i;
    int j;
    for (i=0; i
        for (j=0; j<=T; j++) {
            c[i][j]= -1;
        }
    }
    return Calculate_Max_Value(v, w, N-1, T);
}
int Calculate_Max_Value(int v[N],int w[N], int i, int j) {
    int temp =0;
    if (c[i][j]!=-1) {
        (1)
    }
    if (i==0||j==0) {
        c[i][j]=0;
    } else {
        c[i][j]=Calculate_Max_Value(v, w, i-1, j);
        if( (2) ) {
            temp=(3) ;
            if(c[i][j]
                (4)
            }
        }
    }
}
}
return c [i][j];
}
```

$$c[i][T] = \begin{cases} 0 & \text{若 } i=0 \text{ 或 } T=0 \\ c[i-1][T] & \text{若 } T < w[i] \\ \max(c[i-1][T-w[i]]+v[i], c[i-1][T]) & \text{若 } i > 0 \text{ 且 } T \geq w[i] \end{cases}$$

**问题： 4.1**

(8 分)

根据说明和 C 代码，填充 C 代码中的空 (1) ~ (4)。

**问题： 4.2**

(4 分)

根据说明和 C 代码，算法采用了 (5) 设计策略。在求解过程中，采用了 (6) (自底向上或者自顶向下) 的方式。

**问题： 4.3**

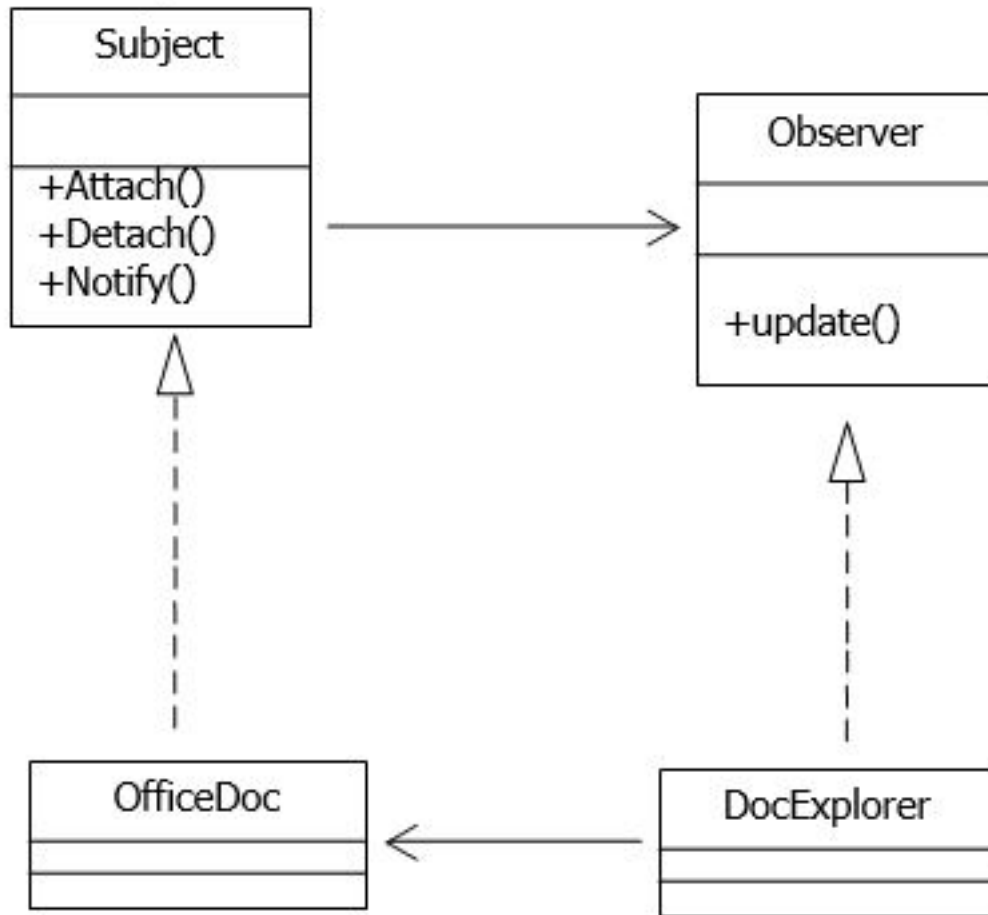
(3 分)

若 5 项物品的价值数组和重量数组分别为  $v[] = \{0, 1, 6, 18, 22, 28\}$  和  $w[] = \{0, 1, 2, 5, 6, 7\}$  背包容量为  $T = 11$ ，则获得的最大价值为 (7)。

**试题五** 阅读下列说明和 Java 代码，将应填入 (n) 处的字句写在答题纸的对应栏内。

**【说明】**

某文件管理系统中定义了类 `OfficeDoe` 和 `DocExplorer`。当类 `OfficeDoe` 发生变化时，类 `DocExplorer` 的所有对象都要更新其自身的状态。现采用观察者 (Observer) 设计模式来实现该需求，所设计的类图如图 6-1 所示。



问题： 5.1

【 Java 代码】

