

考试科目: 高级软件工程

得分: _____

学生所在系: _____ 姓名: _____ 学号: _____

一、判断题 (每题 2 分, 共 10 分)

- ✓ 1. 某个使用管道过滤器结构的系统中, 如果 Data source 和 Data sink 都是被动的, 则至少需要一个主动过滤器。 ()
- ✓ 2. 模型-视图-控制器模式中, 不同的视图对象之间不存在直接的函数调用。 ()
- ✓ 3. 隐式调用风格中往往还需要使用显式调用作为补充。 ()
- ✗ 4. 原型 (Prototype) 模式中, 所有原形对象具有统一的初始化 Initialize 操作接口。 ()
- ✗ 5. 享元 (Flyweight) 模式中, 客户代码可以直接创建非共享对象。 ()

二、选择题 (每题 2 分, 共 30 分)

- D 1. _____ 不能作为管道-过滤器体系结构中的出错处理方案。
A. 单独的错误输出管道
B. 忽略输入数据直到分割符出现
C. 重启流水线
D. 重启出错的过滤器
- C 2. “表示-抽象-控制”模式中, Agent 内可以和其他 Agent 进行交互的组件是 _____
A. 表示 Presentation
B. 抽象 Abstract
C. 控制 Control
D. 三者皆可
- B 3. _____ 不属于代理者 (Broker) 体系结构优点。
A. 定位透明性
B. 容错性高
C. 组件的可变性和可扩展性
D. 不同代理者系统之间的互操作性
- B 4. 微核体系结构中, 能够与客户机 (Clients) 通信的是 _____
A. 外部服务器 (External servers)
B. 适配器 (Adapters)
C. 内部服务器 (Internal servers)
D. 微核 (Microkernel)
- D 5. 不符合黑板模式特点的是 _____
A. 知识库可以分成条件部分和行动部分。
B. 解空间中的元素具有不同的抽象等级。
C. 控制组件监视黑板上的改动并决定接下来的动作。
D. 知识库从解空间中低等级的元素, 推导出高等级的元素。
- B 6. 基于命令 (Command) 模式实现取消 (Undo) 功能时, ConcreteCommand 类不需要的状态信息是 _____

A. 接收者对象

C. 接收者上执行操作及其参数

B. 调用者对象

D. 接收者对象中被改变的值得值

C 7. _____ 设计模式限制类的实例对象只能有一个。

A. 原型(Prototype)

C. 单例(Singleton)

B. 工厂方法(Factory Method)

D. 生成器(Builder)

A 8. 适配器(Adapter)设计模式可以用于_____。

A. 将已有类的接口转换成和目标接口兼容

B. 改进系统性能

C. 将客户端代码数据转换成目标接口期望的合适的格式

D. 使所有接口不兼容类可以一起工作

D 9. 用于为一个对象添加更多功能而不使用子类的是_____设计模式。

A. 桥接(Bridge)

B. 适配器(Adapter)

C. 组合(Composite)

D. 装饰器(Decorator)

C 10. 某软件公司正在设计一个通用的嵌入式数据处理平台, 需要支持多种数据处理芯片之间的数据传递与交换, 该平台的核心功能之一要求能够屏蔽芯片之间的数据交互, 使其耦合松散, 并且可以独立改变芯片之间的交互过程, 针对上述需求, 采用_____最为合适。

A. 抽象工厂模式

B. 策略模式

C. 中介者模式

D. 状态模式

B 11. 某软件公司正在设计一个图像处理软件, 该软件需要支持用户在图像处理中的撤销和重做等动作, 为了实现该功能, 采用_____最为合适。

A. 单例模式

B. 命令模式

C. 访问者模式

D. 适配器模式

A 12. 某互联网公司正在设计一套网络聊天系统, 为了限制用户在使用该系统时发表不恰当言论, 需要对聊天内容进行特定敏感词的过滤, 针对上述功能需求, 采用_____能够灵活配置敏感词的过滤过程。

A. 责任链模式

B. 工厂模式

C. 组合模式

D. 装饰模式

C 13. 某软件公司承接了为某工作流语言开发解释器的工作, 该工作流语言由多种活动节点构成, 具有类 XML 的语法结构, 用户要求解释器工作时, 对每个活动节点进行一系列的处理, 包括执行活动、日志记录、调用外部应用程序等, 并且要求处理过程具有可扩展能力, 针对这种需求, 公司采用_____最为恰当。

A. 适配器模式

B. 迭代器模式

C. 访问者模式

D. 观察者模式

A 14. 某公司开发一个文档编辑器, 该编辑器允许在文档中直接嵌入图形对象, 但开销很大, 用户在系统设计之初提出编辑器在打开文档时必须十分迅速, 可以暂时不显示当前页面以外的图形, 针对这种需求, 公司可以采用_____避免同时创建这些图形对象。

