

# 2013年上半年软件设计师考试下午真题（权威解析+标准答案）

卷面总分：75.0 分

答题时间：150 分钟

测试次数：21381 次

平均得分：42.2 分

是否需要批改：否

## 案例分析题

在下列各题中，请阅读说明材料，根据提问进行解答。

- 1 某慈善机构欲开发一个募捐系统，已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。

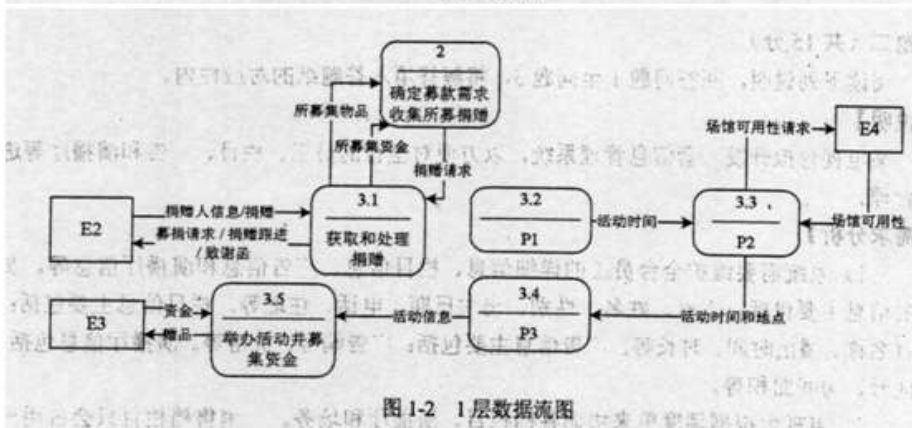
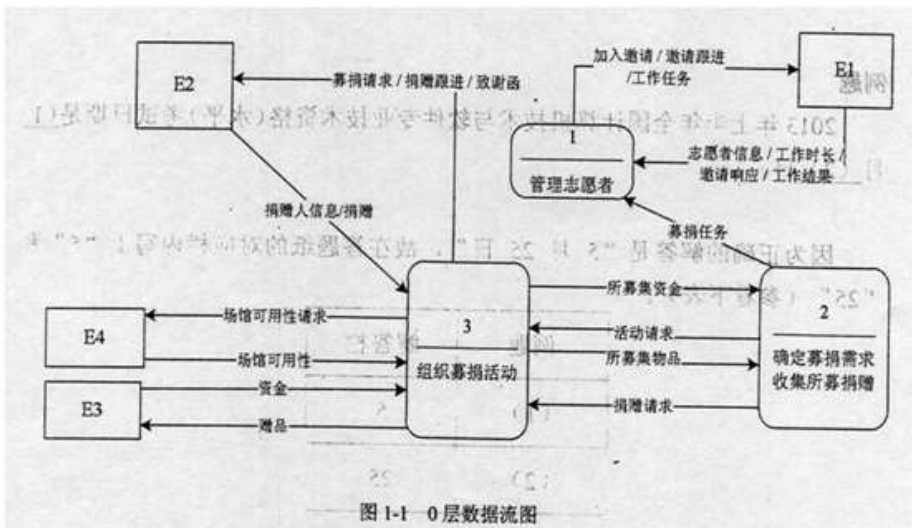
管理志愿者。根据募捐任务给志愿者发送加入邀请、邀请跟进、工作任务；管理志愿者提供的邀请响应、志愿者信息、工作时长、工作结果等。

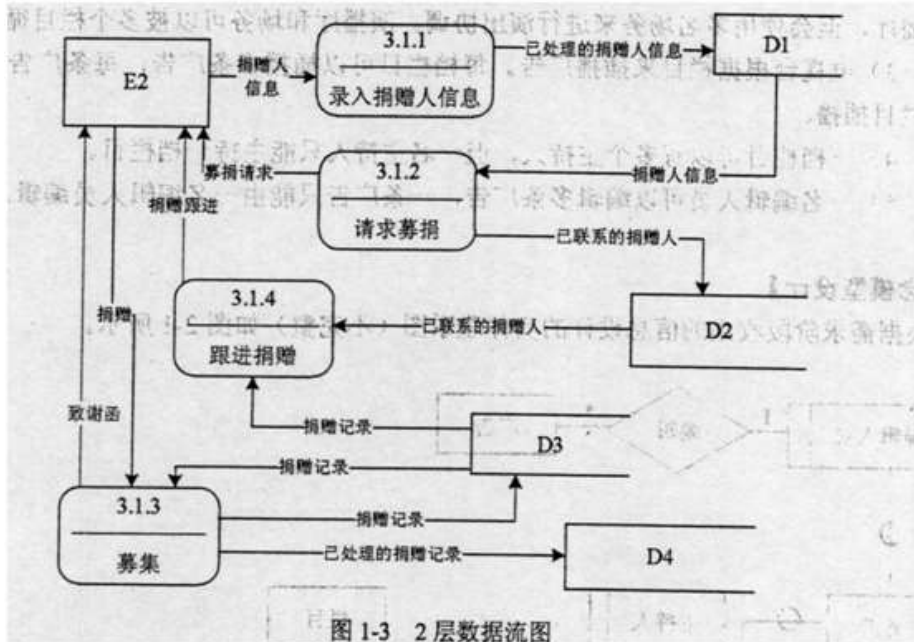
确定募捐需求和收集所募捐赠（资金及物品）。根据需求提出募捐任务、将活动请求和捐赠请求，获取所募集的资金和物品。