```

import java.util.*;
interface Observer {
    public(1);
}
interface Subject {
    public void Attach(Observer obs);
    public void Detach(Observer obs);
    public void Notify();
    public void setStatus(int status);
    public int getStatus();
}
class OfficeDoc implements Subject {
    private List<(2)> myObs;
    private String mySubjectName;
    private int m_status;
    public OfficeDoc(String name) {
        mySubjectName = name;
    }
}
  
```

```

        this.myObs = new ArrayList();
        m_status = 0;
    }
    public void Attach(Observer obs); {
        this.myObs.add(obs);
    }
    public void Detach(Observer obs); {
        this.myObs.remove(obs);
    }
    public void Notify();
        for(Observer obs; this.myObs) {
            (3);
        }
    }
    public void setStatus(int status;) {
        m_status = status;
        System.out.println("SetStatus Subject[" + mySubjectName +
"]status:" + status);
    }
    public int getStatus() {
        return(m_status);
    }
}
class DocExplorer implements Observer {
    private String myObsName;
    public DocExplorer(String name, (4) sub) {
        myObsName = name;
        sub(5);
    }
    public void update() {
        System.out.println("update observer[" + myObsName + "]);
    }
}
class ObserverTest {
    public static void main(String[] args) {
        System.out.println("Hello World!");
        Observer observerA = new DocExplorer("observerA",
subjectA);
        subjectA.setStatus(1);
        subjectA.Notify();
    }
}

```

**试题一 答案： 解析：**

**答案**

E1：帮买顾问； E2：车辆交易系统； E3：物流商。

本题属于常规题型，补充数据流图中的实体名，实体一般为人员、组织机构、第三方系统等。

根据题干描述，“帮买顾问看到有买车线索后，…”可知接收交易线索的 E1 对应实体应该是帮买顾问；根据题干描述“将车辆交易系统的交易信息抓取为线索”可知提供车辆交易信息的 E2 对应实体应该是车辆交易信息；根据题干描述“物流商注册：物流商账号的注册开通。”可知提供物流商注册信息的 E3 对应实体应该是物流商。

**答案**

D1：线索信息表/线索信息存储； D2：订单信息表/订单信息存储；

D3：路线信息表/路线信息存储； D4：合约信息表/合约信息存储；

D5：物流商信息表。

本题属于常规题型，补充数据流图中的数据存储名，一般对应\*\*库、\*\*表、\*\*档案等。

根据题干描述和图示 P1 抓取线索后交易线索数据流入并且有线索数据流出的数据存储，应该是线索存储，即 D1 为线索信息表(线索信息存储等)。根据题干描述“帮买顾问将这个线索发起为订单并在系统中存储”和图示 P2 发起订单，新订单数据流入的数据存储 D2，应该是订单信息表。根据题干描述和图示 P3 路线管理，有路线数据流入的数据存储 D3 应该是路线信息表。根据题干描述和图示 P4 合约管理，有合约数据流入的数据存储 D4 应该是合约信息表。根据题干描述和图示 P5、P6，有新物流商数据流入、物流商信息数据流出的数据存储 D5，应该是物流商信息表。

**答案**

本题属于常规题型，补充缺失的数据流及其起点和终点。

1、根据父图-子图平衡原则，父图存在 E2->二手车物流系统的物流信息数据流，子图不存在，又根据题干描述：寻找物流商：…，给车辆交易系统发送物流信息，所以图 1-2 缺失数据流 p5->E2，物流信息。

2、根据题干描述“系统根据订单的类型(保卖车、全国购和普通二手车)、起止城市、…选择合适的物流商”， P5 寻找物流商缺失订单信息流入。

3、根据题干描述“若符合固定路线和/或包车路线，系统自动分配给合约物流商，若不符合…”， P5 寻找物流商需要依据固定路线和/或包车路线，还需要根据合约的有效期等信息判断是否可以分配，所以需要补充数据流路线信息： D3->P5，合约信息： D4->P5。

**答案**

本题属于扩展题型。结构化语言是一种介于自然语言和形式化语言之间的半形式化语言，是自然语言的一个受限子集。结构化语言没有严格的语法，通常分为内层和外层。外层有

严格的语法，内层接近于自然语言的描述。

外层用来描述控制结构，采用顺序、选择和重复 3 种基本结构。顺序结构是一组祈使句、选择语句、重复语句的顺序排列。选择结构一般用 IF-THEN-ENDIF 等关键字。重复结构一般用 DO-WHILE-ENDDO 等关键字。内层一般使用祈使句的自然语言短语。

数据名	起点	终点
物流信息	P5/寻找物流商	E2/车辆交易系统
新订单信息	D2/订单信息表	P5/寻找物流商
路线信息	D3/路线信息表	P5/寻找物流商
合约信息	D4/合约信息表	P5/寻找物流商

```

寻找物流商 {
    WHILE(有新订单)
    DO{
        IF 订单.类型 IN('保卖车', '全国购')
            THEN 分配订单到竞拍体系;
        ELSE IF 订单.路线符合固定路线或包车路线
            THEN 自动分配订单给合约物流商;
            ELSE 分配订单到竞拍体系;
        ENDIF
    } ENDDO
    给承运的物流商发送信息;
    更新订单的物流信息;
    给车辆交易系统发送物流信息;
}

```

```

竞拍体系 {
    WHILE(收到订单)
    DO{
        推送订单信息给有相关路线的物流商;
        物流商竞拍出价;
        最优报价物流商中标;
    } ENDDO
}

```

或

```

寻找物流商 {
    WHILE(有新订单)
    DO{
        IF 订单.类型=='普通二手车' AND 订单.路线符合固定路线或包车路线
            THEN 自动分配订单给合约物流商;
            ELSE 分配订单到竞拍体系;
        ENDIF
        给承运的物流商发送信息;
        更新订单的物流信息;
        给车辆交易系统发送物流信息;
    } ENDDO
}

```

```

竞拍体系 {
    WHILE(收到订单)
    DO{
        推送订单信息给有相关路线的物流商;
        物流商竞拍出价;
    }
}

```

**试题二 答案： 解析：** (1) a 部门负责人；(b)培训师；(c)新入职员工

(2) 存在缺失联系：员工与部门之间隶属关系，联系类型\*:1。

(或，存在缺失联系：部门与员工之间隶属关系，联系类型 1：\*)。

(本题预估分值 5 分，填空每空 1 分，联系 2 分)

(1) 根据题干描述员工岗位有新入职员工、培训师、部门负责人，所以对于员工的特殊化实体有新入职员工、培训师、部门负责人，又根据图示，(c)与部门之间有负责关系，所以(c)是部门负责人，(b)与课程之间有讲授关系，所以(b)为培训师，(a)与课程之间有培训关系，根据题干描述新入职员工需要选择多门课程进行培训，所以(a)是新入职员工。

(2) 根据题干说明，一个部门有多个员工，但一名员工只属于一个部门，所以员工与部门之间存在隶属关系，并且员工与部门之间联系类型为\*: 1。(或部门与员工之间存在 1: \* 的联系)。

### 答案

(d)岗位，基本工资；(e)课程号；(f)新入职员工/新入职员工工号，课程成绩

(本题预估分值 3 分，每空 1 分)

根据题干说明“员工信息包括员工号、姓名、部门号、岗位、基本工资、电话、家庭住址等”，员工关系缺少属性(d)：岗位，基本工资。

根据题干说明“课程信息包括课程号、课程名称、学时等”，课程关系缺失属性(e)：课程号。

根据题干说明，培训关系是新入职员工与课程之间多对多联系的转换，所以必须包含二者的主键即新入职员工的员工号(员工号唯一标识员工关系中的每一个元组)和课程的课程号(课程号唯一标识课程关系的每一个元组)，又根据说明“新入职员工要选择多门课程进行培训，并通过考试取得课程成绩”，因此培训还需要有自身的属性课程成绩，即培训关系缺失属性(f)：新入职员工/新入职员工工号，课程成绩。

### 答案

(本题预估分值 4 分，每空 1 分)

本题考查对主键和外键的判断，主键可以唯一标识元组，外键是其他关系的主键。

根据题干说明“员工号唯一标识员工关系中的每一个元组”，因此员工关系的主键(g)为员工号，又因为“部门号唯一标识部门关系中的每一个元组”，即部门号是部门关系的主键，在员工关系中，是作为外键(h)的。

根据题干说明和图示可知，讲授关系是培训师与课程之间多对多的联系转换，此时主键应该是二者的主键组合，即主键(i)(培训师，课程号)，又因为培训师是培训师的主键，课程号是课程的主键，所以二者又是讲授关系的外键(j)。本题由于培训地点是否固定并没有给出描述，所以无法判断是否属于主键组合的一部分，因此给出主键(i)(培训师，课程号，培训地点)组合键也可以得分。



## 答案

(本题预估 3 分，判断 1 分，理由 2 分)

存在传递函数依赖。

在员工关系中，员工的岗位有新入职员工，培训师，部门负责人，不同岗位设置不同的基本工资，即存在传递函数依赖，员工号→岗位，岗位→基本工资。

关系模式	主键	外键
员工关系	(g) 员工号	(h) 部门号
讲授关系	(i) (培训师, 课程号)	(j) 课程号、培训师

## 试题三 答案： 解析：

### 答案

A1： Receptionist(接待员) A2： DentalStaff(医护人员) A3： OfficeStaff(办公人员)  
U1： Maintainpatientinfo (记录病人基本信息) U2： Recordofficevisitinfo (记录就诊信息) U3： Printinvoices(打印发票)

本题属于常规考题，考查对参与者和用例名的补充，系统的参与者一般为人员、机构或第三方系统。用例名一般为动词+名词或名词+动词，是对系统功能的概括和描述。

本题根据题干说明，参与者即诊所工作人员包括：医护人员(DentalStaff)、接待员(Receptionist)和办公人员(OfficeStaff)等。

根据用例图已有信息，A2 使用用例 Recorddentalprocedure 和 Searchandprintprocedureinfo，根据题干说明记录治疗信息(Recorddentalprocedure)由每位参与治疗的医护人员分别录入，医护人员可以查询并打印其参与的治疗项目相关信息(Searchandprintprocedureinfo)，因此 A2 为医护人员(DentalStaff)。

根据用例图已有信息 A3 使用用例 Maintai ndentalstaffinfo 和 Enterpayment，根据题干说明记录医护人员信息(Maintai ndentalstaffinfo)，由办公人员录入系统，所以 A3 为 OfficeStaff(办公人员)。并且根据题干办公人员还需要打印发票(Printinvoices)、更新支付状态(Enterpayment)，缺失的 U3 应该是打印发票(Printinvoices)。根据题干描述 A3 与 U2 之间的联系没有意义，不参考。

根据题干描述和图示，A1 对应的参与者应该是 A1： Receptionist(接待员)，接待员需要参与的功能有记录病人基本信息(Maintainpatientinfo)和记录就诊信息(Recordofficevisitinfo)，分别对应 U1、U2，二者与办公人员都没有明确联系，位置可以互换。

### 答案

C1: PatientInvoice C2: InsuranceInvoice C3: Invoice C4: Procedure C5： OfficeVisit

本题属于常规考查题型，补充缺失的类名，常见的实体类类名一般为名词形式，也会有特殊的边界类/接口类和协调类。需要参照题干描述和类图中类与类之间的关系，来确定缺失的类名及其位置。

根据初始类图，存在一组泛化关系，C3 是 C1、C2 的泛化，即 C3 是 C1、C2 的父类，根据题干描述存在这样泛化关系的只有发票(Invoice)、病人发票(PatientInvoice)、医保机构发票(Insurancelnvoice)，又根据多重度来分析，由于可能存在全自费的情况，即医保发票不存在，所以多重度 0...\*对应的 C2 类名应该是医保机构发票(Insurancelnvoice)，C1 对应的是病人发票(PatientInvoice)，父类 C3 对应的是发票(Invoice)。

又根据图示和题干，与医护人员 (DentalStaff) 相关的用例有记录治疗信息(Record dental procedure)、查询并打印其参与的治疗项目相关信息(Search and print procedure info)，其中能够找到相关内容治疗信息 procedure，即 C4 对应的是实体类治疗信息 Procedure，与治疗相关的是就诊信息 office visit，即 C5 对应的是实体类就诊信息 OfficeVisit。

### 答案

C4：治疗项目名称、治疗项目描述、治疗的牙齿和费用、OfficeVisit、DentalStaff

C5：就诊信息、病人就诊时间和费用、支付代码、病人支付费用、医保支付费用、病人 patient

Patient：姓名、身份证号、出生日期、性别等。

DentalStaff：姓名、职位、身份证号、住址、联系电话、职工号。

根据题干描述，“病人基本信息包括病人姓名、身份证号、出生日期、性别、首次就诊时间和最后一次就诊时间等”，因此 Patient 的必要属性包括病人姓名、身份证号、出生日期、性别、首次就诊时间和最后一次就诊时间。

根据题干描述，“医护人员信息包括姓名、职位、身份证号、家庭住址和联系电话等”，因此 DentalStaff 的必要属性包括姓名、职位、身份证号、家庭住址和联系电话等。

根据题干描述，C5 OfficeVisit 就诊信息包括就诊时间、就诊费用、支付代码、病人支付费用和医保支付费用等，又因为病人与就诊 1...\*联系起来，所以还需要补充属性病人 patient。

根据题干描述 C4: Procedure 治疗信息包括：治疗项目名称、治疗项目描述、治疗的牙齿和费用等，又因为治疗信息与医护人员存在 1...\*对应关系，与 C5 就诊信息存在 1...\*对应关系，所以这里还需要补充医护人员 DentalStaff、就诊信息 OfficeVisit。

试题四 答案： 解析：

答案

- (1) `c[i][j]`
- (2) `i>0&&j<=w[i]`
- (3) `Calculate_Max_Value(v, w, i-1, j-w[i])+v[i]`
- (4) `c[i][j]=temp`

结合题干描述中的递归式，函数最终返回值应该是 `c[i][T]`，又根据代码上下文，`T` 在函数中传参为 `j`，所以第(1)空返回的结果为 `c[i][j]`；

根据递归式和代码上下文可知，`c[i][j]=0` 已处理，`if` 后面处理的是 `max` 递归部分，又根据上下文可以看到 `temp` 会与 `c[i][j]` 进行比较，递归时 `i=i-1`，所以这里判断的是 `temp` 与 `c[i-1][T]` 的值，第(3)空的处理是将 `c[i-1][j-w[i]]+v[i]` 传递给 `temp`，即 `temp=Calculate_Max_Value(v, w, i-1, j-w[i])+v[i]`，注意这里不是直接穿 `c[][]` 数组值，而是递归调用 `Calculate_Max_Value()` 函数。那么第(2)空缺失的判断条件根据第3个表达式条件为 `i>0` 且 `T<=w[i]`，对应代码中的参数即 `i<0&&j<=w[i]`。

第(4)空是 `c[i][j]<temp` 比较后的返回值，根据题干，我们需要返回 `max` 最大值，最终返回的结果是 `c[i][j]`，因此要保证 `c[i][j]` 最大，当 `c[i][j]<temp` 时，将 `temp` 赋值给 `c[i][j]`，即 `c[i][j]=temp`。

#### 答案

- (5) 动态规划
- (6) 自顶向下

本题采用递归形式，并且求取的是全局最优解，中间结果存在二维数组 `c[][]` 当中，所以采用的是动态规划法。动态规划法采用递归形式，`i` 会从 `N-1` 递减至 `0`，所以是自顶向下的方式。

#### 答案

- (7) 40

本题考查最优解，可以忽略题干描述，直接凑数，可得第3和第4个物品，重量为11，价值为40，此时为最优解，即为最大价值。

#### 试题五 答案： 解析：

#### 答案

- (1) `void update()` ;
- (2) `Observer;`
- (3) `obs.update()` ;
- (4) `Subject;`

(5) Attach(this);

本题是对观察者模式的考查，观察者模式的意图是：定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

本题根据 Observer 接口的实现类 DocExplorer，实现类包括同名构造函数和 update() 方法，所以接口 Observer 缺失的是 update() 方法，并且没有方法体，第(1)空填写 void update()。

第2空是 myObs 表单类型的缺失，根据代码上下文，在构造函数中，根据 this.myObs=new ArrayList()，可以知道 myObs 是 Observer 表单，第(2)空填写 Observer。

第3空是 Notify() 方法体的缺失，根据代码上下文，Notify 传入了一个参数 Observer obs，又根据代码上下文可知 Observer 只有一个 update() 方法，此时方法体调用的应该是 update() 方法，调用方法的对应是传入的 obs，第(3)空填写 obs.update()。

第4空、第5空缺失的是 DocExplorer 类的同名构造函数传入的参数类型以及构造方法体，这里结合根据观察者模式填空，对于实际观察者类，需要与被观察者联系起来，所以这里是与被观察者 Subject 联系，也就是调用 Subject 中的 Attach() 添加观察者列表。因此第(4)空需要填写参数类型 Subject，形参名 sub 已经给出了提示；sub 调用增加观察者方法，将当前观察者添加到对应列表，即第(5)空填写 Attach(this)。



苹果 扫码或应用市场搜索“软考真题”下载获取更多试卷



安卓 扫码或应用市场搜索“软考  
真题”下载获取更多试卷