目录

[UML图中类之间的关系:依赖,泛化,关联,聚合,组合,实现 3](#_Toc90252546)

[UML行为建模-顺序图 9](#_Toc90252547)

[UML构件图 14](#_Toc90252548)

[二、包含的元素 14](#_Toc90252549)

[1.组件（Component）->构件 14](#_Toc90252550)

[2.接口（Interface） 15](#_Toc90252551)

[3.外部接口——端口 16](#_Toc90252552)

[4.连接器（Connector）——连接件 17](#_Toc90252553)

[4..依赖关系（Dependency） 18](#_Toc90252554)

[三、相关概念比较 18](#_Toc90252555)

[1.组件与类 18](#_Toc90252556)

[2.组件与包 19](#_Toc90252557)

[四、组件图建模技术 19](#_Toc90252558)

[概念理解 24](#_Toc90252559)

[《软件工程》题集 28](#_Toc90252560)

[一、选择题（在每小题的四个备选答案中，选出一个正确答案，并将正确答案的序号填在题干的括号内，每小题1分） 28](#_Toc90252561)

[二、填空题（请把答案写在相应的横线上,每小题1.5分） 36](#_Toc90252562)

[三、简答题(简要回答下列问题，每小题4分\*5=20) 38](#_Toc90252563)

[四、应用题(每小题10分\*3=30) 44](#_Toc90252564)

[软件工程试题一 49](#_Toc90252565)

[一．单项选择题 49](#_Toc90252566)

[二．判断题(本大题共\_\_15\_\_题，每题\_\_\_1\_\_\_分，共\_\_\_15\_\_\_\_\_分。 50](#_Toc90252567)

[三．简答题 50](#_Toc90252568)

[四．综合题(本大题共\_\_3\_\_题，每题\_\_\_10\_\_\_分，共\_\_\_30\_\_\_\_\_分。) 51](#_Toc90252569)

[软件工程试题二 51](#_Toc90252570)

[一、填空题（每空1分，共25分） 51](#_Toc90252571)

[二、单项选择题（每小题2分，共20分） 51](#_Toc90252572)

[三、判断题（正确的在括号内打上“√”，错误的打上“×”。每题2分，共20分） 52](#_Toc90252573)

[四、简答题（每题5分，共20分） 52](#_Toc90252574)

[五、程序设计与算法描述题（共15分） 52](#_Toc90252575)

[软件工程试题三 53](#_Toc90252576)

[一、选择题：（每题1分，共20分）（将答案写在题号前的（）中） 53](#_Toc90252577)

[二．判断题（每题1分，共10分）将答案写在题号前的（ ）中，正确用√， 错误用χ。 54](#_Toc90252578)

[三、简答题：（每题5分，共25分） 54](#_Toc90252579)

[四、应用题（每题8分，共40分） 55](#_Toc90252580)

[软件工程试题四 55](#_Toc90252581)

[一、选择题 55](#_Toc90252582)

[二、填空题 57](#_Toc90252583)

[三、概念题 57](#_Toc90252584)

[四、简答题 58](#_Toc90252585)

**UML图中类之间的关系:依赖,泛化,关联,聚合,组合,实现**

**类与类图**

1) 类(Class)封装了数据和行为，是面向对象的重要组成部分，它是具有相同属性、操作、关系的对象集合的总称。

2) 在系统中，每个类具有一定的职责，职责指的是类所担任的任务，即类要完成什么样的功能，要承担什么样的义务。一个类可以有多种职责，设计得好的类一般只有一种职责，在定义类的时候，将类的职责分解成为类的属性和操作（即方法）。

3) 类的属性即类的数据职责，类的操作即类的行为职责

**一、依赖关系(Dependence)**

依赖关系（Dependence）：假设A类的变化引起了B类的变化，则说名B类依赖于A类。

依赖关系(Dependency) 是一种使用关系，特定事物的改变有可能会影响到使用该事物的其他事物，在需要表示一个事物使用另一个事物时使用依赖关系。大多数情况下，依 赖关系体现在某个类的方法使用另一个类的对象作为参数。

在UML中，依赖关系用带箭头的虚线表示，由依赖的一方指向被依赖的一方。



public class Driver

{

public void drive(Car car)

{

car.move();

}

……

}

public class Car

{

public void move()

{

......

}

……

}

依赖关系有如下三种情况：

1、A类是B类中的（某中方法的）局部变量；

2、A类是B类方法当中的一个参数；

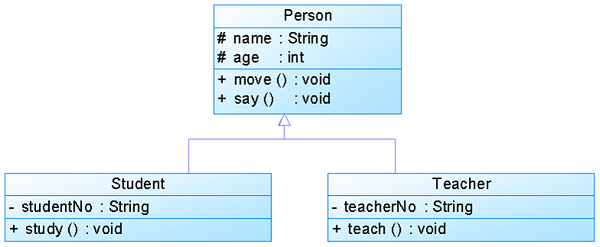
3、A类向B类发送消息，从而影响B类发生变化；

**二、泛化关系（Generalization）**

泛化关系（Generalization）：A是B和C的父类，B,C具有公共类（父类）A，说明A是B,C的一般化（概括，也称泛化）

泛化关系(Generalization)也就是继承关系，也称为“is-a-kind-of”关系，泛化关系用于描述父类与子类之间的关系，父类又称作基类或超类，子类又称作派生类。在UML中，泛 化关系用带空心三角形的直线来表示。

在代码实现时，使用面向对象的继承机制来实现泛化关系，如在Java语言中使用extends关键字、在C++/C#中使用冒号“：”来实现。



public class Person

{

protected String name;

protected int age;

public void move()

{

……

}

public void say()

{

……

}

}

public class Student extends Person

{

private String studentNo;

public void study()

{

……

}

}

在UML当中，对泛化关系有三个要求：

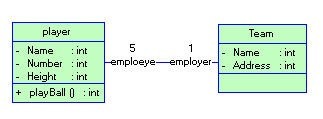
1、子类与父类应该完全一致，父类所具有的属性、操作，子类应该都有；

2、子类中除了与父类一致的信息以外，还包括额外的信息；

3、可以使用父类的实例的地方，也可以使用子类的实例；

**三、关联关系（Association）**

关联关系（Association）:类之间的联系，如客户和订单，每个订单对应特定的客户，每个客户对应一些特定的订单，再如篮球队员与球队之间的关联（下图所示）。



其中，关联两边的"employee"和“employer”标示了两者之间的关系，而数字表示两者的关系的限制，是关联两者之间的多重性。通常有“\*”（表示所有，不限），“1”（表示有且仅有一个），“0...”（表示0个或者多个），“0，1”（表示0个或者一个），“n...m”(表示n到m个都可以),“m...\*”（表示至少m个）。

关联关系(Association) 是类与类之间最常用的一种关系，它是一种结构化关系，用于表示一类对象与另一类对象之间有联系。

在UML类图中，用实线连接有关联的对象所对应的类，在使用Java、C#和C++等编程语言实现关联关系时，通常将一个类的对象作为另一个类的属性。

在使用类图表示关联关系时可以在关联线上标注角色名。

1) 双向关联: 默认情况下，关联是双向的。



public class Customer

{

private Product[] products;

……

}

public class Product

{

private Customer customer;

……

}

2 ) 单向关联:类的关联关系也可以是单向的，单向关联用带箭头的实线表示.



public class Customer

{

private Address address;

……

}

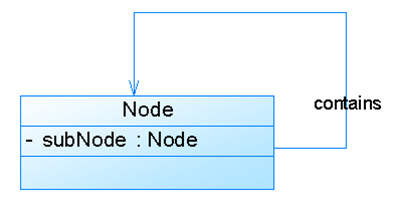
public class Address

{

……

}

3) 自关联: 在系统中可能会存在一些类的属性对象类型为该类本身，这种特殊的关联关系称为自关联。



public class Node

{

private Node subNode;

……

}

4) 重数性关联: 重数性关联关系又称为多重性关联关系(Multiplicity)，表示一个类的对象与另一个类的对象连接的个数。在UML中多重性关系可以直接在关联直线上增加一个数字表示与之对应的另一个类的对象的个数。





public class Form

{

private Button buttons[];

……

}

public class Button

{

…

}

**四、聚合关系（Aggregation）**

聚合关系（Aggregation）:表示的是整体和部分的关系，整体与部分 可以分开.

聚合关系(Aggregation) 表示一个整体与部分的关系。通常在定义一个整体类后，再去分析这个整体类的组成结构，从而找出一些成员类，该整体类和成员类之间就形成了聚合 关系。

在聚合关系中，成员类是整体类的一部分，即成员对象是整体对象的一部分，但是成员对象可以脱离整体对象独立存在。在UML中，聚合关系用带空心菱形的直线表示。



public class Car

{

private Engine engine;

public Car(Engine engine)

{

this.engine = engine;

}

public void setEngine(Engine engine)

{

this.engine = engine;

}

……

}

public class Engine

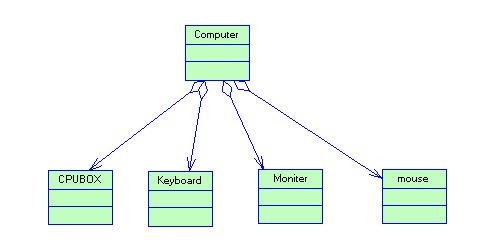
{

……

}

如：电话机包括一个话筒

电脑包括键盘、显示器，一台电脑可以和多个键盘、多个显示器搭配，确定键盘和显示器是可以和主机分开的，主机可以选择其他的键盘、显示器组成电脑；

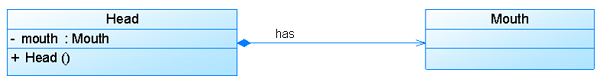
****

**五、组合关系（Composition）**

组合关系（Composition）:也是整体与部分的关系，但是整体与部分不可以分开.

组合关系(Composition)也表示类之间整体和部分的关系，但是组合关系中部分和整体具有统一的生存期。一旦整体对象不存在，部分对象也将不存在，部分对象与整体对象之 间具有同生共死的关系。

在组合关系中，成员类是整体类的一部分，而且整体类可以控制成员类的生命周期，即成员类的存在依赖于整体类。在UML中，组合关系用带实心菱形的直线表示。



public class Head

{

private Mouth mouth;

public Head()

{

mouth = new Mouth();

}

……

}

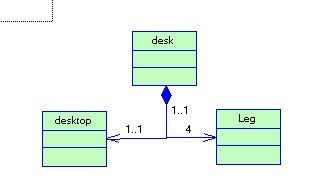
public class Mouth

{

……

}

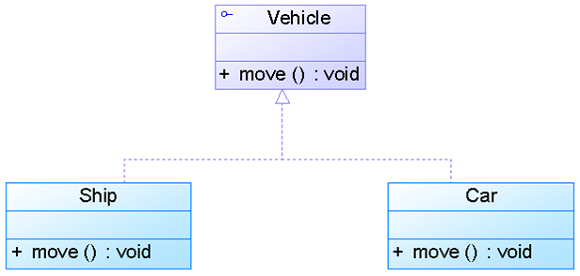
如：公司和部门，部门是部分，公司是整体，公司A的财务部不可能和公司B的财务部对换，就是说，公司A不能和自己的财务部分开； 人与人的心脏.



**六、实现关系（Implementation)**

实现关系（Implementation）：是用来规定接口和实线接口的类或者构建结构的关系，接口是操作的集合，而这些操作就用于规定类或者构建的一种服务。

接口之间也可以有与类之间关系类似的继承关系和依赖关系，但是接口和类之间还存在一种实现关系(Realization)，在这种关系中，类实现了接口，类中的操作实现了接口中所 声明的操作。在UML中，类与接口之间的实现关系用带空心三角形的虚线来表示。



public interface Vehicle

{

public void move();

}

public class Ship implements Vehicle

{

public void move()

{

……

}

}

public class Car implements Vehicle

{

public void move()

{

……

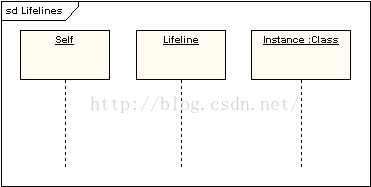
}

}

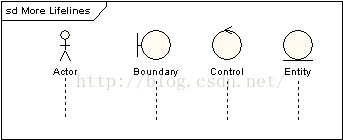
**UML行为建模-顺序图**

**顺序图**顺序图是交互图的一种形式，它显示对象沿生命线发展，对象之间随时间的交互表示为从源生命线指向目标生命线的消息。顺序图能很好地显示那些对象与其它那些对象通信，什么消息触发了这些通信，顺序图不能很好显示复杂过程的逻辑。

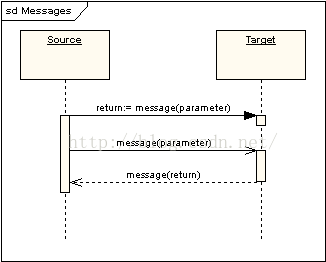
**生命线**一条生命线在顺序图中代表一个独立的参与者。表示为包含对象名的矩形，如果它的名字是"self"，则说明该生命线代表控制带顺序图的类元。

****

有时，顺序图会包含一个顶端是执行者的生命线。这情况说明掌握这个顺序图的是用例。健壮图中的边界，控制和实体元素也可以有生命线。

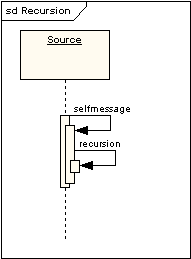
****

**消息**消息显示为箭头。消息可以完成传输，也可能丢失和找回，它可以是同步的，也可以是异步的，即可以是调用，也可以是信号。在下图中，第一条消息是同步消息(标为实箭头)完成传输，并隐含一条返回消息。第二条消息是异步消息 (标为实线箭头)，第三条是异步返回消息(标为虚线)。

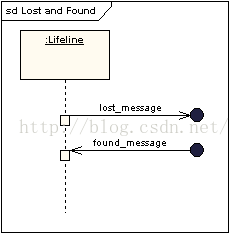
****

**执行发生**向下延伸的细条状矩形表示执行事件或控制焦点的激活。在上图中有三个执行事件。第一个是源对象发送两条消息和收到两条回复。第二个是目标对象收到一条同步消息并返回一条回复。第三个是目标对象收到一条异步消息并返回一条回复。

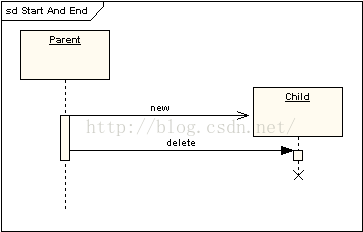
**内部通信**内部消息表现为一个操作的递归调用，或一个方法调用属于同一个对象的其他方法。显示为生命线上执行事件的嵌套控制焦点。