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，从募捐机构获取资金并向其发放赠品。获取和处理捐赠，根据捐赠请求，提供所募集的捐赠；处理与捐赠人之间的交互，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；对捐赠记录进行记录后，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踪，并将捐赠跟进情况发送给捐赠人。

先采用结构化方法对募捐系统进行分析和设计，获得如图1-1、图1-2和图1-3所示分层数据流图。





【问题1】（4分）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】（7分）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层数据流图，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】（4分）使用说明中的词语，给出图1-3中的数据存储D1~D4的名称。

填写我的答案

段落格式

代码语言

元素路径:

2 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、广告和演播室等进行管理。

【需求分析】

（12）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。员工的信息主要包括：工号、姓名、性别、出生日期、电话、住址等。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。

（13）电视台分局调度单来协调各档栏目、演播厅和场务。一销售档栏目只会占用一个演播厅，但会使用多么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。

（14）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的栏目插播。

（15）一档栏目可以有多个主持人，但一名主持人只能支持一档节目。

（16）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。

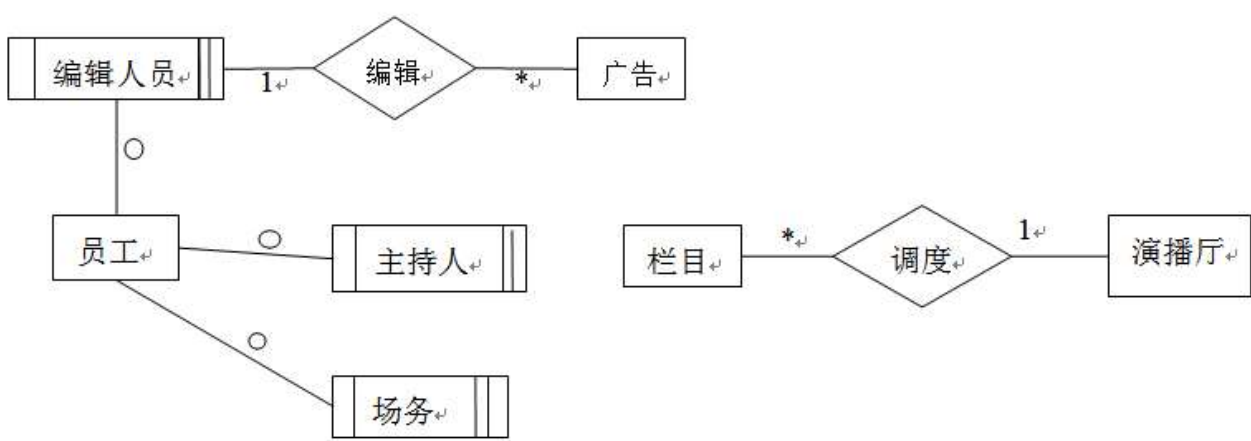


图 2-1 实体联系图

### 【逻辑结构设计】

根据概念模式设计阶段完成的实体联系图，得出如下关系模型（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) )

演播单 ( (3) , 播出时间 )

调度单 ( (4) )

【问题1】（7分）

补充图2-1中的联系和联系类型。

【问题2】（5分）

根据图2-1，将逻辑结构设计阶段生产的关系模型的空(1)~(4)补充完整，并用下划线指出(1)~(4)所在关系模型的主键。

【问题3】（3分）

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。

### 填写我的答案

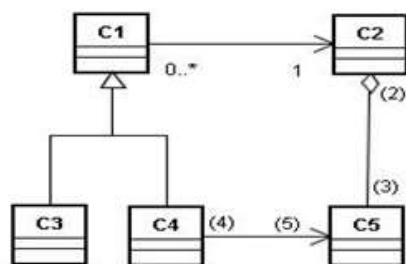
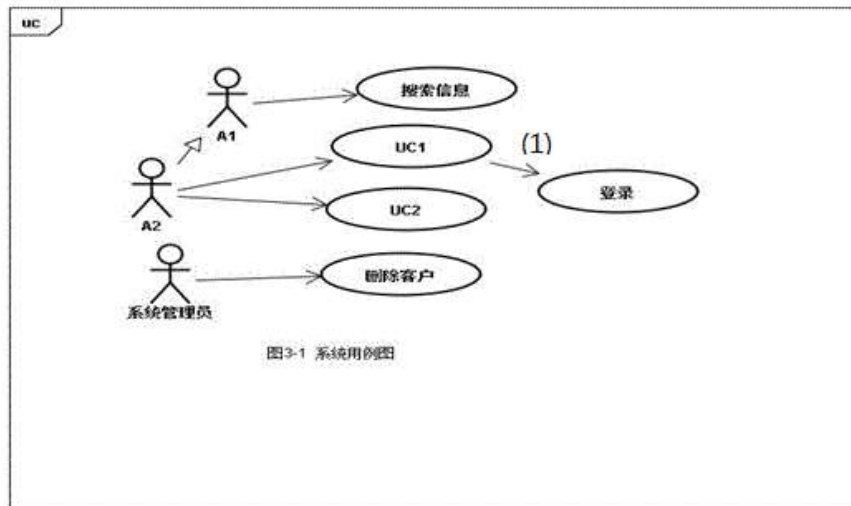
段落格式