A. 代理模式

B. 外观模式

C. 桥接模式

D. 组合模式

C 15. 某软件公司正在设计一个图像处理软件, 该软件需要封装各种图像处理算法, 用户

能够根据需要灵活选择合适的处理算法。软件还支持高级用户根据一定的规则添加自定义处理算法。为了实现该功能,采用_____最为合适。

- A. 模板方法模式 B. 访问者模式 C. 策略模式 D. 观察者模式

三、简答题(每题5分,共20分)

1. 简述映像模式中的两层结构与纯粹的分层体系结构之间的区别。
2. 简述黑板体系结构和仓库系统之间的区别。
3. 简述“Model-View-Controller”结构和“Document-View”结构的关系。
MFC VCTH
4. 简述抽象工厂(Abstract Factory)模式中,如何实现其中的工厂类(分别基于 Factory Method 模式和 Prototype 模式实现)。

四、软件设计题(10分)

请设计程序,实现场景:“猫大叫一声,所有的老鼠都开始逃跑,主人被惊醒。”

要求:

- (1) 要有联动性,老鼠和主人的行为是被动的;
- (2) 考虑可扩展性,猫的叫声可能引起其他联动效应。

五、软件设计题(10分)

某咖啡店售卖咖啡时,可以根据顾客的要求在其中加入各种配料,咖啡店会根据所加入的配料来计算费用。咖啡店所供应的咖啡及配料的种类和价格如下表所示:

咖啡	价格/杯(Y)	配料	价格/份(Y)
浓缩咖啡(Espresso)	25	摩卡(Mocha)	10
深度烘焙咖啡(Dark Roast)	20	奶泡(Whip)	8

请给出设计方案,实现计算费用的功能。

六、软件设计题(20分)

Windows Media Player 和 RealPlayer 是两种常用的媒体播放器,它们的 API 结构和调用方法存在区别。请设计一个应用程序,要求支持这两种播放器 API,而且在将来可能还需要支持新的媒体播放器,请问如何设计该应用程序?

判断题 (2*15=30 分)

1. 在管道-过滤器风格中, 每个管道的一端多连接一个过滤器 (0) ~~X~~
2. 在分层系统风格中, 只在相邻层之间发生交互。 (1) ☒
3. 在黑板系统风格中, 是由输入流触发动作的。 (0) ~~X~~
(解释: 输入流触发的是数据库的方式, 黑板系统风格由中心数据结构的当前状态触发的)
4. 在抽象工厂模式中, 可以方便地添加新种类的产品。 (0) ~~X~~
(抽象工厂的缺点之一就是难以支持新种类的产品, 但它使得易于交换产品系列)
5. Builder 模式中, 由具体生成器生成的产品可以不具有公共父类。 (66 页) (1) ☒
6. Prototype 模式中, 需要对每个数据子类实现 Clone 操作。 (79 页) (1) ☒
7. 不能用多重继承方法实现真正的 Bridge 模式。 (103) (1) ☒
8. 装饰模式中, 要求被装饰对象具有一个公共的父类。 (117) (1) ☒
9. 在 Façade 模式中, 子系统知道 Façade 的存在。 (128) (0) ~~X~~
10. 使用 Flyweight 模式, 只有内部状态可以保存在 Flyweight 对象中。 (129) (1) ☒
11. 适配器 Adapter 和代理 Proxy 均为实体提供了一个相同的接口。 (1) ~~X~~
(这个题目怪怪的)
12. 空迭代器的 IsDone 操作一定返回 True。 (1) ☒
13. 使用 C++ 实现备忘录模式时, 需要将 Originator 作为 Memeto 的友元。 (1) ☒
14. 使用 Strategy 模式时, Strategy 对客户是透明的。 (0) ~~X~~ P208
15. Visitor 模式中, Visitor 可以访问的对象必须具有相同的父类。 (0) ~~X~~ P218
(visitor 和访问对象没有继承关系) X 不需要访问时, 源会将自身作为参数传递给访问者可以

二、简答题 (10*2=20 分)

1. 简述 MVC 风格与 Observer 模式的关系 (201) P70 P71 看笔记
MVC 风格包括了三类对象。模型 Model 是应用对象, 视图 View 是它在屏幕上的表示, 控制器 Controller 定义用户界面对用户输入的响应方式。
MVC 风格的 Model 和 View 对象的关系就是使用了 Observer 模式, view-controller 关系就是 strategy 模式, view 对象采用了 composite 模式。
MVC 风格包括了 Observer 模式。
2. 简述如何利用 command 模式, 实现对用户的操作进行撤销与重做。 (158) P143
这个就抄书上的吧, 简答题, 多写会错, 还被扣分。
3. 参考《设计模式》一书中, 2.8.3 节内前序循环器 preorderIterator 的实现, 实现后续循环器 PostIterator。 (20 分)

类 postorderIterator 从根图元得到 iterator, 并通过根图元将其子节点按从左到右的顺序推入队列中, 并将队列头元素作为第一个元素。

Void postIterator::First ()

```

Iterator<Glyph*> *l = _root->CreateIterator();
Iterator<Glyph*> *proot = l;
_iterators.RemoveAll();
while(!l->isDone())
{
    l->First(); → 将 iterator 指向第一个元素
    _iterators.Push(l);
    Iterator<Glyph*> *l = l->CurrentItem->CreateIterator();
}
    
```


With elements PostOrderIterator(TreeLinkedList T).

{ if (null == T) return; //递归基
TreeLinkedList subtree = T.getFirstChild(); //从当前节点取子节点
while (null != subtree) {

 //元素PostOrderIterator(subtree); //递归调用
 subtree = subtree.CurrentItem(); //只是调用队列头的Iterator的CurrentItem方法

 //Insert Last(T).Glyph* postorderIterator::CurrentItem() const
 //最后访问该节点

 Return _iterators.size() > 0;

 _iterators.Top()->CurrentItem(); 0;

Next 方法是将队列头内容弹出

Void PostOrderIterator::Next() const

{ While (_iterators.size() > 0)

{ Delete _iterators.Pop();

_iterators.Pop()->next();

四、软件设计题：通用查询条件生成模块 (30分)

在对数据库进行查询时，需要用到各种各样的查询条件，要求设计一个独立的通用查询条件生成模块，能帮助用户产生和修改所需要的查询条件。其他限制如下：

1) 对该模块可以配置要处理的数据库表中各个字段的类型，从而保证该模块的通用性。例如可以配置信息如下：

姓名|name|string<生日|birthday|date><年龄|age|integer>

2) 提供界面供用户进行查询条件的生产与修改。

3) 该模块最终生成的查询条件返回结果是一个字符串，该字符串可用于sql语句在where子句中，例如一个可能的结果是：

age<6 or (name like '王%' and age>12)

要求：针对此模块，描述软件的设计方案，并说明其中用到的设计模式。

1. 录入记录

2. A → DoMakeQueryUI (window *). 产生界面

2007 年高级软件工程试卷

学号: _____

姓名: 法桐桐

成绩: _____

一、判断题 (每题 2 分, 共计 30 分)

1. 在隐式调用风格中, 往往不需要使用显式调用作为 补充。 (X) 20/41
2. 在管道-过滤器风格中, 过滤器自身是无状态的。 (✓) 11/41
3. 从 Observer 模式的角度看待 MVC 风格, 其中 View 和 Controller 均可以视为 Observer。 (✓)
4. 在抽象工厂模式中, 可以方便地添加新种类的产品。X (X) p59
5. Builder 模式中, 由具体生成器生成的产品可以不具有公共父类。 (✓)
6. Prototype 模式中, 需要对每个数据子类实现 Clone 操作。 (✓) p79-7
7. 在解释器模式中, 需要为每一个规则至少定义一个类。 (✓)
8. 装饰模式中, 要求被装饰对象具有一个公共的父类。 (✓)
9. 在 Facade 模式中, 了系统知道 Facade 的存在。X (X)
10. 使用 Flyweight 模式, 有利于节省空间及运算开销。 (X)
11. 适配器 Adapter 和代理 Proxy 均为实体提供了一个相同的接口。 (X)
12. 空迭代器的 IsDone 操作一定返回 True。 (✓)
13. 使用 C++ 实现备忘录模式时, 需要将 Originator 作为 Memo 的友元。 (✓)
14. 使用 Strategy 模式时, Strategy 对客户是透明的。 (X)
15. Visitor 模式适用于添加新的操作, 也适用于增加新的数据类型。 (X)

二、简答题 (每题 10 分, 共计 30 分)

- P44 1. 简述 Adapter 模式与 Bridge 模式的区别。 Bridge 模式的目的是将接口部分和实现部分分离, 从而
- P154 2. 简述如何利用 Command 模式, 实现对用户的操作进行撤销与重做。对它们可以较为容易地实现
- P129 3. 简述 Flyweight 模式中内部状态与外部状态的关系。把它的加以分离, 从而

四、软件设计题: 通用查询条件生成模块 (40 分)

在对数据库进行查询时, 需要用到各种各样的查询条件。要求设计一个独立的通用查询条件生成模块, 能帮助用户产生和修改所需要的查询条件。其他限制如下:

- 1) 对该模块可以配置要处理的数据库表中各个字段的类型, 从而保证该模块的通用性。例如可能的配置信息如下:
 $\langle \text{姓名}/\text{name}/\text{string} \rangle$ 、 $\langle \text{生日}/\text{birthday}/\text{date} \rangle$ 、 $\langle \text{年龄}/\text{age}/\text{integer} \rangle$...
- 2) 提供界面供用户进行查询条件的生成与修改。
- 3) 该模块最终生成的查询条件返回结果是一个字符串, 该字符串可以用于 SQL 语句的 Where 子句中。例如一个可能的结果是:
 $\text{age} < 6 \text{ or } (\text{name like '王\%' and age} > 12)$
- 4) 不需要考虑与数据库的交互问题。

要求: 针对此模块, 描述软件设计方案, 并说明其中用到的设计模式。并说明用户在界面上如何对查询条件进行操作。