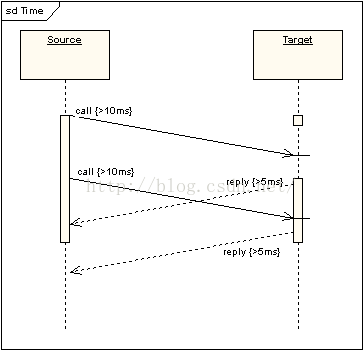
**迷路消息和拾取消息**迷路消息是那些发送了却没有到达指定接收者，或者到达的接收者不再当前图中。拾取消息是收到来自那些未知的发送者，或者来自没有显示在当前图的发送者的消息。它们都表明是去往或来自一个终点元素。



**生命线开始与结束**生命线可以在顺序图时间刻度范围内创建和销毁，在下面的例子中，生命线被停止符号（叉号）终止。在前面的例子中，生命线顶端的符号（Child）显示在比创建它的对象符号（parent）沿页面要低的位置上。下图显示创建和终止对象。



**时间和期限约束**消息默认显示为水平线。因为生命线显示为沿屏幕向下的时间通道，所以当给实时系统建模，或是有时间约束的业务过程建模，考虑执行动作所需时间长度是很重要的。因此可以给消息设置一个期限约束，这样的消息显示为下斜线。

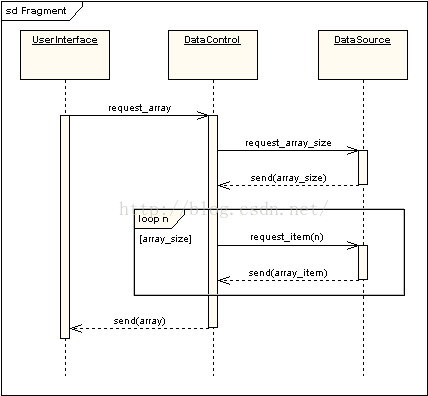


**复合片段**

如前面所说，顺序图不适合表达复杂的过程逻辑。在一种情况下，有许多机制允许把一定程度的过程逻辑加入到图中，并把它们放到复合片段的标题下。复合片段是一个或多个处理顺序被包含在一个框架中，并在指定名称的环境下执行。 片段可以是:

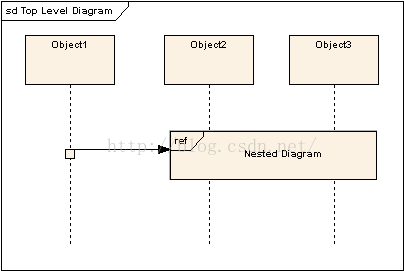
* 选择性片段 (显示 “alt”) 为 if…then…else 结构建模。
* 选项片段 (显示 “opt”) 为 "switch"(开关) 结构建模。
* 中断片段对被处理事件的可选择顺序建模，而不是该图的其他部分。
* 并行片段(显示 “par”) 为并发处理建模。
* 弱顺序片段 (显示 “seq”) 包含了一组消息，这组消息必须在后继片段开始之前被处理。但不会把片段内消息的先后顺序强加到不共享同一条生命线的消息上。
* 严格顺序片段 (显示 “strict”) 包含了一系列需要按照给定顺序处理的消息。
* 非片段 (显示 “neg”) 包含了一系列不可用的消息。
* 关键片段 具有关键部分。
* 忽略片段 声明一个没有意义的消息，如果它出现在当前上下文中。
* 考虑片段与忽略片段相反，不包含在考虑片段内的消息都应该被忽略。
* 断言片段 (显示 “assert”)标明任何没有显示为声明操作数的顺序都是无效的。
* 循环片段 包含一系列被重复的消息。

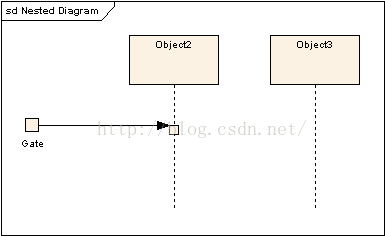
下图显示的是循环片段：



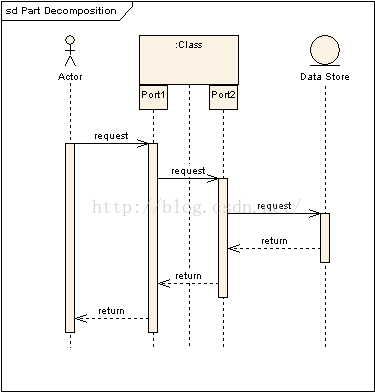
这也是一个类似于复合片段的交互发生。 交互发生被其他图参考，显示为左上角带"ref"，将被参考图名显示在方框的中间。

**门**门是连接片段内消息和片段外消息的连接点。 在EA中，门显示为片段框架上的小正方形。作用为顺序图与页面外的连接器。 用来表示进来的消息源，或者出去消息的终点。下面两个图显示它们在实践中的使用。注意：" top level diagram"中的门用消息箭头指向参考片段，在这里没有必要把它画成方块。

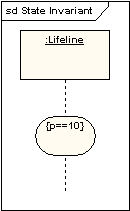




**部分分解**一个对象可以引出多条生命线，使得对象内部和对象之间的消息显示在同一图上。



**状态常量 / 延续**状态常量是生命线的约束，运行时始终为"真"。显示为两侧半圆的矩形，如下图：



延续虽与状态常量有同样的标注，但是被用于复合片段，并可以延伸跨越多条生命线。

**UML构件图**

组件图(Component Diagram)又称为构件图，他描述的是在**软件系统中**遵从并实现一组**接口**的物理的、**可替换的**软件模块。

构件图 = 构件(Component) + 接口(Interface) + 关系(Relationship) + 端口(Port) + 连接器(Connector)

* 在面向对象系统的**物理方面**进行建模要用到两种图：**组件图和配置图。**

**二、包含的元素**

**1.组件（Component）->构件**

* 组件是定义了良好接口的物理实现单元，是系统中可替换的物理部件。
* 组件代表系统的一个物理实现块，代表逻辑模型元素如类、接口、协同等的物理打包。
* 构件通过它的提供接口和请求接口展现行为。
* 由于在UML2.0中，构件是一种类，因此构件具有属性、操作和可见性。这些概念的含义与在类图中定义的是一样的，只是在这里把这些概念应用在构件上。

**组件的命名：**

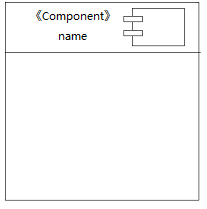
  组件的名称有两种：简单名和路径名。并依据目标操作系统可以添加相应的扩展名，例如java和dll。

**表示方式**：组件用一个左侧带有突出两个小矩形的矩形来表示。



**UML2.0中表示方式**：构件用加构造型《component》的矩形框来表示，左上角添加以前的构件符号，如果没有构件细节可在中央直接写上名字。

（构造型和左上角的图标可以二选一）



**UML2.0把构件分为基本构件和包装构件**

* **基本构件**  
  注重于把构件定义为在系统中可执行的元素。
* **包装构件**  
  扩展了基本构件的概念，它注重于把构件定义为一组相关的元素，这组元素为开发过程的一部分。也即, 包装构件定义了构件的命名空间方面。在构件的命名空间中，可以包括类、接口、构件、包、用况、依赖（如映射）和制品。按照这种扩展，构件也具有如下的含义：可以用构件来装配大粒度的构件，方法为把所复用的构件作为大粒度构件的成分，并把它们的请求和提供接口连接在一起（简单理解：组件包含组件，组拼大组件）。

**组件的种类**

1. 配置组件（Deployment Component）：运行系统需要配置的组件，是形成可执行文件的基础—操作系统、JAVA虚拟机、DBMS；
2. 工作产品组件（Work Product Component）：包括模型、源代码和用于创建配置组件的数据文件，它们是配置组件的来源—UML图、java类和数据库表；
3. 执行组件（Execution Component）：在运行时创建的组件，是最终可运行的系统产生的允许结果—.net组件

**2.接口（Interface）**

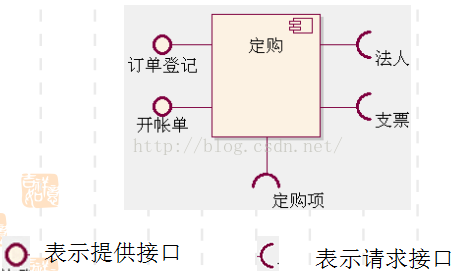
* 接口（interface）接口由一组操作组成，它指定了一个契约，这个契约必须由实现和使用这个接口的构件的所遵循。

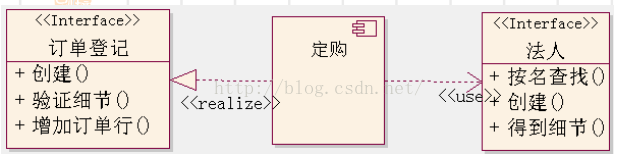
**接口分提供接口和请求接口**

* 把构件实现的接口称为提供接口（供接口），这意味着构件的提供接口是给其它构件提供服务的。实现接口的构件支持由该接口所拥有的特征，包括接口拥有的约束。
* 构件使用的接口被称为请求接口（需接口），即构件向其它构件请求服务时要遵循的接口

**表示方式：**

* 供接口用“棒棒糖”式的图形表示，即由一个封闭的圆形与一条直线组成。
* 需接口用“插座”式的图形表示，即由一个半圆与一条直线组成。





在java语言中供接口通过关键字implements来显式地表示，需接口被类所使用的任何接口类型隐式地定义。

**3.外部接口——端口**

* 端口是UML2.0引入的概念
* 端口描述了在构件与它的环境之间以及在构件与它的内部构件之间的一个显示地交互点
* 端口是一个封装构件的显示的对外窗口，**所有进出构件的交互都要通过端口。**
* 使用端口能在更大的程度上增加构件的封装性和可替代性。
* 端口是构件的一部分，端口的实例随着它们所属的构件的实例一起被创建和撤消。

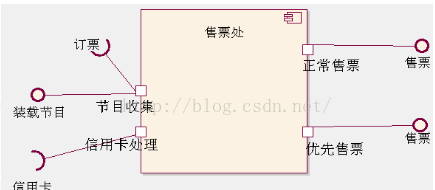
**<1>接口与端口的关系**

  提供接口说明了通过端口来提供服务，请求接口说明了通过端口需要从其它构件获得服务。

  一个构件可以通过一个特定端口同另一个构件通讯，而且通讯完全是通过由端口支持的接口来描述的。

**<2>表示方式**

  尾部加小方框的正常接口表示，小方框就被称为端口。



**4.连接器（Connector）——连接件**

**UML2.0提供两种类型的连接器：**

1. 代理连接器（Delegation Connector）——委托连接件：连接外部接口的端口和内部接口。
2. 组装连接器（Assembly Connector）——组装连接件：组件连接器表示构件之间的关系，它连接构件内部的类，将一个构件的供接口和一个构件的需接口捆绑在一起

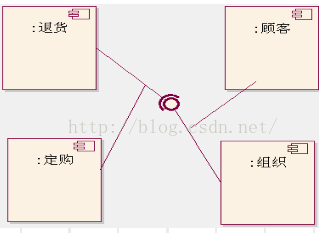
* 连接端口意味着请求端口要调用提供端口中的操作，以得到服务。
* 立端口和接口的优点在于在设计时，两个构件彼此不需要了解对方的内部，只要它们的接口是相互兼容的即可。
* 如果一个端口提供一个特定的接口而另一个端口需要这个接口，且接口是兼容的，那么这两个端口-便是可连接的。

**<1>组装连接件**

有两种表示装配连接件的方法：

1. 如果要显式地把两个构件实例衔接在一起，在它们的端口之间画一条线即可。
2. 如果两个构件实例相连是由于它们有兼容的接口，则可以使用一个“球－穴”标记来表示构件实例之间的连接关系。

  装配连接件是两个构件实例间的连接件，它定义一个构件实例提供服务，另一个构件实例使用这些服务。装配连接件用于把一个请求接口或端口与一个提供接口或端口的连接起来。在执行时，消息起源于一个请求端口，沿着连接件传递，被交付到一个提供端口



**<2>委托连接件**

  委托有这样的含义：具体的消息流将发生在所连接的端口之间，可能要跨越多个层次，最终到达要对消息进行处理的最终部件实例。这样，使用委托连接件可对构件行为的层次分解建模。