代码语言

元素路径:

表 3-1 类列表

类名	说明
InternetClient	网络用户
CustomerList	客户集, 维护城市网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



【问题1】（5分）

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

【问题2】（7分）

根据说明中的描述，给出图3-2中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】（3分）

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，在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。简要说明选择候选类的原则，以及对候选类集合进行删除的原则。

填写我的答案

段落格式

代码语言

元素路径:

假设任务已经按照其运行时间从大到小排序，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，然后将剩余的任务一次放入最先空闲的机器。

【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，下标从0开始，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\*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，下标从0开始，其中元素count[i]表示机器i运行的任务数。

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]=i;
    (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++){ //确定空闲时间
        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++){ //确定完成所有任务所需要的时间
    if( (4) ){
        max=d[i];
    }
}
}
```

【问题1】（8分）

根据说明和C代码，填充C代码中的空（1）~（4）。

【问题2】（2分）

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 （6）（用O符号表示）

【问题3】（5分）

考虑实例m=3（编号0~2），n7（编号0~6），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的时间为 ( 10 )。

填写我的答案

段落格式 代码语言

元素路径:

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

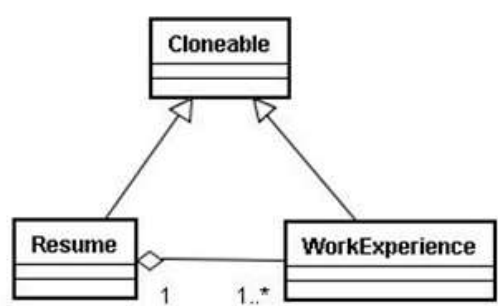


图5-1 类图

【C++代码】

```
#include<string>
using namespace std;
class Cloneable{
public:
    (1) ;
};
class workExperience:public Cloneable{ //工作经历
private:
    string workData;
    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){/*实现省略*/}
    void SetPersonInfo(string sex,string age){/*实现省略*/}
    void SetWorkExperience(string workDate,string company){/*实现省略*/}
    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->SetPersonInfo( "男" , "29" );
    a-> SetWorkExperience( "1998-2000" , "XXX公司" );
    Resume * b =    (5) ;
    b-> SetWorkExperience( "2001-2006" , "YYY公司" );
    return 0;
}
```

填写我的答案

段落格式 代码语言

元素路径:

6 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。希望每份简历中的工作经历有所不同，并尽量减少序中的重复代码。  
现采用原型模式（Prototype）来实现上述要求，得到如图5-1所示的类图。

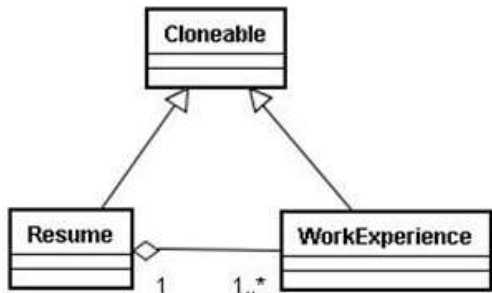


图5-1 类图

【Java代码】

```
public class workExperience (1) Cloneable{ //工作经历
    private String workData;
    private String company;
```

```
public Object clone(){
    (2) ;
    Obj.workDate= this.workDate;
    Obj.company = this.company;
    return Obj;
}
//其余代码省略
}
public 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 WorkExperience();
}
private Resume(WorkExperience work){
    this.work = (4) ;
}
public void SetPersonInfo(string sex,string age){/*实现省略*/}
public void SetWorkExperience(string workDate,string company){/*实现省略*/}
public Object clone(){
    Resume Obj = (5) ;
    return Obj;
}
}
```

```
Class WorkResume{
public static void main(){
    Resume a = new Resume( “张三” );
    a.SetPersonInfo( “男” , “29” );
    a.SetWorkExperience( “1998-2000” , “XXX公司” );
    Resume b = (6) ;
    b. SetWorkExperience( “2001-2006” , “YYY公司” );
}
}
```

填写我的答案

段落格式

代码语言

元素路径: