

全国计算机技术与软件专业技术资格（水平）考试

2013 年上半年 软件设计师 下午试卷

（考试时间 14:00～16:30 共 150 分钟）

请按下述要求正确填写答题纸

- 1.在答题纸的指定位置填写你所在的省、自治区、直辖市、计划单列市的名称。
- 2.在答题纸的指定位置填写准考证号、出生年月日和姓名。
- 3.答题纸上除填写上述内容外只能写解答。
- 4.本试卷共 7 道题，试题一至试题四是必答题，试题五至试题六选答 1 道。每题 15 分，满分 75 分。
- 5.解答时字迹务必清楚，字迹不清时，将不评分。
- 6.仿照下面例题，将解答写在答题纸的对应栏内。

例题

2013 年上半年全国计算机技术与软件专业技术资格（水平）考试日期是（1）月（2）日。

因为正确的解答是“5 月 20 日”，故在答题纸的对应栏内写上“5”和“20”（参看下表）。

例题	解答栏
（1）	5
（2）	20

试题一至试题四是必答题

试题一

某慈善机构欲开发一个募捐系统,以跟踪记录为事业或项目向目标群体进行募捐而组织的集体性活动,该系统的主要功能如下所述。

(1) 管理志愿者。根据募捐任务给志愿者发送加入邀请、邀请跟进、工作任务;管理志愿者提供的邀请响应、支援站信息、工作时长、工作结果等。

(2) 确定募捐需求和收集所募捐赠(资金及物品)。根据需求提出的募捐任务、活动请求和募捐请求,获取所募集的资金和物品。

(3) 组织募捐活动。根据活动请求,确定活动时间范围。根据活动时间,搜索场馆,即向场馆发送场馆可用性请求,获得场馆可用性。然后根据活动时间和地点推广募捐活动,根据相应的活动信息举办活动,从募款机构获取资金并向其发放赠品。获取和处理捐赠,根据捐赠要求,提供所募集的捐赠;处理与处理人之间的交互。即:

录入捐赠人信息,处理后存入捐赠人信息表;从捐赠人信息表中查询捐赠人信息,向捐赠人发送捐赠请求,并将已联系的捐赠人存入已联系的捐赠人表。根据捐赠请求进行募集,募得捐赠后,将捐赠记录存入捐赠表;对捐赠记录进行处理后,存入已处理捐赠表,向捐赠人发送致谢函。根据已联系的捐赠人和捐赠记录进行跟进,将捐赠跟进情况发送给捐赠人。

现采用机构化方法对募捐系统进行分析与设计,获得如图 1-1、1-2 和 1-3 所示分层数据流图。

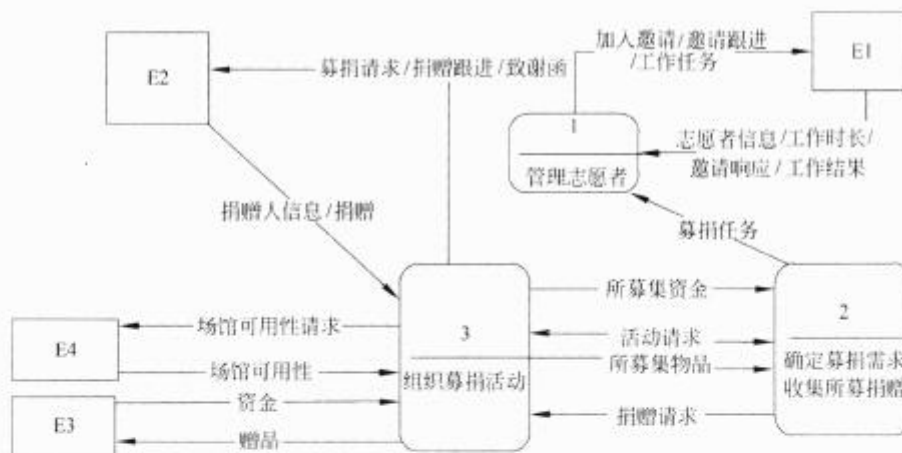


图 1-1 0 层数据流图

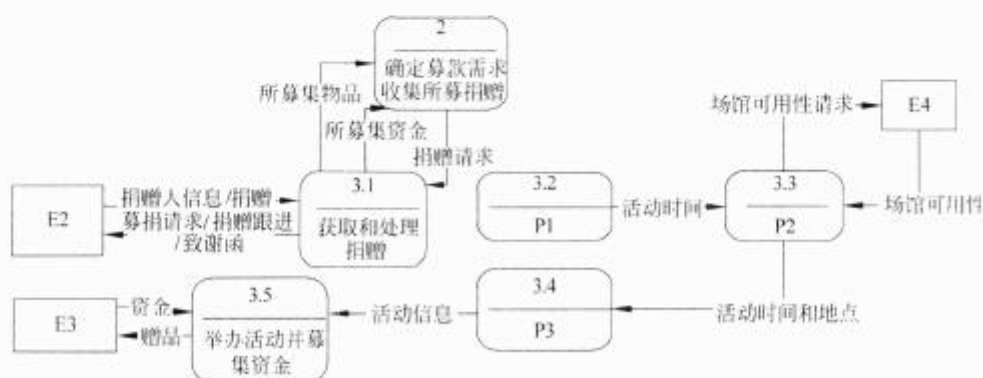


图 1-2 1 层数据流图

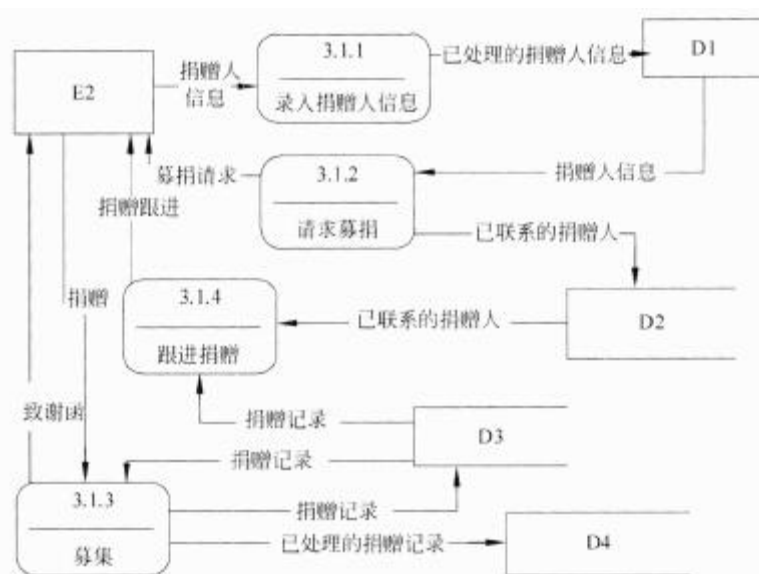


图 1-3 2 层数据流图

【问题 1】

使用说明中的词语，给出图 1-1 中的实体 E1~E4 的名称。

【问题 2】

在建模 DFD 时，需要对有些复杂加工（处理）进行进一步精化，图 1-2 为图 1-1 中处理 3 的进一步细化的 1 层数据图，图 1-3 为图 1-2 中 3.1 进一步细化的 2 层数据流图。补全图 1-2 中加工 P1、P2 和 P3 的名称和图 1-2 与图 1-3 中缺少的数据流。

【问题 3】

使用说明中的词语，给出图 1-3 中的数据存储 D1~D4 的名称。

试题二

【说明】

某电视台拟开发一套信息管理系统，以方便对全台的员工、栏目、广告和演播厅等进行管理。

【需求分析】

(1) 系统需要维护全台员工的详细信息、栏目信息、广告信息和演播厅信息等。员工的信息包括：工号、姓名、性别、出生日期、电话、住址等。栏目信息主要包括：栏目名称、播出时间、时长等。广告信息主要包括：广告编号、价格等。演播厅信息包括：房间号、房间面积等。

(2) 电视台根据调度单来协调各个栏目、演播厅和场务。一档栏目只会占用一个演播厅，但会使用多名场务来进行演出协调。演播厅和场务可以被多个栏目循环使用。

(3) 电视台根据栏目来插播广告。每档栏目可以插播多条广告，每条广告也可以在多档栏目插播。

(4) 一档栏目可以有多个主持人，但一名主持人只能主持一档栏目。

(5) 一名编辑人员可以编辑多条广告，一条广告只能由一名编辑人员编辑。

【概念模型设计】

根据需求阶段收集的信息设计的实体联系图（不完整）如图 2-1 所示。

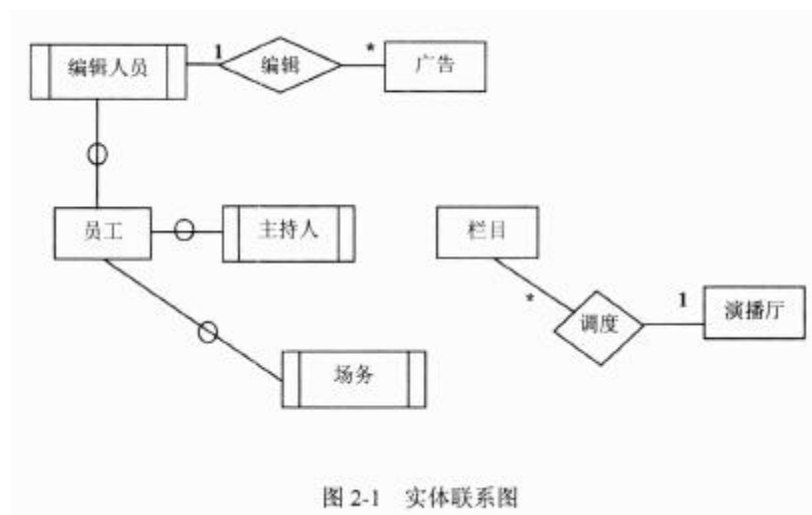


图 2-1 实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

演播厅（房间号，房间面积）

栏目（栏目名称，播出时间，时长）

广告（广告编号，销售价格，（1））

员工（工号，姓名，性别，出生日期，电话，住址）

主持人（主持人工号，（2））

插播单（（3），播出时间）

调度单（（4））

【问题 1】

补充图 2-1 中的联系和联系的类型。

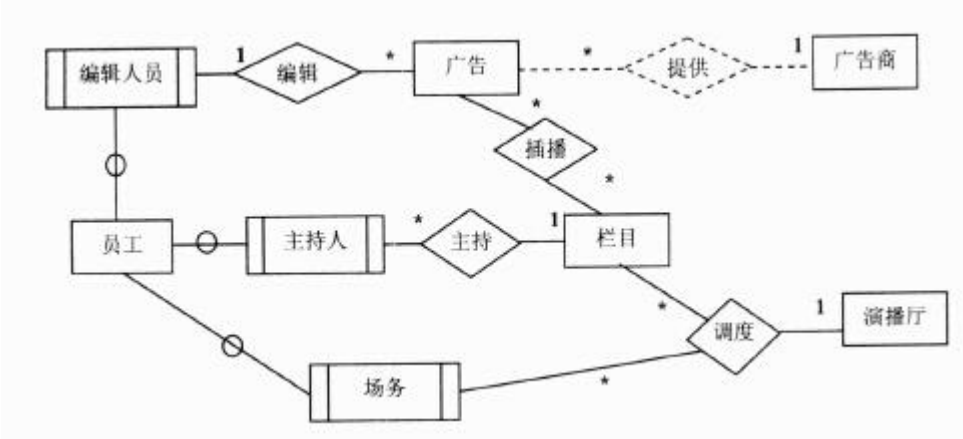
【问题 2】

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）~（4）补充完整，并用下划线指出空（1）~（4）所在关系模式的主键。

【问题 3】

现需要记录广告商信息，增加广告商实体。一个广告商可以提供多条广告，一条广告只由一个广告商提供。请根据该要求，对图 2-1 进行修改，画出修改后的实体间联系和联系的类型。

试题三



【说明】

某城市拟开发一个基于 Web 的城市黄页，公开发布该城市重要的组织或机构（以下统称为客户）的基本信息，方便城市生活。该系统的主要功能描述如下：

- (1) 搜索信息：任何使用 Internet 的网络用户都可以搜索发布在城市黄页中的信息，例如客户的名称、地址、联系电话等。
- (2) 认证：客户若想在城市黄页上发布信息，需要通过系统的认证。认证成功后，该客户成为系统授权用户。
- (3) 更新信息：授权用户登录系统之后，可以更改自己的在城市黄页中的相关信息，例如变更联系电话等。
- (4) 删除客户：对于拒绝继续在城市黄页上发布信息的客户，由系统管理员删除该客户的相关信息。

系统采用面向对象方法进行开发，在开发过程中认定出如表 3-1 所示的类。系统的用例图和类图分别如图 3-1 和 3-2 所示。

表 3-1 类列表	
类名	说明
InternetClient	网络用户
CustomerList	客户集，维护城市黄页上的所有客户信息
Customer	客户信息，记录单个客户的信息
RegisteredClient	授权用户
Administrator	系统管理员

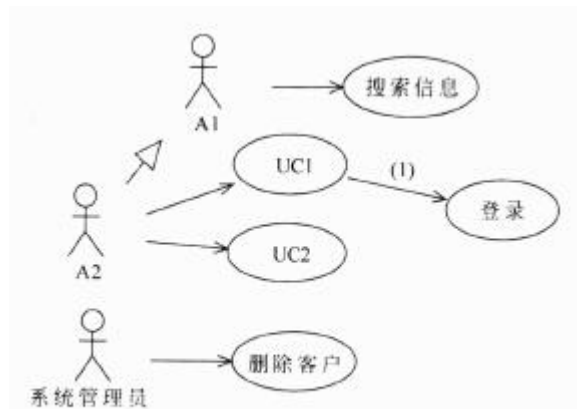


图 3-1 系统用例图

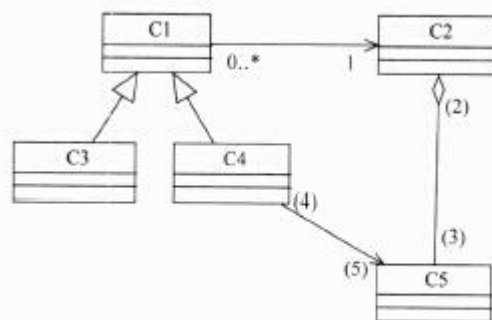


图 3-2 系统类图

【问题 1】

根据说明中的描述，给出图 3-1 中的 A1 和 A2 处所对应的参与者、UC1 和 UC2 处所对应的用例以及（1）处的关系。

【问题 2】

根据说明中的描述，给出图 3-2 中的 C1~C5 所对应的类名（表 3-1 中给出的类名）和（2）~（5）处所对应的多重度。

【问题 3】

认定类是面向对象分析中非常关键的一个步骤。一般首先从问题域中得到候选类集合，再根据相应的原则从该集合中删除不作为类的，剩余的就是从问题域中认定出来的类。简要说明选择题选类的原则，以及对候选类集合进行删除的原则。

试题四

阅读下列说明和 C 代码，回答问题 1 至问题 3, 将解答写在答题纸的对应栏内。

【说明】

设有 m 台完全相同的机器运行 n 个独立的任务，运行任务 i 所需要的时间为 t_i ，要求确定一个调度方案，使得完成所有任务所需要的时间最短。

假设任务已经按照其运行时间从大到小顺序。算法基于最长运行时间作业优先的策略；按顺序先把每个任务分配到一台机器上，然后将剩余的任务依次放入空闲的机器。

【C 代码】

下面是算法的 C 语言实现。

(1) 常量和变量说明

m : 机器数

n : 任务数

$t[]$: 输入数组，长度为 n ，其中每个元素表示任务的运行时间，下标从 0 开始

$s[][]$: 二维数组，长度为 $m \times n$ ，下标从 0 开始，其中元素 $s[i][j]$ 表示机器 i 运行的任务 j 的编号

$d[]$: 数组，长度为 m ，其中元素 $d[i]$ 表示机器 i 的运行时间，下标从 0 开始

$count[]$: 数组，长度为 m ，下标从 0 开始，其中元素 $count[i]$ 表示机器 i 的运行任务数

i : 循环变量

j : 循环变量

k : 临时变量

max : 完成所有任务的时间

min : 临时变量

(2) 函数 `schedule`

```
void schedule() {  
    int i, j, k, max=0  
    for(i=0; i<m; i++) {  
        d[i]=0;  
        for(j=0; j<n; j++) {  
            s[i][j]=0;  
        }  
    }
```

```

    }

    for(i=0;i<m;i++) {                                //分配前 m 个任务

        s[i][0]=j;

        (1) ;

        count[i]=1;

    }

    for( (2) 、 i<n,i++) {                            //分配后 n-m 个任务

        int min=d[0];

        k=0;

        for(j=1;j<m;j++) {                            //确定空闲机器

            if(min>d[j]) {

                min=d[j];

                k=j;                                    // 机器 K 空闲

            }

        }

        (3) ;

        count[k]=count[k]+1;

        d[k]=d[k]+t[i];

    }

    for(i=0;i<m;i++) {                                //确定完成所有任务需要的时间

        if( (4) ) {

            max=d[i];

        }

    }

}

```

【问题 1】

根据说明和 c 代码，填充 C 代码中的空 (1) ~ (4)。

【问题 2】

根据说明和 C 代码，该问题采用了 (5) 算法设计策略，时间复杂度为

(6) (用 0 符号表示)。

【问题 3】

考虑实例 $m=3$ (编号 $0 \sim 2$), $n=7$ (编号 $0 \sim 6$), 各任务的运行时间为 $\{16, 14, 6, 5, 4, 3, 2\}$ 。
则在机器 0、1 和 2 上运行的任务分别为 (7)、(8) 和 (9) (给出任务编号)。从任务开始运行到完成所需要的时间为 (10) 。

从下列的 2 道试题（试题五至试题六）中任选 1 道解答。
如果解答的试题数超过 1 道，则题号小的 1 道解答有效。

试题五

【说明】

现要求实现一个能够自动生成求职简历的程序。简历的基本内容包括求职者的姓名、性别、年龄及工作经历等。希望每份简历中的工作经历有所不同，并尽量减少程序中的重复代码。

现采用原型（Prototype）模式来实现上述要求，得到如图 5-1 所示的类图。

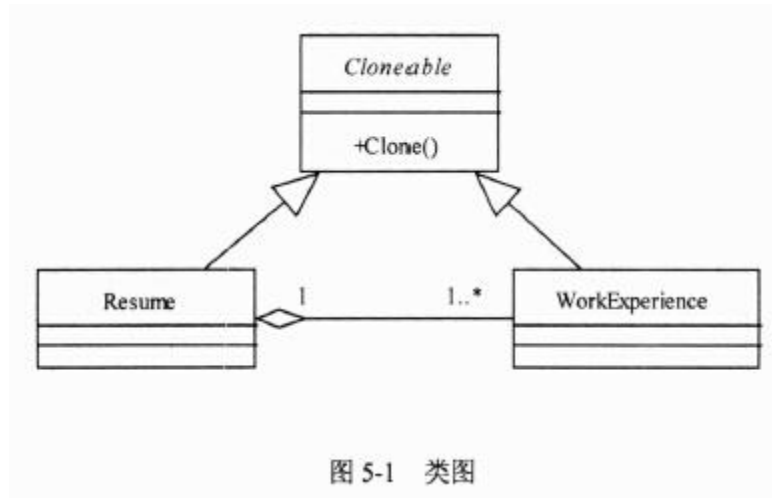


图 5-1 类图

【C++代码】

```
#include<string>

using namespace std;

class Cloneable {
public;

    (1)          ;

};

class WorkExperience:public Cloneable {    //工作经历
private;

    string workDate;

    string company;

public;

    Cloneable* Clone() {
```

```

        (2)                ;

    obj->workDate=this->workDate;

    obj->company=this->company;

    return obj;

    }

    //其余代码省略

};

class Resume:public Cloneable {           //简历

private;

    string name;    string sex;    string age;

    WorkExperience* work;

    Resume(WorkExperience* work) {

        this->work=                (3)                ;

    }

public;

    Resume(string name){    /* 实现略 */    }

    void SetPersonallInfo(string sex, string age){                /* 实现略 */

    /*                }

    void SetWorkExperience(string workDate, string company){    /* 实现略 */

    /*    }

    Cloneable* Clone() {

        (4)                ;

    obj->name=this->name;

    obj->sex=this->sex;

    obj->age=this->age;

    return obj;

    }

};

int main() {

    Resume *a=new Resume(“张三”);

```

```
a->SetPersonalInfo(“男”, “29”);  
a->SetWorkExperience(“1998~2000”, “XXX 公司”);  
Resume *b=          (5)          ;  
b->SetWorkExperience(“2001~2006”, “YYY 公司”);  
return 0;  
}
```

【问题 1】

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

试题六

【说明】

现要求实现一个能够自动生成求职简历的程序。简历的基本内容包括求职者的姓名、性别、年龄及工作经历等。希望每份简历中的工作经历有所不同，并尽量减少程序中的重复代码。

现采用原型（Prototype）模式来实现上述要求，得到如图 6-1 所示的类图。

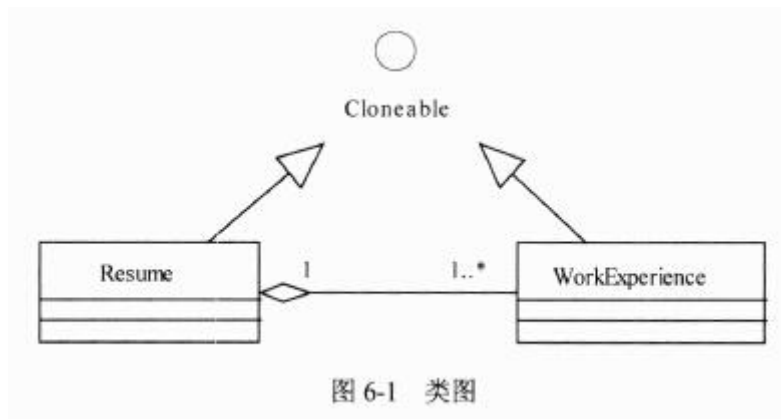


图 6-1 类图

【Java 代码】

```
class WorkExperience          (1)          Cloneable {          //工作经历

    private string workDate;

    private string company;

    public Object Clone() {

                                (2)                                ;

        obj.workDate=this.workDate;

        obj.company=this.company;

        return obj;

    }

}

class Resume                  (3)          Cloneable {          //简历

    private string name;

    private string sex;

    private string age;
```

```

private WorkExperience work;

public Resume(string name) {
    this.name=name;          work=new SetWorkExperience();
}

private Resume(WorkExperience work) {
    this.work=          (4)          ;
}

public void SetPersonallInfo(string sex, string age;) {    /* 代码略 */    }
public void SetWorkExperience(string workDate, string company) {    /* 代码
略 */    }

public Object Clone() {
    Resume obj=          (5)          ;
    // 其余代码省略
    return obj;
}
}

class WorkResume {
    public static void main(string[]arg) {
        Resume a=new Resume(“张三”);
        a.SetPersonallInfo(“男”, “29”);
        a.SetWorkExperience(“1998~2000”, “XXX 公司”);
        Resume b=          (6)          ;
        b.SetWorkExperience(“2001~2006”, “YYY 公司”);
    }
}

```

【问题 1】

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。