**委托连接件把外部对构件端口的请求分发到构件内部的部件实例进行处理，或者通过构件端口把构件内部部件实例向构件外部的请求分发出去。**

  构件内部的一个部件可以是另一个构件或是一个类。注意，必须在两个提供端口间或两个请求端口间定义委托连接件。

**注意事项**：因为构件是可以嵌套的，所以内部构件之间的连接（球-穴）是组装连接件，内部构件与端口之间的连接（实线箭头）是委托连接件。

**4..依赖关系（Dependency）**

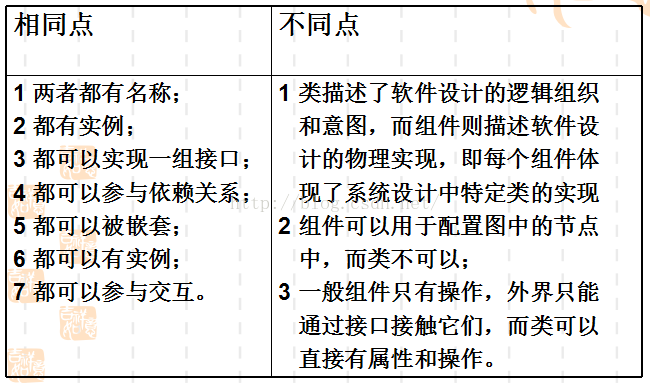
组件图用依赖关系表示各组件之间存在的关系类型。

  在UML中，组件图中依赖关系的表示方法与类图中依赖关系相同，都是一个由客户指向提供者的虚线箭头。



**三、相关概念比较**

**1.组件与类**



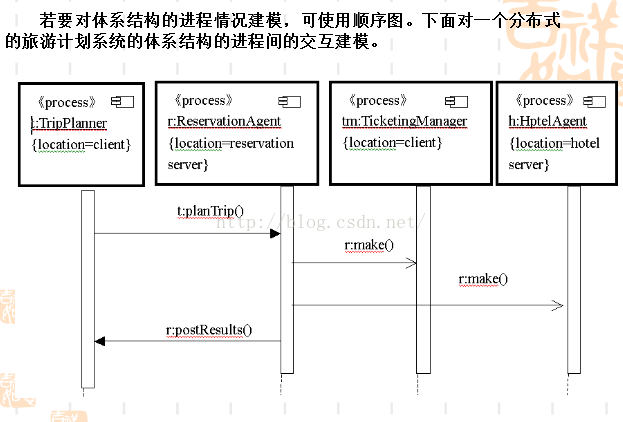
**2.组件与包**

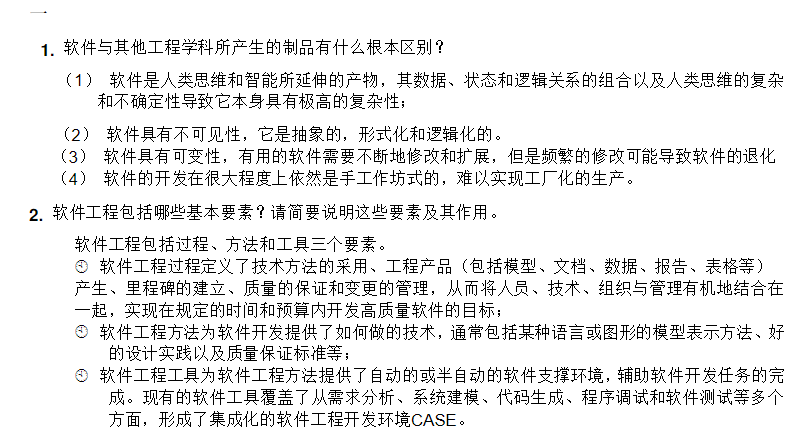
二者均为分组组织机制，但也有许多不同之处：

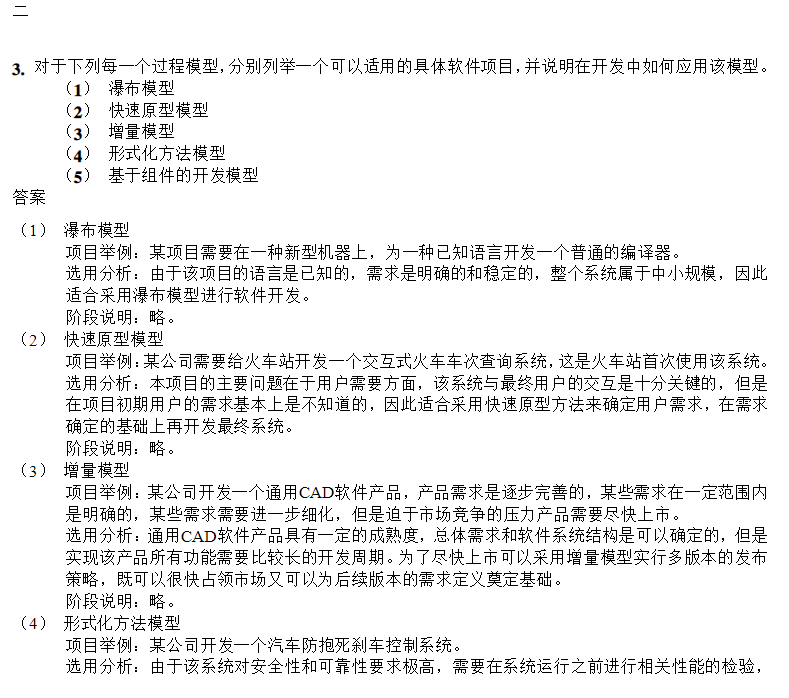
* 一个组件代表一个物理的代码模块，而包可以包含成组的逻辑模型元素，也可以包含物理的组件；可以用包来组织用例(use case)，不可以用组件来组织用例。
* 一个类可以出现在多个组件中，却只能在一个包中定义。
* 配置图(deployment diagram)中，节点(node)中可以放组件，但不可以放包。
* 包只是类型(type)，组件可以是实例也可以是类型。
* 包可以作为开发视图(development view)，用于管理。组件可作为物理视图(physical view)，用于部署。但反之不然。

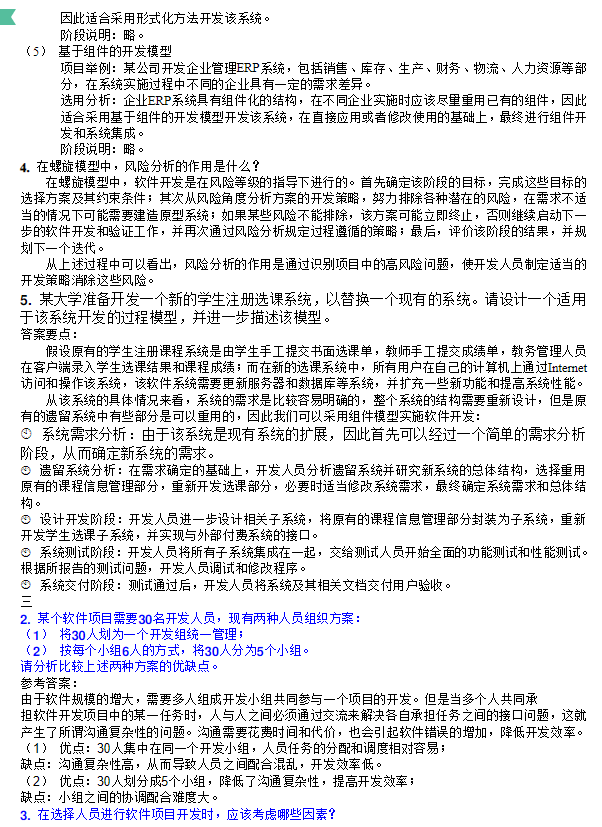
**四、组件图建模技术**

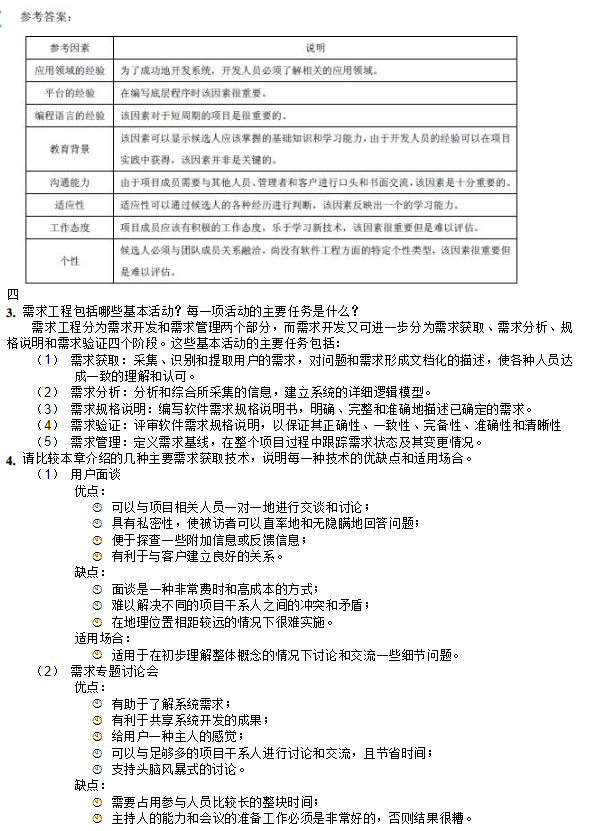
1. 对系统中的组件建模—分解系统，考虑有关系统的组成管理、软件的重用和物理节点的配置等因素，把关系密切的可执行程序和对象分别归入组件，找出相应的类、接口等模型元素。
2. 对相应组件提供的接口建模。
3. 对组件之间的依赖关系建模。
4. 将逻辑设计映射成物理实现。
5. 对建模的结果进行精化和细化。











**概念理解**

1、什么是软件工程？它目标和内容是什么？

软件工程就是用科学的知识和技术原理来定义，开发，维护软件的一门学科。

软件工程目标：付出较低开发成本；达到要求的功能；取得较好的性能；开发的软件易于移植；只需较低的维护费用；能按时完成开发任务，及时交付使用；开发的软件可靠性高。

软件工程内容：包括开发技术和开发管理两个方面。

软件开发中常见的软件过程模型有瀑布模型、原型模型、螺旋模型、喷泉模型、统一软件过程等。

其中喷泉模型、统一软件过程适用于面向对象的软件开发。

结构化分析：简称SA，面向数据流进行数据分析的方法。采用自顶向下逐层分解的分析策略。顶层抽象地描述整个系统，底层具体地画出系统工程的每个细节。中间层则是从抽象到具体的过渡。使用数据流图，数据字典，作为描述工具，使用结构化语言，判定表，判定树描述加工逻辑。

把一个软件需求转换为软件表示时，首先设计出软件总的体系结构。称为概要设计或结构设计。

基本任务：

⑴设计软件系统结构

⑵进行数据结构及数据库的设计

⑶编写概要设计的文档

⑷评审

**软件概要设计即该阶段的基本任务：**

把一个软件需求转换为软件表示时，首先设计出软件总的体系结构。称为概要设计或结构设计。

基本任务：

⑴设计软件系统结构

⑵进行数据结构及数据库的设计

⑶编写概要设计的文档

⑷评审

软件维护有哪些内容？

（1）校正性维护。在软件交付使用后，一些隐含的错误在某些特定的使用环境下会暴露出来。为了识别和纠正错误，修改软件性能上的缺陷，应进行确定和修改错误的过程，这个过程就称为校正性维护。

（2）适应性维护。为了使应用软件适应计算机硬件、软件环境及数据环境的不断发生的变化而修改软件的过程称为适应性维护。

（3）完善性维护。为增加软件功能、增强软件性能、提高软件运行效率而进行的维护活动称为完善性维护。

（4）预防性维护。为了提高软件的可维护性和可靠性而对软件进行的修改称为预防性维护。

**软件测试要经过哪些步骤?这些测试与软件开发各阶段之间有什么关系?**

软件测试要经过的步骤是：单元测试→集成测试→确认测试→系统测试。

单元测试对源程序中每一个程序单元进行测试，检查各个模块是否正确实现规定的功能，从而发现模块在编码中或算法中的错误。该阶段涉及编码和详细设计文档。

集成测试是为了检查与设计相关的软件体系结构的有关问题，也就是检查概要设计是否合理有效。

确认测试主要是检查已实现的软件是否满足需求规格说明书中确定了的各种需求。

系统测试是把已确认的软件与其他系统元素(如硬件、其他支持软件、数据、人工等)结合在一起进行测试。以确定软件是否可以支付使用。

**什么是软件生存周期？它有哪几个活动？**

软件生存周期：一个软件从提出开发要求开始直到该软件报废为止的整个时期。

包括：可行性分析和项目开发计划，需求分析，概要设计，详细设计，编码，测试，维护。

衡量模块独立性的两个标准是什么？各表示什么含义？

内聚和耦合。

内聚：又称为块内联系，指模块内部各成分之间相互关联的程度，以高内聚为设计目标。

耦合：也称块间联系，模块之间相互联系程度的度量，联系越紧密，耦合性越强，独立性越差，以低耦合为设计目标。

结构化分析：简称SA，面向数据流进行数据分析的方法。采用自顶向下逐层分解的分析策略。顶层抽象地描述整个系统，底层具体地画出系统工程的每个细节。中间层则是从抽象到具体的过渡。

实现的步骤：⑴了解当前系统的工作流程，获得当前系统的物理模型。 ⑵抽象出当前系统的逻辑模型。 ⑶建立目标系统的逻辑模型。 ⑷作进一步补充和优化。

需求分析：开发人员准确地理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的需求规格说明的过程。

基本任务：

⑴问题识别

⑵分析与综合，导出软件的逻辑模型

⑶编写文档

数据流图（DFD）一般可分为"变换流"，"事物流"两种。

"变换流"：由输入﹑输出﹑变换（或称处理）三部分组成，是一顺序结构。

"事物流"：它的某个加工，分离成许多发散的数据流，形成许多加工路径，并且根据输入值选择其中一个路径来执行。（这人加工称为事务处理中心）

单元测试主要针对模块的以下五个基本特征进行测试：

(1)模块接口

(2)局部数据结构

(3)重要的执行路径

(4)错误处理

(5)边界条件

**什么是软件生存周期模型？有哪些主要模型？**

生存周期模型：描述软件开发过程中各种活动如何执行的模型。对软件开发提供强有力的支持，为开发过程中的活动提供统一的政策保证，为参与开发的人员提供帮助和指导，是软件生存周期模型化技术的基础，也是建立软件开发环境的核心。

主要有：瀑布模型，增量模型，螺旋模型，喷泉模型，基于知识的模型，变换模型。

软件设计的基本原理包括哪些内容？

⑴模块化：模块是软件的组成部分，是具有独立功能且可命名的一段程序，所有模块组成整体，可以满足问题的要求。

⑵抽象：认识复杂现象过程所使用权的工具，只考虑事物本质的共同特性，忽略细节和其它因素。通过抽象确定组成软件的过程实体。

⑶信息隐蔽：将模块实现自身功能的细节与数据"隐蔽"起来。模块间仅交换为完成系统功能所必须的信息。

⑷模块独立性：每个模块只完成系统要求的独立的子功能。

结构化程序设计的基本要点是什么?

主要有三个：

1、采用自顶向下、逐步求精的程序设计方法

2、使用三种基本控制结构构造程序。任何程序都可以由顺序、选择、重复(循环)三种基本控制结构构造，这三种基本结构的共同点是单入口、单出口。

3、主程序员组的组织形式。

**详细设计有哪几种描述方法？**

详细设计的描述方法有图形、表格和语言，其中图形常用结构化程序流程图、盒图和PAD(问题分析图)为描述工具，语言常用过程设计语言(PDL)来作为工具。

单元测试中采用什么方法?

测试的方法是为被测试模块编写驱动模块和桩模块来实现被测试单元的可运行。通过驱动模块来模拟被测试模块的上级调用模块，以上级模块调用被测模块的格式驱动被测模块，接收被测模块的结构并输出。桩模块则用来代替被测试模块所调用的模块。它的作用是返回被测模块所需的信息。

**文档在软件工程中的作用**

文档在软件工程中的作用如下：

(1)提高软件开发过程的能见度

(2)提高开发效率

(3)作为开发人员阶段工作成果和结束标志

(4)记录开发过程的有关信息便于使用与维护；

(5)提供软件运行、维护和培训有关资料；

(6)便于用户了解软件功能、性能。

软件工程三要素之间的关系

软件质量是软件工程的生命线，软件工程以质量保证为基础。

质量管理促进了过程的改进,创造了许多行之有效的软件开发方法和工具。

软件工程釆用层次化的方法,每个层次都包括过程、方法、工具三要素。

方法支撑过程和工具，过程和工具又促进方法学的研究。

面向对象的特征是：

(1)对象唯一性。每个对象都有自身唯一的标识，在对象生存期中，其标识不变，不同的对象不能有不同的标识。

(2)分类性。是指将具有一致的数据结构(属性)和行为(操作)的对象抽象成类

(3)继承性。是子类自动共享父类数据结构和方法的机制，这是类之间的一种关系。

(4)多态性。是指相同的操作或函数过程可以作用于多种类型的对象上并获得不同的结果。

黑盒测试和白盒测试的区别：前者基于功能，后者基于结构；

黑盒测试常用方法有：边界值、等价类、因果图、错误推测法等。

**数据字典?数据字典与数据流图的关系。**

数据字典是关于数据的信息的集合，对数据流程图中的各个元素做完整的定义与说明，是数据流程图的补充工具；

数据字典与数据流图的关系：数据流图和数据字典共同构成系统的逻辑模型，没有数据字典数据流图就不严格，然而没有数据流图数据字典也难于发挥作用。 数据流图和对数据流图中每个元素的精确定义放在一起，才能共同构成系统的规格说明。

软件生命周期分为三个时期八个阶段：

软件定义：问题定义、可行性研究；

软件开发：需求分析、概要设计、详细设计、编码、测试；

软件运行：软件维护

**三种面向对象模型的主要功能**

①对象模型：表示了静态的结构化的系统数据性质，描绘了系统的静态结构，从客观世界实体的对象关系角度来描绘对象。

②动态模型：该模型描述了系统的控制结构，它表示了瞬间的、行为化的系统控制性质，它关心的是系统的控制及操作的执行顺序，它从对象的事件和状态的角度出发，表现了对象的交互行为。

③功能模型：表示变化的系统“功能”性质，它指明系统应该“做什么”，因此功能模型更直接的反映了用户对目标系统的要求。

调试则是在进行了成功的测试之后才开始的工作。调试的目的是确定错误的原因和位置，并改正错误，因此调试也称为纠错(Debug)。调试的技术手段有简单的调试方法、归纳法、演绎法和回溯法等。

**《软件工程》题集**

## 一、选择题（在每小题的四个备选答案中，选出一个正确答案，并将正确答案的序号填在题干的括号内，每小题1分）

1.软件产品的开发主要是( D  )。

 A.复制 B.再生产 C.拷贝 D.研制

2.作坊式小团体合作生产方式的时代是( C )时代。

  A.程序设计 B.软件生产自动化 C.程序系统 D.软件工程

3.软件工程与计算机科学性质不同，软件工程着重于(  C )。

  A.理论研究 B.原理探讨 C.建造软件系统 D.原理的理论

4.将每个模块的控制结构转换成计算机可接受的程序代码是(  A )阶段的任务。

 A.编码 B.需求分析 C.详细设计 D.测试

5.可行性研究的目的是决定(   B )。

  A.开发项目 B.项目值得开发否 C.规划项目 D.维护项目

6．SA方法用DFD描述(  D ) 。

 A．系统的控制流程 B．系统的数据结构

 C．系统的基本加工 D．系统的功能

7.进行需求分析可使用多种工具，但(  C )是不适用的。

 A. 数据流图(DFD) B. 判定表 C. PAD图  D. 数据字典

8.结构化分析（SA）方法将欲开发的软件系统分解为若干基本加工，并对加工进行说明，下述是常用的说明工具，其中便于对加工出现的组合条件的说明工具是(  A )。a.结构化语言；b.判定树；c.判定表。

 A. b和c B. a,b和c C. a和c D. a和b

 9.软件概要设计结束后得到(   B )。

A.初始化的软件结构图    B.优化后的软件结构图

C.模块详细的算法         D.程序编码

10.软件设计阶段一般又可分为 (  B  )。

 A.逻辑设计与功能设计 B.概要设计与详细设计

C.概念设计与物理设计 D.模型设计与程序设计

11.软件设计中划分模块的一个准则是( C )。

A.低内聚低耦合 B.低内聚高耦合

C.高内聚低耦合 D.高内聚高耦合

12.PDL是软件开发过程中用于(  C  )阶段的描述工具。

  A.需求分析 B.概要设计 C.详细设计 D.编程

13.下列语言中哪个语言是低级语言(  D )

  A. PASCAL B.VISUAL BASIC  C. [C++](http://www.educity.cn/incsearch/search.asp?key=C%2B%2B)  D.[汇编语言](http://www.educity.cn/incsearch/search.asp?key=%BB%E3%B1%E0%D3%EF%D1%D4)

14.软件测试的目的是(  B )。

 A.试验性运行软件 B.发现软件错误

C.证明软件正确         D.找出软件中全部错误

15.软件测试中,白盒法是通过分析程序的( B  )来设计测试用例的。

  A.应用范围 B.内部逻辑 C.功能 D.输入数据

 16.黑盒法是根据程序的( C  )来设计测试用例的。

  A.应用范围 B.内部逻辑 C.功能 D.输入数据

 17.下面的逻辑测试覆盖中，测试覆盖最弱的是(  C )

  A.条件覆盖                  B.条件组合覆盖  C.语句覆盖                  D.条件及判定覆盖

18.把一组具有相似特性的对象组合在一起，称之为( C )。

A、多态性 B、继承性 C、类 D、消息

19.面向对象的主要特征除对象唯一性、封装、继承外，还有（ A ）

A、多态性              B、完整性 C、可移植性            D、兼容

20.下列模型属于成本估算方法的有( A )

A、COCOMO模型 B、McCall模型

C、McCabe度量法 D、时间估算法

21.软件生存周期中时间最长的是(  D )阶段。

 A.需求分析    B.概要设计    C.测试         D.维护

22.与计算机科学的理论研究不同，软件工程是一门(  C  )学科。

 A.理论性   B.原理性      C.工程性      D.心理性

23.从结构化的瀑布模型看，在它的生命周期中的八个阶段中，下面的几个选项中哪个环节出错，对软件的影响最大(  C )。

 A、详细设计阶段         B、概要设计阶段        C、 需求分析阶段     D、 测试和运行阶段

24. 程序的三种基本控制结构的共同特点是(  D  )。

 A.不能嵌套使用             B.只能用来写简单程序  C.已经用硬件实现           D.只有一个入口和一个出口

25.软件工程的出现主要是由于(  C )。

 A.程序设计方法学的影响                 B.其它工程科学的影响    C. 软件危机的出现                    D.计算机的发展

26.在[软件工程](http://www.educity.cn/incsearch/search.asp?key=%C8%ED%BC%FE%B9%A4%B3%CC)项目中，不随参与人数的增加而使软件的生产率增加的主要问题是(  D )

 A、 工作阶段间的等待时间     B、 生产原型的复杂性  C、 参与人员所需的工作站数   D、 参与人员之间的通信困难

27. SA方法的基本思想是(  C  )

A．自底向上逐步抽象　　　B．自底向上逐步分解 C．自顶向下逐步分解　　　D．自顶向下逐步抽象

28.判定表和判定树是数据流图中用以描述加工的工具，它常描述的对象是(  D  )

 A. 逻辑判断                          B. 层次分解  C. 操作条目                          D. 组合条件

29.软件需求说明书是软件需求分析阶段的重要文件，下述哪个(些)是其应包含的内容(  D )

    a.数据描述

b.功能描述

c.模块描述

d.性能描述

　A. b 　　　　                       B. c和d 　C. a，b和c 　　                     D. a，b和d

30.软件结构图中,模块框之间若有直线连接,表示它们之间存在着(  A  )关系。

 A.调用           B.组成             C.链接          D.顺序执行

31.软件设计中一般将用到图形工具，下列哪种(些)可用作设计的图形工具(  C )

 a.结构图

 b.实体联系图

 c.IPO图

 d.层次图

 A、 a和b                   B、 c和d  C、 a，c，d                D、 全部

32. 程序设计语言一般简单来说，可划分为低级语言和高级语言两大类，与高级语言相比，用低级语言开发的程序，具有如下哪种特点( C )

 A、 运行效率低，开发效率低  B、 运行效率低，开发效率高

C、 运行效率高，开发效率低  D、 运行效率高，开发效率高

33. 下面的叙述哪些是正确的(  C )

(1)在软件开发过程中，编程作业的代价最高。

(2)良好的程序设计风格应以缩小程序占用的存储空间和提高程序的运行速度为原则。

(3)为了提高程序的运行速度，有时采用以存储空间换取运行速度的方法。

(4)对同一算法，用高级语言编写的程序比用低级语言编写的程序运行速度快。

(5)COBOL是一种非过程型语言。

(6)LISP是一种逻辑型程序设计语言。

 A、 (1)、(3)、(5)             B、 (2)、(3)、(4)

C、 (3)                       D、 (4)、(6)

34、为了提高软件测试的效率,应该(  D  )。

 A.随机地选取测试数据

 B.取一切可能的输入数据作为测试数据

 C.在完成编码以后制定软件的测试计划

 D.选择发现错误可能性较大的数据作为测试用例

35. 与设计测试用例无关的文档是(  A  )。

 A.项目开发计划        B.需求规格说明书  C.设计说明书          D.源程序

 36. 使用白箱测试方法时确定测试数据应根据(  A )和指定的覆盖标准。

 A.程序的内部逻辑             B.程序的复杂程度  C.使用说明书                 D.程序的功能

37. 软件部件的内部实现与外部可访问性的分离,是指软件的(  C  )。

 A.继承性        B.共享性         C.封装性          D.抽象性

38、一个面向对象系统的体系结构通过它的成分对象和对象间的关系确定，与传统的面向数据流的结构化开发方法相比，它具有的优点是( A ) .

A、设计稳定            B、变换分析 C、事务分析            D、模块独立性

39、软件管理的主要职能包括(  D )

 A.人员管理、计划管理　        B.[标准](http://www.educity.cn/incsearch/search.asp?key=%B1%EA%D7%BC)化管理、[配置管理](http://www.educity.cn/incsearch/search.asp?key=%C5%E4%D6%C3%B9%DC%C0%ED)  C.[成本管理](http://www.educity.cn/incsearch/search.asp?key=%B3%C9%B1%BE%B9%DC%C0%ED)、[进度管理](http://www.educity.cn/incsearch/search.asp?key=%BD%F8%B6%C8%B9%DC%C0%ED)　        D.(A)和(B)

40、任何项目都必须精心做好项目管理工作，最常用的计划管理工具是(  D )

 A.数据流程图　                B.程序结构图  C.因果图　　　                D.PERT图

41. 软件工程方法学的目的是：使软件生产规范化和工程化，而软件工程方法得以实施的主要保证是(  C )

 A、 硬件环境               B、软件开发的环境     C、软件开发工具和软件开发的环境                 D、 开发人员的素质

42. 在下面的软件开发方法中，哪一个对软件设计和开发人员的开发要求最高( B )

 A、结构化方法          B、原型化方法          C、面向对象的方法    D、控制流方法

43. 软件开发中常采用的结构化生命周期方法，由于其特征而一般称其为(  A )

 A、 瀑布模型     B、 对象模型     C、 螺旋模型      D、 层次模型

44. 对于分层的DFD，父图与子图的平衡指子图的输入、输出数据流同父图相应加工的输入、输出数据流(   A )。

 A．必须一致     B．数目必须相等   C．名字必须相同    D．数目必须不等

45. 软件需求分析阶段的工作，可以分为四个方面：对问题的识别.分析与综合.编写需求分析文档以及(  B )

A. 软件的总结                       B. 需求分析评审   C. 阶段性报告                       D. 以上答案都不正确

46．在结构化分析方法中，用以表达系统内数据的运动情况的工具有(  B  )

  A. 数据流图 　　       　           B. 数据字典   C. 结构化语言　　                   D. 判定表与判定树

47. 为了提高模块的独立性，模块内部最好是( C  )。

A.逻辑内聚    B.时间内聚     C.功能内聚    D.通信内聚

48. 面向数据流的设计方法把（ C ）映射成软件结构。

A、数据流           B、模块化        C、控制结构         D、信息流

49. .数据流图是常用的进行软件需求分析的图形工具，其基本图形符号是(  C )

 A. 输入.输出.外部实体和加工   B. 变换.加工.数据流和[存储](http://www.educity.cn/incsearch/search.asp?key=%B4%E6%B4%A2)

 C. 加工.数据流.数据存储和外部实体  D. 变换.数据存储.加工和数据流

50. 软件结构使用的图形工具,一般采用(  C )图。

 A.DFD            B.PAD              C.SC            D.ER

51. 在面向数据流的软件设计方法中，一般将信息流分为(  A )

 A、 变换流和事务流         B、 变换流和控制流

 C、 事务流和控制流         D、 数据流和控制流

52. 程序的三种基本控制结构是(  B  )。

 A.过程、子程序和分程序     B.顺序、选择和重复

C.递归、堆栈和队列         D.调用、返回和转移

53. 白箱测试方法重视( A )的度量。

 A.测试覆盖率                 B.测试数据多少  C.测试费用                   D.测试周期

54、软件的集成测试工作最好由( D )承担，以提高集成测试的效果。

 A.该软件的设计人员  B.该软件开发组的负责人  C.该软件的编程人员  D.不属于该软件开发组的软件设计人员

55. 据国家标准GB8566-8计算机软件开发的规定，软件的开发和维护划分为八个阶段，其中组装测试的计划是在( C )阶段完成的？

 A.可行性研究和计划           B.需求分析  C.概要设计                   D.详细设计

 56. 软件生命周期的最后的一个阶段是(  B  )

 A.书写[软件文档](http://www.educity.cn/incsearch/search.asp?key=%C8%ED%BC%FE%CE%C4%B5%B5)          B.软件维护  C.稳定性测试            D.书写详细用户说明

57. McCall提出了表明软件质量的11个质量特性。它们是正确性、可靠性、可使用性、完整性、( D ) 、可移植性、可复用性、效率、可测试性、互联性。

 A.模块独立性、[安全](http://www.educity.cn/incsearch/search.asp?key=%B0%B2%C8%AB)性                       B.文档完备性、灵活性

 C.可维护性、安全性                          D.可维护性、灵活性

58、每个对象可用它自己的一组属性和它可以执行的一组（ C  ）来表征。

A、行为                B、功能 C、操作                D、数据

59、面向对象软件技术的许多强有力的功能和突出的优点，都来源于把类组织成一个层次结构的系统，一个类的上层可以由父亲，下层可以有子类，这种层次结构系统的一个重要性质是（ B ），一个类获得其父亲的全部描述（数据和操作）

A、传递性              B、继承性 C、复用性              D、并行性

60、软件开发在需求分析、设计、编码、测试这几个阶段所需不同层次的技术人员大致是( C )

 A.初级、高级、高级、初级　　  B.中级、中级、高级、中级

C.高级、中高级、初级、中高级  D.中级、中高级、中级、初级

61、软件重用的单位是(  A  )。

 A.软件模块 B.性能        C.系统        D.功能

62、下列不属于软件工程方法学三要素的是( D )：

A、方法 B、工具 C、过程 D、操作

63、软件开发中大约要付出(  C )％的工作量进行测试和排错。

 A.20       B.30          C.40          D.50

64、软件开发常使用的两种基本方法是结构化和原型化方法，在实际的应用中，它们之间的关系表现为 (  B )

 A、 相互排斥            B、 相互补充           C、 独立使用        D、 交替使用

65、系统分析员是( B )

A、用户中系统的直接使用者 B、用户和软件人员的中间人

C、软件的编程人员 D、用户和软件人员的领导

66、结构化分析SA方法以数据流图、( D )和加工说明等描述工具，即用直观的图和简洁的语言来描述软系统模型。

A、DFD图 B、PAD图 C、IPO图 D、数据字典

67、制定软件计划的目的在于尽早对欲开发的软件进行合理估价，软件计划的任务是(  D )

 A、 组织与管理               B、 分析与估算

 C、 设计与测试               D、 规划与调度

68、在E-R 模型中，包含以下基本成分( C )。

A、数据、对象、实体 B、控制、联系、对象

C、实体、联系、属性 D、对象、属性、操作

69、结构设计是一种应用最广泛的系统设计方法，是以( A )为基础、自顶向下、逐步求精和模块化的过程。

A、数据流 B、数据流图 C、数据库 D、数据结构

70、如果某种内聚要求一个模块中包含的任务必须在同一段时间内执行，则这种内聚为( A )。

A、时间内聚　　B、逻辑内聚　　C、通信内聚　　D、信息内聚

71、初步用户手册在(   B )阶段编写。

A.可行性研究　B．需求分析　C．软件概要设计　D．软件详细设计

72、在软件结构化设计中，好的软件结构设计应该力求做到( B )。

A、顶层扇出较少，中间层扇出较高，底层模块低扇入

B、顶层扇出较高，中间层扇出较少，底层模块高扇入

C、顶层扇入较少，中间层扇出较高，底层模块高扇入

D、顶层扇入较少，中间层扇入较高，底层模块低扇入

73、结构图(SC)和层次方框图是不同的，前者 ( B )。

A、描述程序的过程特性 B、描述程序的层次特性

C、描述程序的过程特征和层次特征 D、即不描述程序的过程特征，也不描述程序的层次特征

74、软件详细设计主要采用的方法是( D )。

A、模块设计 B、结构化设计 C、PDL语言 D、结构化程序设计

75、程序的三种基本控制结构是( B )

A、过程、子程序和分程序　　B、顺序、选择和循环 C、递归、堆栈和队列　 D、调用、返回和转移

76、软件设计将涉及软件的构造、过程和模块的设计，其中软件过程是指( B )

 A、 模块间的关系           B、 模块的操作细节

 C、 软件层次结构           D、 软件开发过程

77、软件测试的手段有多种，通过人工来评审文档或程序，借以发现其中错误，该手段是( D )。

A、黑盒测试 B、正确性测试 C、动态测试 D、静态测试

78、黑盒测试在设计测试用例时，主要需要研究( A ) 。

A、需求规格说明与概要设计说明 B、详细设计说明

C、项目开发计划 D、概要设计说明与详细设计说明

79、在整个软件维护阶段所花费的全部工作中，( C ) 所占比例最大。

A、改正性维护 B、适应性维护 C、完善性维护 D、预防性维护

80、下列哪个改变不属于软件维护中的副作用？( C )

 A.修改代码                    B.修改数据  C.修改需求分析                D.修改文档

81.作坊式小团体合作生产方式的时代是( C  )时代。

 A.程序设计    B.软件生产自动化 C.程序系统  D.软件工程

82.软件需求分析阶段的工作，可以分为以下4个方面：对问题的识别、分析与综合、编写需求分析文档以及( C )。

A. 总结 B. 阶段性报告 C. 需求分析评审 D. 以上答案都不正确

83.在结构化分析方法中，用以表达系统内数据的运动情况的工具有( A )。

A. 数据流图 B. 数据词典 C. 结构化语言 D. 判定表与判定树

84.如果一个软件是给许多客户使用的，大多数软件厂商要使用几种测试过程来发现那些可能只有最终用户才能发现的错误，( A )测试是由一个用户在开发者的场所来进行的.测试的目的是寻找错误的原因并改正之。

A. α 　　　　　B. β　　　 　　C. γ 　　　　D. σ

85.来辅助软件开发、运行、维护、管理、支持等过程中的活动的软件称为软件开发工具,通常也称为( D )工具。

A.CAD B.CAI C.CAM D.CASE

86. 螺旋模型综合了( A )的优点,并增加了风险分析。

A.瀑布模型和演化模型 B.瀑布模型和喷泉模型

C.演化模型和喷泉模型 D.原型和喷泉模型

87．SA方法的分析步骤是首先调查了解当前系统的工作流程，然后(   A )。

A．获得当前系统的物理模型，抽象出当前系统的逻辑模型，建立目标系统的逻辑模型

B．获得当前系统的物理模型，抽象出目标系统的逻辑模型，建立目标系统的物理模型

C．获得当前系统的物理模型，建立当前系统的物理模型，抽象出目标系统的逻辑模型

D．获得当前系统的物理模型，建立当前系统的物理模型，建立目标系统的物理模型

88. 在软件需求规范中，下述哪些要求可以归类为过程要求(  A )

A. 执行要求                         B. 效率要求 C. 可靠性要求                       D. 可移植性要求

89. 数据流图是常用的进行软件需求分析的图形工具，其基本图形符号是(  C )

A. 输入.输出.外部实体和加工  B. 变换.加工.数据流和[存储](http://www.educity.cn/incsearch/search.asp?key=%B4%E6%B4%A2)

C. 加工.数据流.数据存储和外部实体  D. 变换.数据存储.加工和数据流

90. 设计软件结构一般不确定(  C )。

A.模块的功能            B.模块的接口

C.模块内的局部数据      D.模块间的调用关系

91. 模块的独立性是由内聚性和耦合性来度量的，其中内聚性是(  B )

A、 模块间的联系程度       B、 模块的功能强度

C、 信息隐蔽程度           D、 接口的复杂程度

92. Jackson方法实现从(  A  )导出(   C )。

A.[数据结构](http://www.educity.cn/incsearch/search.asp?key=%CA%FD%BE%DD%BD%E1%B9%B9)     B.数据流图       C.程序结构      D.软件模块层次结构

93.下列语言中哪个语言是低级语言(  D )

 A、 PASCAL             B、 VISUAL BASIC  C、 [C++](http://www.educity.cn/incsearch/search.asp?key=C%2B%2B)                D、 [汇编语言](http://www.educity.cn/incsearch/search.asp?key=%BB%E3%B1%E0%D3%EF%D1%D4)

94、PAD图的控制执行流程为( C )

 A、自下而下、从左到右        B、自上而下、循环执行

 C、自上而下、从左到右        D、都不对

95. 程序设计语言的心理特性在语言中表现不应包括(  C )。

 A.歧义性       B.简洁性        C.保密性        D.传统性

 96. 在软件工程中，白箱测试方法可用于测试程序的内部结构。此方法将程序可作为( A  )

 A.路径的集合                B.循环的集合  C.目标的集合                D.地址的集合

97.集成测试的主要方法有两个，一个是\_\_\_\_\_一个是\_\_\_\_\_ (  B  )

 A.白箱测试方法、黑箱测试方法

B.渐增式测试方法、非渐增式测试方法

C.等价分类方法、边缘值分析方法

D.因果图方法、错误推测方法

98、软件维护工作的最主要部分是(  C  )

A.校正性维护            B.适应性维护  C.完善性维护            D.预防性维

99、软件产品质量是生产者和用户都十分关心的问题，质量管理只看到产品的质量，近年来质量管理向(  A )发展，重要的基本假设是过程的质量直接影响产品的质量。

 A.过程质量的控制                   B.过程的改进

C.产品的改进                       D.技术的革新

100．结构分析方法就是面向（ D）自顶向下逐步求精进行需求分析的方法。

A．目标 B.功能 C. 对象 D.数据流

101. 软件是一种(  B  )产品。

 A.物质       B.逻辑      C.有形        D.消耗

102. 瀑布模型把软件生命周期划分为八个阶段：问题的定义、可行性研究、软件需求分析、系统总体设计、详细设计、编码、测试和运行、维护。八个阶段又可归纳为三个大的阶段：计划阶段、开发阶段和(  C )。

A、详细计划            B、可行性分析          C、 运行阶段         D、 测试与排错

103. 结构化分析方法是一种预先严格定义需求的方法，它在实施时强调的是分析对象的( B )

 A、控制流              B、数据流              C、程序流            D、指令流

104. 在软件的可行性研究中，可以从不同的角度对软件的可行性进行研究，其中是从软件的功能可行性角度考虑的是( B )

 A、 经济可行性               B、技术可行性  C、 操作可行性               D、 法律可行性

105. 需求规格说明书的内容还应包括对(  B )的描述。

 A．主要功能 B．算法的详细过程 C．用户界面及运行环境 D．软件的性能

106、系统测试是将软件系统与硬件、外设和网络等其他因素结合，对整个软件系统进行测试。(   A )不是系统测试　的内容。

A．路径测试　　B．可靠性测试　　　C．安装测试　　　 D．安全测试

107、在结构化分析方法中，数据字典是重要的文档。对加工的描述是数据字典的组成内容之一，常用的加工描述方法(  C )。

A．只有结构化语言　　　　　　　　 B．有结构化语言和判定树

C．有结构化语言、判定树和判定表　 D．有判定树和判定表

108、项目管理工具中，将网络方法用于工作计划安排的评审和检查的是(   A )。

A．Gantt图　　 B．PERT网图　　　　C．因果分析图　　 D．流程图

109、软件设计的主要任务是设计软件的结构、过程和模块，其中软件结构设计的主要任务是要确定(  C )。

A．模块间的操作细节　　　　　　　 B．模块间的相似性

C．模块间的组成关系　　　　　　　 D．模块的具体功能

110、下述任务中，不属于软件工程需求分析阶段的是(   D ) 。

A．分析软件系统的数据要求　　　　 B．确定软件系统的功能需求

C．确定软件系统的性能要求　　　　 D．确定软件系统的运行平台

111. 加工是对数据流图中不能再分解的基本加工的精确说明，下述哪个是加工的最核心(  B )

 A. 加工顺序                          B. 加工逻辑

 C. 执行频率                          D. 激发条件

112. 为了提高模块的独立性，模块内部最好是(  C )。

A.逻辑内聚    B.时间内聚     C.功能内聚    D.通信内聚

113. PDL是软件开发过程中用于(  C  )阶段的描述工具。

 A.需求分析   B.概要设计    C.详细设计    D.编程

114、适合在互联网上编写程序可供不同平台上运行的面向对象程序设计语言是( B )

 A、 ALGOL              B、 JAVA C、 SMALLTALK          D、 LISP

115. 软件测试是软件质量保证的主要手段之一，测试的费用已超过( A  )的30%以上，因此提高测试的有效性非常重要。

 A.软件开发费用                     B.软件维护费用  C.软件开发和维护费用               D.软件研制费用

 116. 在软件测试中，逻辑覆盖标准主要用于( B )

 A.黑箱测试方法              B.白箱测试方法

 C.灰箱测试方法              D.软件验收方法

 117. 在软件维护工作中进行的最少的部分是( D )

 A.校正性维护            B.适应性维护  C.完善性维护            D.预防性维护

118、火车是一种陆上交通工具,火车和陆上交通工具之间的关系是( D )关系。

 A.组装          B.整体成员        C.has a         D.一般具体

119、动态模型的描述工具是(  C  )。

 A.对象图        B.结构图         C.状态图          D.设计图

120、McCall提出的软件质量的11个质量特性分为三组，下面哪一个不属于这三组( A )

 A.产品需求                     B.产品运行  C.产品修正                        D.产品转移

121.瀑布模型的存在问题是（ B ）

A．用户容易参与开发 B．缺乏灵活性

C．用户与开发者易沟通 D．适用可变需求

122.开发软件所需高成本和产品的低质量之间有着尖锐的矛盾，这种现象称做( C )

A.软件工程 B.软件周期

C.软件危机 D.软件产生

123.数据耦合、公共耦合、标记耦合、控制耦合的耦合性从低到高的顺序是（ B ）

A.数据、公共、标记、控制 B.数据、标记、控制、公共

C.控制、数据、标记、公共 D.控制、数据、公共、标记

124.在SD方法中全面指导模块划分的最重要的原则是( D )

A.程序模块化 B.模块高内聚

C.模块低耦合 D.模块独立性

125．软件测试的目的是（ B ）。

A． 评价软件的质量 B. 发现软件的错误

C． 找出软件的所有错误 D. 证明软件是正确的

126．在设计测试用例时，（ A ）是用得最多的一种黑盒测试方法。

A． 等价类划分 B. 边界值分析 C. 因果图 D. 判定表

127. 需求分析最终结果是产生( B )。

A. 项目开发计划 B. 需求规格说明书

C. 设计说明书 D. 可行性分析报告

128. Jackson图中，模块框之间若有直线连接，表示它们之间存在( B )。

A. 调用关系 B. 组成关系 C. 链接关系 D. 顺序执行关系

129. 软件详细设计的主要任务是确定每个模块的( C )。

A. 功能 B. 外部接口 C. 算法和使用的数据结构 D. 编程

130．为了提高软件的可维护性，在编码阶段应注意（ D ）

A.保存测试用例和数据 B.提高模块的独立性

C.文档的副作用 D.养成好的程序设计风格

131．设年利率为i，现存入p元，若计复利，n年后可得钱数为（　B　）

A．p﹡(1+i﹡n) B．p﹡(i+1)n

C．p﹡(1+i)﹡n D．p﹡(i+n)

132．在考察系统的一些涉及时序和改变的状态时，要用动态模型来表示。动态模型着重于系统的控制逻辑，它包括两个图：一个是事件追踪图，另一个是（ A ）。

A ．状态图 B. 数据流图 C. 系统结构图 D. 时序图

133. 对象实现了数据和操作的结合，使数据和操作( C )于对象的统一体中。

A. 结合 B. 隐藏 C. 封装 D. 抽象

134．具有风险分析的软件生存周期模型是（ C ）

A.瀑布模型 B.喷泉模型

C.螺旋模型 D.增量模型

135.软件生命周期中所花费用最多的阶段是（ D ）

A．详细设计 B．软件编码

C．软件测试 D．软件维护

136. 在整个软件维护阶段所花费的全部工作中，（ C ）所占比例最大。

A． 校正性维护 B. 适应性维护 C. 完善性维护 D. 预防性维护

137．模块（ B ）定义为受该模块内一个判断影响的所有模块集合。

A． 控制域 B. 作用域 C.宽度 D. 接口

138．指出PDL是下列哪种语言（ A ）。

A．伪码 B、高级程序设计语言

C． 中级程序设计语言 D、 低级程序设计语言

139．在对数据流的分析中，主要是找到中心变换，这是从（ C ）导出结构图的关键。

A． 数据结构 B.实体关系 C. 数据流图 D. E—R图

140．结构分析方法就是面向（ B ）自顶向下逐步求精进行需求分析的方法。

A． 目标 B.数据流 C.功能 D. 对象

141.开发软件所需高成本和产品的低质量之间有着尖锐的矛盾，这种现象称做( C )

A.软件工程 B.软件周期

C.软件危机 D.软件产生

142.快速原型模型的主要特点之一是( D )

A.开发完毕才见到产品 B.及早提供全部完整的软件产品

C.开发完毕后才见到工作软件 D.及早提供工作软件

143.模块的内聚性最低的是( C )

A.逻辑内聚 B.时间内聚

C.偶然内聚 D.功能内聚

144.软件详细设计主要采用的方法是( D )

A.模块设计 B.结构化设计

C.PDL语言 D.结构化程序设计

145.程序的三种基本控制结构是（ B ）

A．过程、子程序和分程序

B．顺序、选择和重复

C．递归、堆栈和队列

D．调用、返回和转移

146.软件生命周期中所花费用最多的阶段是（ D ）

A．详细设计 B．软件编码

C．软件测试 D．软件维护

147．通过（ B ）可以完成数据流图的细化。

A． 结构分解 B、 功能分解 C、 数据分解 D、 系统分解

148．Jackson方法是面向（ B ）的设计方法。

A． 系统设计 B、 数据结构 C、 DFD D、 OO

149．软件测试的目的是（ B ）。

A． 评价软件的质量 B. 发现软件的错误

C． 找出软件的所有错误 D. 证明软件是正确的

150．在考察系统的一些涉及时序和改变的状态时，要用对象-行为模型来表示。动态模型着重于系统的控制逻辑，它包括三个图：一个是事件轨迹图和事件流图，另一个是（ A ）。

A .状态转换图 B. 数据流图 C. 系统结构图 D. 时序图

151．结构分析方法就是面向（ B ）自顶向下逐步求精进行需求分析的方法。

A． 目标 B.数据流 C.功能 D. 对象

152．在多层次的结构图中，其模块的层次数称为结构图的（ A ）。

A． 深度 B. 跨度 C. 控制域 D. 粒度

153. 研究开发所需要的成本和资源是属于可行性研究中的( B )研究的一方面。

A. 技术可行性 B. 经济可行性 C. 社会可行性 D. 法律可行性

154. 软件详细设计的主要任务是确定每个模块的( A )。

A. 算法和使用的数据结构 B. 外部接口 C. 功能 D. 编程

155. 软件结构图中，模块框之间若有直线连接，表示它们之间存在( A )。

A. 调用关系 B. 组成关系 C. 链接关系 D. 顺序执行关系

156. 需求分析最终结果是产生( B )。

A. 项目开发计划 B. 需求规格说明书

C. 设计说明书 D. 可行性分析报告

157. 在详细设计阶段，经常采用的工具有( A )。

A. PAD B. SA C. SC D. DFD

158.因计算机硬件和软件环境的变化而作出的修改软件的过程称为( C )

A.教正性维护 B.适应性维护

C.完善性维护 D.预防性维护

159. 对象实现了数据和操作的结合，使数据和操作( C )于对象的统一体中。

A. 结合 B. 隐藏 C. 封装 D. 抽象

160．为了提高软件的可维护性，在编码阶段应注意（ D ）

A.保存测试用例和数据 B.提高模块的独立性

C.文档的副作用 D.养成好的程序设计风格

**《软件工程》填空题**

## 二、填空题（请把答案写在相应的横线上,每小题1.5分）

1、软件是 数据 、 计算机程序 及其说明程序的 各种文档 。

2、概要设计主要是把各项 功能需求 转换成系统的 体系结构 。

3、面向对象开发方法包括 OOA 、OOD 、 OOP 三部分。

4、结构化设计中以数据流图为基础的两种具体分析设计方法是 变换分析 、 事物分析 设计。

5、在单元测试时，需要为被测模块设计 驱动模块 和 桩模块 。

6、CMM把软件过程从无序到有序的进化分成5个阶段，排序而形成5个逐层提高的等级，分别是初始级、可重复级 、已定义级、 已管理级 和可优化级。

7.子类自动共享父类的 属性 和 操作 的机制称为继承。

8. 软件工程管理的具体内容包括对开发人员、组织机构、 用户 、 文档资料 等方面的管理。

9、可行性研究的三个方面是 技术可行性 、 社会可行性 和\_\_经济可行性\_\_。

10、在软件概要设计阶段，建立软件结构后，还应为每个模块写一份 处理说明 和\_接口说明\_\_。

11、在画分层的DFD时，父图与子图的输入输出数据流要\_\_平衡\_\_。

12、在详细设计阶段，除了对模块内的算法进行设计，还应对模块内的\_\_数据结构\_进行设计。

13. 对象的抽象是\_\_\_类\_\_\_。

14. 基线的作用是把各阶段的开发工作划分得更加明确，便于检查与确认阶段成果。因此，基线可以作为项目的一个\_\_\_检查点\_\_。

15. [软件工程](http://www.educity.cn/incsearch/search.asp?key=%C8%ED%BC%FE%B9%A4%B3%CC)包括软件开发技术和\_\_软件工程管理\_\_两大部分内容。

16、开发过程管理包括项目计划、控制和\_\_\_任务管理\_\_等。

17、CASE是多年来在软件开发管理、软件开发方法、软件开发环境和\_\_软件工具\_\_等方面研究和发展的产物。

18、数据字典中有四类条目，分别是\_\_\_数据流 、数据项、 数据存储 、基本加工。

19、用于描述基本加工的小说明的三种描述工具是结构化语言 、 判定表 、 判定树\_。

20、子类只继承一个父类的属性和操作,这称为\_\_单重继承\_\_。

21、 McCabe复杂性度量又称\_\_环路度量\_。

22、喷泉模型是一种以用户需求为动力，以\_\_对象\_\_为驱动的模型。

23、Jackson图不仅可表示程序结构，还可表示\_\_\_数据结构\_\_。

24、详细设计的任务是确定每个模块的内部特性，即模块的 算法 和 数据结构 。

25、UML提供的构建系统动态模型的机制有 动态视图 视图（包括顺序图和协作图）、 静态视图 视图、功能视图。

26、软件质量管理由软件质量保证、 软件质量规划 和 软件质量控制 三方面活动组成。

27、用等价类划分法设计一个测试用例时,使其覆盖\_\_尽可能多的\_\_尚未被覆盖的合理等价类; 覆盖\_\_一个\_不合理等价类。

28、需求分析阶段产生的最重要的文档是\_\_需求规格说明书\_。

29、两个模块都使用同一张表，模块的这种耦合称为\_\_\_公共耦合\_\_。

30、被测试程序不在机器上运行,而是采用人工检测和计算机辅助分析检测的手段称为\_静态\_测试。

31、采用手工方法开发软件只有程序而无文档，维护困难，这是一种\_\_非结构化\_\_维护。

32、结构化方法是一种面向数据流的开发方法。由结构化分析、\_结构化方法\_、结构化程序设计构成。

33、 对象之间进行通信的构造叫做\_消息\_。

34、 按软件服务对象的范围可以把软件划分为 项目 软件和 产品 软件。

35、瀑布模型将软件生命期划分为若干阶段，各个阶段之间是通过 文档 实现过渡的。

36、概要设计主要是把各项功能需求转换成系统的\_\_体系结构\_\_。

37、详细设计的任务是确定每个模块的内部特性，即模块的 数据结构 和 算法 。

38、结构化程序设计方法提倡开发人员的组织形式是\_\_主程序员组\_。

39、软件质量管理由软件质量保证、软件质量规划和 软件质量控制 三方面活动组成。

40、类具有操作,它是对象\_\_行为  \_的抽象。

41、产品管理包括版本管理和\_\_配置管理\_\_\_。

42、集成测试中的具体方法是\_\_\_渐增式 和 非渐增式\_\_\_测试方法。

43、增量模型适合于\_\_软件需求\_不明确、设计方案有一定风险的软件项目。

44、结构化分析方法的分析策略是\_\_自顶向下逐步求精\_\_。

45、可行性研究实质上是要在较高层次上以较抽象的方式进行\_\_\_需求分析和设计\_\_的过程。

46、衡量模块独立性的两个定性标准是\_\_耦合性与内聚性\_\_\_。

47、Jackson方法是一种面向\_\_数据结构\_\_的设计方法。

48、维护阶段是软件生存周期中花费精力和费用\_\_\_最多\_\_\_的阶段。

49、继承性是子类自动共享其父类的\_\_数据结构和方法\_\_机制。

50、在面向对象方法中，人们常用状态图描述类中对象的\_动态行为\_。

51、规定功能的软件，在一定程度上能从错误状态自动恢复到正常状态，则称该软件为\_\_\_容错\_\_软件。

52.软件工程包括3个要素，即方法、工具、过程。

53．软件生存周期是指一个软件从提出开发要求开始直到该软件报废不止的整个时期。通常，软件生存周期包括问题定义、可行性研究，需求分析，总体设计， 详细设计 ，\_编码\_，测试，维护活动，可以将这些活动以适当方式分配到不同阶段去完成

54．数据流图中的箭头表示\_\_数据流\_\_。软件结构图中的箭头方向表示\_\_\_模块调用\_\_。

55．\_\_\_可行性研究\_\_\_的目的就是用最小的代价在尽可能短的时间内确定该软件项目是否能够开发，是否值得去开发。

56.数据流图仅反映系统必须完成的逻辑功能，所以它是一种\_\_功能\_模型。

## 三、简答题(简要回答下列问题，每小题4分\*5=20)

1、什么是软件工程？它目标和内容是什么？

软件工程就是用科学的知识和技术原理来定义，开发，维护软件的一门学科。

软件工程目标：付出较低开发成本；达到要求的功能；取得较好的性能；开发的软件易于移植；只需较低的维护费用；能按时完成开发任务，及时交付使用；开发的软件可靠性高。

软件工程内容：包括开发技术和开发管理两个方面。

2、软件开发中有哪几种过程模型？哪些适用于面向对象的软件开发？

软件开发中常见的软件过程模型有瀑布模型、原型模型、螺旋模型、喷泉模型、统一软件过程等。

其中喷泉模型、统一软件过程适用于面向对象的软件开发。

4、什么是软件概要设计？该阶段的基本任务是什么？  
把一个软件需求转换为软件表示时，首先设计出软件总的体系结构。称为概要设计或结构设计。

基本任务：

⑴设计软件系统结构

⑵进行数据结构及数据库的设计

⑶编写概要设计的文档

⑷评审

6、软件测试要经过哪些步骤?这些测试与软件开发各阶段之间有什么关系?（答案不是很好，书本上的V模型，画出V模型）

软件测试要经过的步骤是：单元测试→集成测试→确认测试→系统测试。

单元测试对源程序中每一个程序单元进行测试，检查各个模块是否正确实现规定的功能，从而发现模块在编码中或算法中的错误。该阶段涉及编码和详细设计文档。

集成测试是为了检查与设计相关的软件体系结构的有关问题，也就是检查概要设计是否合理有效。

确认测试主要是检查已实现的软件是否满足需求规格说明书中确定了的各种需求。

系统测试是把已确认的软件与其他系统元素(如硬件、其他支持软件、数据、人工等)结合在一起进行测试。以确定软件是否可以支付使用。

7、什么是软件生存周期？它有哪几个活动？

软件生存周期：一个软件从提出开发要求开始直到该软件报废为止的整个时期。

包括：可行性分析和项目开发计划，需求分析，概要设计，详细设计，编码，测试，维护。

8、衡量模块独立性的两个标准是什么？各表示什么含义？

内聚和耦合。

内聚：又称为块内联系，指模块内部各成分之间相互关联的程度，以高内聚为设计目标。

耦合：也称块间联系，模块之间相互联系程度的度量，联系越紧密，耦合性越强，独立性越差，以低耦合为设计目标。

10、什么是需求分析？需求分析阶段的基本任务是什么？

需求分析：开发人员准确地理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的需求规格说明的过程。

基本任务：

⑴问题识别

⑵分析与综合，导出软件的逻辑模型

⑶编写文档

12、单元测试有哪些内容?

单元测试主要针对模块的以下五个基本特征进行测试：

(1)模块接口

(2)局部数据结构

(3)重要的执行路径

(4)错误处理

(5)边界条件

14、软件设计的基本原理包括哪些内容？

⑴模块化：模块是软件的组成部分，是具有独立功能且可命名的一段程序，所有模块组成整体，可以满足问题的要求。

⑵抽象：认识复杂现象过程所使用权的工具，只考虑事物本质的共同特性，忽略细节和其它因素。通过抽象确定组成软件的过程实体。

⑶信息隐蔽：将模块实现自身功能的细节与数据"隐蔽"起来。模块间仅交换为完成系统功能所必须的信息。

⑷模块独立性：每个模块只完成系统要求的独立的子功能。

21、说明面向对象的特征？（答案错误，自己去整理答案）

面向对象的特征是：

(1)对象唯一性。每个对象都有自身唯一的标识，在对象生存期中，其标识不变，不同的对象不能有不同的标识。

(2)分类性。是指将具有一致的数据结构(属性)和行为(操作)的对象抽象成类

(3)继承性。是子类自动共享父类数据结构和方法的机制，这是类之间的一种关系。

(4)多态性。是指相同的操作或函数过程可以作用于多种类型的对象上并获得不同的结果。

24、请简述黑盒测试和白盒测试有何区别，黑盒测试有哪些常用方法？

黑盒测试和白盒测试的区别：前者基于功能，后者基于结构；

黑盒测试常用方法有：边界值、等价类、因果图、错误推测法等。

26、CMM全称是什么?分为哪几个级别？

CMM全称是软件能力成熟度模型；

从无序到有序的进化分成5个级别，分别为：

1 初始级；2 可重复级；3 已定义级；4 已管理级；5 优化级。

28、软件生命周期划分为哪几个阶段？

软件生命周期分为三个时期八个阶段：

软件定义：问题定义、可行性研究；

软件开发：需求分析、概要设计、详细设计、编码、测试；

软件运行：软件维护

**三、简答题(简要回答下列问题)**

1、什么是软件工程？它目标和内容是什么？

软件工程就是用科学的知识和技术原理来定义，开发，维护软件的一门学科。

软件工程目标：付出较低开发成本；达到要求的功能；取得较好的性能；开发的软件易于移植；只需较低的维护费用；能按时完成开发任务，及时交付使用；开发的软件可靠性高。

软件工程内容：包括开发技术和开发管理两个方面。

**2、软件开发中有哪几种过程模型？哪些适用于面向对象的软件开发？**

软件开发中常见的软件过程模型有瀑布模型、原型模型、螺旋模型、喷泉模型、统一软件过程等。

其中喷泉模型、统一软件过程适用于面向对象的软件开发。

**3、什么是结构分析方法？该方法使用什么描述工具？**

结构化分析：简称SA，面向数据流进行数据分析的方法。采用自顶向下逐层分解的分析策略。顶层抽象地描述整个系统，底层具体地画出系统工程的每个细节。中间层则是从抽象到具体的过渡。使用数据流图，数据字典，作为描述工具，使用结构化语言，判定表，判定树描述加工逻辑。

4、**什么是软件概要设计？该阶段的基本任务是什么？**  
把一个软件需求转换为软件表示时，首先设计出软件总的体系结构。称为概要设计或结构设计。

基本任务：

⑴设计软件系统结构

⑵进行数据结构及数据库的设计

⑶编写概要设计的文档

⑷评审

5、软件维护有哪些内容？

（1）校正性维护。在软件交付使用后，一些隐含的错误在某些特定的使用环境下会暴露出来。为了识别和纠正错误，修改软件性能上的缺陷，应进行确定和修改错误的过程，这个过程就称为校正性维护。

（2）适应性维护。为了使应用软件适应计算机硬件、软件环境及数据环境的不断发生的变化而修改软件的过程称为适应性维护。

（3）完善性维护。为增加软件功能、增强软件性能、提高软件运行效率而进行的维护活动称为完善性维护。

（4）预防性维护。为了提高软件的可维护性和可靠性而对软件进行的修改称为预防性维护。

6、软件测试要经过哪些步骤?这些测试与软件开发各阶段之间有什么关系?

软件测试要经过的步骤是：单元测试→集成测试→确认测试→系统测试。

单元测试对源程序中每一个程序单元进行测试，检查各个模块是否正确实现规定的功能，从而发现模块在编码中或算法中的错误。该阶段涉及编码和详细设计文档。

集成测试是为了检查与设计相关的软件体系结构的有关问题，也就是检查概要设计是否合理有效。

确认测试主要是检查已实现的软件是否满足需求规格说明书中确定了的各种需求。

系统测试是把已确认的软件与其他系统元素(如硬件、其他支持软件、数据、人工等)结合在一起进行测试。以确定软件是否可以支付使用。

**7、什么是软件生存周期？它有哪几个活动？**

软件生存周期：一个软件从提出开发要求开始直到该软件报废为止的整个时期。

包括：可行性分析和项目开发计划，需求分析，概要设计，详细设计，编码，测试，维护。

8、衡量模块独立性的两个标准是什么？各表示什么含义？

内聚和耦合。

内聚：又称为块内联系，指模块内部各成分之间相互关联的程度，以高内聚为设计目标。

耦合：也称块间联系，模块之间相互联系程度的度量，联系越紧密，耦合性越强，独立性越差，以低耦合为设计目标。

**9、什么是结构分析方法？结构化分析方法通过哪些步骤来实现？**

结构化分析：简称SA，面向数据流进行数据分析的方法。采用自顶向下逐层分解的分析策略。顶层抽象地描述整个系统，底层具体地画出系统工程的每个细节。中间层则是从抽象到具体的过渡。

实现的步骤：⑴了解当前系统的工作流程，获得当前系统的物理模型。 ⑵抽象出当前系统的逻辑模型。 ⑶建立目标系统的逻辑模型。 ⑷作进一步补充和优化。

10、**什么是需求分析？需求分析阶段的基本任务是什么**？

需求分析：开发人员准确地理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的需求规格说明的过程。

基本任务：

⑴问题识别

⑵分析与综合，导出软件的逻辑模型

⑶编写文档

11、什么样是"事物流"？什么是"变换流"？

数据流图（DFD）一般可分为"变换流"，"事物流"两种。

"变换流"：由输入﹑输出﹑变换（或称处理）三部分组成，是一顺序结构。

"事物流"：它的某个加工，分离成许多发散的数据流，形成许多加工路径，并且根据输入值选择其中一个路径来执行。（这人加工称为事务处理中心）

**12、单元测试有哪些内容?**

单元测试主要针对模块的以下五个基本特征进行测试：

(1)模块接口

(2)局部数据结构

(3)重要的执行路径

(4)错误处理

(5)边界条件

**13、什么是软件生存周期模型？有哪些主要模型？**

生存周期模型：描述软件开发过程中各种活动如何执行的模型。对软件开发提供强有力的支持，为开发过程中的活动提供统一的政策保证，为参与开发的人员提供帮助和指导，是软件生存周期模型化技术的基础，也是建立软件开发环境的核心。

主要有：瀑布模型，增量模型，螺旋模型，喷泉模型，基于知识的模型，变换模型。

14、软件设计的基本原理包括哪些内容？

⑴模块化：模块是软件的组成部分，是具有独立功能且可命名的一段程序，所有模块组成整体，可以满足问题的要求。

⑵抽象：认识复杂现象过程所使用权的工具，只考虑事物本质的共同特性，忽略细节和其它因素。通过抽象确定组成软件的过程实体。

⑶信息隐蔽：将模块实现自身功能的细节与数据"隐蔽"起来。模块间仅交换为完成系统功能所必须的信息。

⑷模块独立性：每个模块只完成系统要求的独立的子功能。

15、结构化程序设计的基本要点是什么?

主要有三个：

1、采用自顶向下、逐步求精的程序设计方法

2、使用三种基本控制结构构造程序。任何程序都可以由顺序、选择、重复(循环)三种基本控制结构构造，这三种基本结构的共同点是单入口、单出口。

3、主程序员组的组织形式。

**16、详细设计有哪几种描述方法？**

详细设计的描述方法有图形、表格和语言，其中图形常用结构化程序流程图、盒图和PAD(问题分析图)为描述工具，语言常用过程设计语言(PDL)来作为工具。

17、单元测试中采用什么方法?

测试的方法是为被测试模块编写驱动模块和桩模块来实现被测试单元的可运行。通过驱动模块来模拟被测试模块的上级调用模块，以上级模块调用被测模块的格式驱动被测模块，接收被测模块的测试结构并输出。桩模块则用来代替被测试模块所调用的模块。它的作用是返回被测模块所需的信息。

18、什么是软件配置管理?什么是基线?

软件配置管理，简称SCM(Software Configuration Management),是指一组管理整个软件生存期各阶段中变更的活动。软件配置管理技术可以使软件变更所产生的错误达到最小并最有效地提高生产率。

基线：是软件生存期中各开发阶段的一个特定点，它的作用是把开发各阶段工作的划分更加明确化，使本来连续的工作在这些点上断开，以便于检查与肯定阶段成果。

19、**简述文档在软件工程中的作用？**

文档在软件工程中的作用如下：

(1)提高软件开发过程的能见度

(2)提高开发效率

(3)作为开发人员阶段工作成果和结束标志

(4)记录开发过程的有关信息便于使用与维护；

(5)提供软件运行、维护和培训有关资料；

(6)便于用户了解软件功能、性能。

20、软件工程三要素之间的关系？

软件质量是软件工程的生命线，软件工程以质量保证为基础。

质量管理促进了过程的改进,创造了许多行之有效的软件开发方法和工具。

软件工程釆用层次化的方法,每个层次都包括过程、方法、工具三要素。

方法支撑过程和工具，过程和工具又促进方法学的研究。

**21、说明面向对象的特征**？

面向对象的特征是：

(1)对象唯一性。每个对象都有自身唯一的标识，在对象生存期中，其标识不变，不同的对象不能有不同的标识。

(2)分类性。是指将具有一致的数据结构(属性)和行为(操作)的对象抽象成类

(3)继承性。是子类自动共享父类数据结构和方法的机制，这是类之间的一种关系。

(4)多态性。是指相同的操作或函数过程可以作用于多种类型的对象上并获得不同的结果。

**22、**简述"变换分析"设计步骤。

⑴确定变换中心，逻辑输入和输出。

⑵设计软件结构的顶层和第一层--变换结构。

⑶设计中﹑下层模块，对第一层的模块自顶向下分解。

⑷设计的优化 ：输入部分求精，每个物理输入设置专门模块，以体现系统的外部接口。其它输入模块与转换数据模块都很简单时，可将它们合并。

23、请简述可行性研究有哪些步骤？

复查系统规模和目标；研究当前的系统；导出新系统的高层逻辑模型；重新定义问题；导出和评价可供选择的解决方案；推荐可行的行动方案；草拟开发计划；书写文档并提交审查。

**24、请简述黑盒测试和白盒测试有何区别，黑盒测试有哪些常用方法？**

黑盒测试和白盒测试的区别：前者基于功能，后者基于结构；

黑盒测试常用方法有：边界值、等价类、因果图、错误推测法等。

25、什么是软件可维护性？常见的软件维护活动有哪几类？

软件可维护性的定义：软件能够被理解、校正、适应及增强功能的容易程度；

常见的软件维护活动有：改正型、适应型、完善型、预防型。

26、**CMM全称是什么?分为哪几个级别？**

CMM全称是软件能力成熟度模型；

从无序到有序的进化分成5个级别，分别为：

1 初始级；2 可重复级；3 已定义级；4 已管理级；5 优化级。

27、**什么是数据字典?简述数据字典与数据流图的关系。**

数据字典是关于数据的信息的集合，对数据流程图中的各个元素做完整的定义与说明，是数据流程图的补充工具；

数据字典与数据流图的关系：数据流图和数据字典共同构成系统的逻辑模型，没有数据字典数据流图就不严格，然而没有数据流图数据字典也难于发挥作用。 数据流图和对数据流图中每个元素的精确定义放在一起，才能共同构成系统的规格说明。

28、**软件生命周期划分为哪几个阶段？**

**软**件生命周期分为三个时期八个阶段：

软件定义：问题定义、可行性研究；

软件开发：需求分析、概要设计、详细设计、编码、测试；

软件运行：软件维护

**29、简述三种面向对象模型的主要功能？**

①对象模型：表示了静态的结构化的系统数据性质，描绘了系统的静态结构，从客观世界实体的对象关系角度来描绘对象。

②动态模型：该模型描述了系统的控制结构，它表示了瞬间的、行为化的系统控制性质，它关心的是系统的控制及操作的执行顺序，它从对象的事件和状态的角度出发，表现了对象的交互行为。

③功能模型：表示变化的系统“功能”性质，它指明系统应该“做什么”，因此功能模型更直接的反映了用户对目标系统的要求。

30、详细设计的基本任务是什么？

详细设计的基本任务包括：为每个模块进行详细的算法设计；为模块内的数据结构进行设计；对数据库进行物理设计；其他设计；编写详细设计说明书；评审。

g

**31、黑盒测试和白盒测试有什么区别？**

（1）黑盒测试法: 把程序看作一个黑盒子，完全不考虑程序的内部结构和处理过程。它只检查程序功能是否能按照规格说明书的规定正常使用，程序是否能适当地接收输入数据并产生正确的输出信息，程序运行过程中能否保持外部信息的完整性。黑盒测试又称为功能测试，包含等价类划分法、边界值法、错误推测法、因果图法等。

（2）白盒测试法: 是把程序看成装在一个透明的白盒子里，测试者完全知道程序的结构和处理算法。这种方法按照程序内部的逻辑测试程序，检测程序中的主要执行通路是否都能按预定要求正确工作。白盒测试又称为结构测试，包含逻辑覆盖法和基本路径测试法。

**32、什么是软件工程？软件工程面临的问题是什么？**

软件工程就是用科学的知识和技术原理来定义，开发，维护软件的一门学科。

软件工程面临的问题：软件的费用，可靠性，可维护性，软件生产率和软件的重用。

33、软件工程三要素之间的关系？

软件质量是软件工程的生命线，软件工程以质量保证为基础。

质量管理促进了过程的改进,创造了许多行之有效的软件开发方法和工具。

软件工程釆用层次化的方法,每个层次都包括过程、方法、工具三要素。

方法支撑过程和工具，过程和工具又促进方法学的研究。

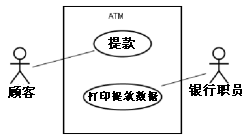
34、简述结构化程序设计方法的基本要点。

(1)采用自顶向下，逐步求精的程序设计方法。

(2)使用三种基本控制结构构造程序，分别是顺序，选择和循环

(3)采用主程序员组的组织形式。

35、顾客在自动柜员机中插入银行卡，执行提款的服务项目；银行职员要求自动柜员机打印所有的提款数据。画出自动柜员机的Use Case图。？



36、简述提高可维护性的方法。

(1)建立明确的软件质量目标；

(2)利用先进的软件开发技术和工具；

(3)建立明确的质量保证工作；

(4)选择可维护的程序设计语言；

(5)改进程序文档。

**37、简述文档在软件工程中的作用。**

(1)提高软件开发过程的能见度；

(2)提高开发效率；

(3)作为开发人员阶段工作成果和结束标志；

(4)记录开发过程的有关信息便于使用与维护；

(5)提供软件运行、维护和培训有关资料；

(6)便于用户了解软件功能、性能。

38、什么是软件危机？产生原因是什么？

软件开发技术的进步未能满足发展的要求。在软件开发中遇到的问题找不到解决的办法，问题积累起来，形态尖锐的矛盾，导致了软件危机。

产生原因：

⑴ 软件规模越来越大，结构越来越复杂。

⑵ 软件开发管理困难而复杂。

⑶ 软件包开发费用不断增加。

⑷ 软件开发技术落后。

⑸ 生产方式落后，仍采用手工方式。

⑹ 开发工具落后，生产率提高缓慢。

39、调试的目的是什么?调试有哪些技术手段?

调试则是在进行了成功的测试之后才开始的工作。调试的目的是确定错误的原因和位置，并改正错误，因此调试也称为纠错(Debug)。调试的技术手段有简单的调试方法、归纳法、演绎法和回溯法等。

**40、**在传统软件工程方法指导下，数据流图和数据字典的关系如何？

数据字典是关于数据的信息的集合，对数据流程图中的各个元素做完整的定义与说明，是数据流程图的补充工具。

数据流图和数据字典共同构成系统的逻辑模型，没有数据字典数据流图就不严格，然而没有数据流图数据字典也难于发挥作用。 数据流图和对数据流图中每个元素的精确定义放在一起，才能共同构成系统的规格说明。

## 四、应用题(每小题10分\*3=30)

1．某培训中心要研制一个计算机管理系统。它的业务是：

将学员发来的信件收集分类后，按几种不同的情况处理。

（1）如果是报名的，则将报名数据送给负责报名事务的职员，他们将查阅课程文件，检查该课程是否额满，然后在学生文件、课程文件上登记，并开出报告单交财务部门，财务人员开出发票给学生。

（2）如果是想注销原来已选修的课程，则由注销人员在课程文件、学生文件和帐目文件上做相应的修改，并给学生注销单。

（3）如果是付款的，则由财务人员在帐目文件上登记，也给学生一张收费收据。

要求：

（1）对以上问题画出数据流程图。

（2）画出该培训管理的软件结构图。

解答：

(1)数据流图

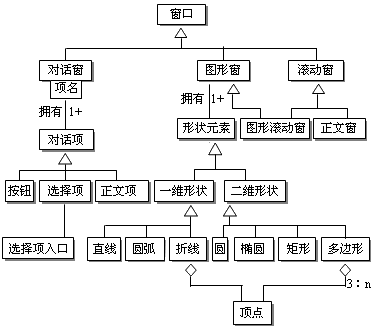


(2)软件体系结构

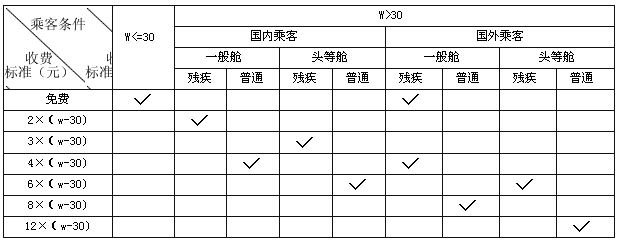


2．建立窗口系统的对象模型，问题陈述如下：“窗口分为对话窗、图形窗、滚动窗三种。对话窗中有若干对话项，由唯一的项名字来确定；对话项分为按钮、选择项和正文项三种，选择项中有若干选择项入口。图形窗中有若干形状元素，形状元素分为一维形状和二维形状。一维形状又分为直线、圆弧、折线；二维形状分为圆、椭圆、矩形、多边形，其中折线与多边形由若干有序顶点组成。正文窗是滚动窗的一种，而图形滚动窗既是图形窗又是滚动窗。”

解答：（泛化关系）



3．假设某航空公司规定，乘客可以免费托运行李的重量不超过30公斤。当行李的重量超出30公斤时 ，对一般舱的国内乘客超重部分每公斤收费4元，对头等舱的国内乘客超重部分每公斤收费6元。对国外乘客 超重部分每公斤收费比国内乘客多一倍，对残疾乘客超重部分每公斤收费比正常乘客少一半。试画出相应判定表。



4．下图描述了某个子程序的处理流程，根据判定覆盖、条件覆盖、判定/条件覆盖、条件组合覆盖、路径覆盖等五种覆盖标准，从供选择的答案中分别找出满足相应标准的最小的测试数据组。

(1)a=5b=1(2)a=5b=-1

(3)a=5b=1(4)a=5b=1

a=-5b=-1a=0b=-1

(5)a=5b=-1(6)a=5b=1

a=-5b=1a=0b=0

a=-5b=-1a=-5b=-1

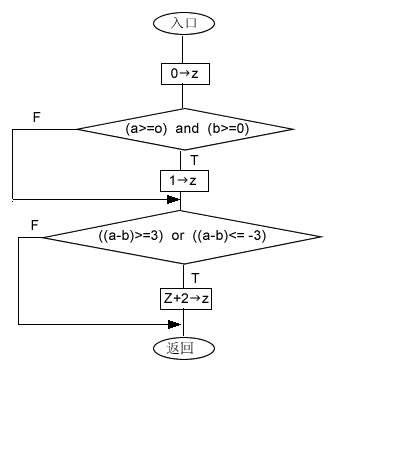
(7)a=5b=1(8)a=5b=1

a=0b=1a=0b=-1

a=0b=-1a=-5b=1

a=-5b=1a=-5b=-1

流程图如下：



解答：

达到判定覆盖为（4）；

达到条件覆盖为（3）；

达到判定/条件覆盖为（6）；

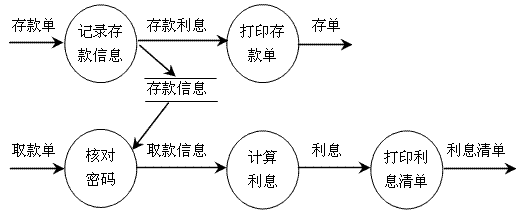
达到条件组合覆盖为（8）；

达到路径覆盖为（7）。

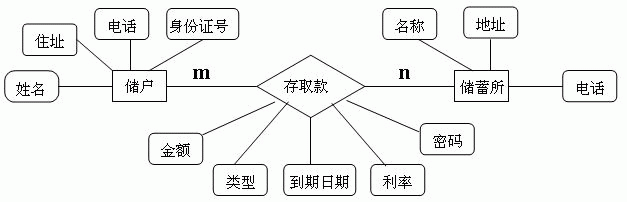
7．某银行计算机储蓄系统的工作流程大致如下：储户填写的存款单或取款单由业务员键入系统，如果是存款则系统记录存款人的姓名、住址（或电话号码）、身份证号码、存款类型、存款日期、到期日期、利率及密码（可选）等信息，并印出存款单给储户；如果是取款而且存款时留有密码，则系统首先核对储户密码，若密码正确或存款时未留密码，则系统计算利息并印出利息清单给储户。 请用数据流图描绘本系统的功能，并画出系统的E-R图。

解答：

DFD图：



ER图：



8．请对下列子程序进行测试：



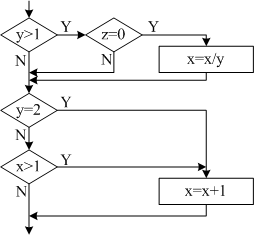
该子程序接受x, y, z的值，并将计算结果x的值返回给调用程序。

要求：

(1)画出程序流程图。

(2)依据条件组合覆盖标准设计测试用例。

（1）程序流程图



（2）条件组合覆盖设计测试用例

① y=2, z=0, x=4;

② y=2, z=1, x=1;

③ y=1, z=0, x=2;

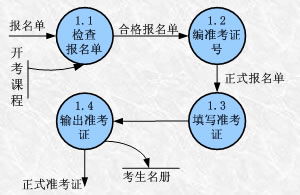
④ y=1, z=1, x=1.

9．自学考试报名过程中有个“记录报名单”的加工。该加工主要是根据报名表(姓名、 性别、身份证号、课程名)和开考课程(课程名、开考时间)、经校核，编号、填写、输出准考证给报 名者， 同时记录到考生名册中(准考证号、姓名、课程)。

请绘制该加工的DFD图，并写出数据词典中的数据流条目。

解答：

(1)DFD图：



(2)数据流词典

报名单 = 姓名 + 性别 + 身份证号 + 课程名

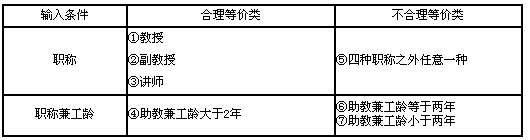
开考课程 = 课程名 + 开考时间

考生名册 = 准考证号 + 姓名 + 课程

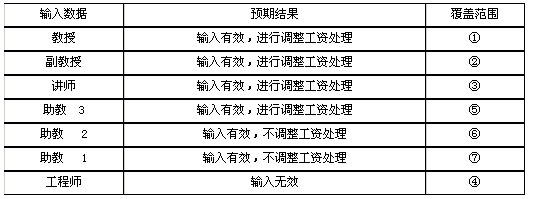
12．某“调整工资”处理模块接受一个“职称”的变量，根据职称的不同（助教，讲师，副教授，教授）作不同的处理，其中若是助教还必须输入工龄，只有工龄超过两年才能调整工资。请用等价类划分法设计测试用例。

解答；

划分等价类：



计测试用例：



17．某校制定了教师的讲课课时津贴标准。对于各种性质的讲座，无论教师是什么职称，每课时津贴一律是50元；而对于一般的授课，则根据教师的职称来决定每课时津贴费：教授30元，副教授25元，讲师20元，助教15元。请用判定表描述上述问题。

解答：

判定表：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
| 教授 |  | T | F | F | F |
| 副教授 |  | F | T | F | F |
| 讲师 |  | F | F | T | F |
| 助教 |  | F | F | F | T |
| 讲座 | T | F | F | F | F |
| 50 | ╳ |  |  |  |  |
| 30 |  | ╳ |  |  |  |
| 25 |  |  | ╳ |  |  |
| 20 |  |  |  | ╳ |  |
| 15 |  |  |  |  | ╳ |

**软件工程分析题**

**五、分析题(每小题15分\*1=15)**

**根据提供的论题，写出500字左右的分析报告。**

**1．（软件工程的思想，流程，作用，每一个级别的特征）**

**4．（写下每一个内聚的特征）**

****

**12．**

****

**软件工程试题一**

一．单项选择题

1. 以下不是软件危机的典型表现的是： ( D ) 。

A. 对软件开发成本和进度的估计常常很不准确

B. 用户对“已完成的”软件系统不满意的现象经常发生

C. 软件产品的质量往往靠不住

D. 采用面向对象方法开发软件

3. 下列哪个不是软件工程方法学中的要素( C )。

A. 方法 B. 工具 C. 程序 D. 过程

4. 下列模块的独立性的耦合方式中，最好的是( C )

A.公共耦合 B. 控制耦合 C. 数据耦合 D. 内容耦合

5．概要设计中，开发人员要解决的最重要的问题是 ( D ) 。

A. 要让软件做什么 B. 要给该软件提供哪些信息

C. 要求软件工作效率怎样 D. 要让该软件模块具有何种结构

6. 为了提高模块的独立性，模块内部最好是 ( C ) 。

A. 逻辑内聚 B. 时间内聚 C. 功能内聚 D. 通信内聚

7. 程序的三种基本控制结构的共同特点是 ( C ) 。

A. 不能嵌套使用 B. 只能用来写简单程序

C. 只能一个入口和一个出口 D. 已经用硬件实现

8. 以下说法错误的是 ( ) 。 A

A. 面向对象分析与面向对象设计的定义没有明显区别

B. 从面向对象分析到面向对象设计，是一个逐渐扩充模型的过程

C. 在实际的软件开发过程中，面向对象分析与面向对象设计的界限是模糊的

D. 面向对象分析和面向对象设计活动是一个多次反复迭代的过程

9. 软件测试过程中的确认测试主要是为了发现 ( A ) 阶段的错误。

A. 需求分析 B. 概要设计 C. 详细设计 D. 编码

10. 在面向对象的高层设计体系结构中， ( C ) 是用来描述用户一般使用软件习惯的部分

A. 人机交互界面 B. 问题论域 C. 系统交互界面 D. 数据管理

11. 为适应软硬件环境变化而修改软件的过程是 ( B ) 。

A. 校正性维护 B. 适应性维护

C. 完善性维护 D. 预防性维护

12. 在UML的各类图中，（ A ）图是描述软件功能的。

A 用例图 B 类图 C 活动图 D 顺序图

13. （ D ）不属于OO三要素。

A.对象 B. 继承 C.封装 D.多态

14．（C ）在ATM机系统中，下列哪个不能在面向对象设计中被定义为类。 C

A 中央计算机 B 柜台事务 C 拨号 D 营业厅

15．在黑盒测试中，作为相关黑盒测试有效补充的测试方法是 ( B ) 。

A. 等价类划分法 B. 边界值分析法 C. 错误推测法 D. 因果图法

二．判断题(本大题共\_\_15\_\_题，每题\_\_\_1\_\_\_分，共\_\_\_15\_\_\_\_\_分。

1. 软件简单的说就是程序和相关的数据及文档。 T
2. 测试用例由一系列输入和实际的输出组成。 F
3. 软件工程的集成测试都是自顶向下的。 F
4. 编码是尽可能采用局部变量。 T
5. 在进行需求分析时需同时考虑维护性问题。 T
6. 顺序图描述了对象之间动态的交互关系，着重体现对象间消息传递的时间顺序。 T
7. 在进行概要设计时应加强模块间的联系。 F
8. 模块越小，模块化的优点越明显。一般来说，模块的大小都在10行以下。 F
9. 模块间的联系越紧密越好。 F
10. 用黑盒法测试时，测试用例是根据程序内部逻辑设计的。 F
11. 维护就是在软件交付使用后进行的修改。 T
12. 一组测试用例是判定覆盖，则一定是语句覆盖。 T
13. 面向对象软件工程方法就是至上而下、逐步求精的方法。 F
14. 向一个已经延期的项目增加人手，往往不会加快其开发进程。 T
15. 应尽可能多用GOTO语句。 F

三．简答题

1.简述螺旋模型。

螺旋模型（Spiral Model）采用一种周期性的方法来进行系统开发。这会导致开发出众多的中间版本。使用它，项目经理在早期就能够为客户实证某些概念。该模型是快速原型法，以进化的开发方式为中心，在每个项目阶段使用瀑布模型法。这种模型的每一个周期都包括需求定义、风险分析、工程实现和评审4个阶段，由这4个阶段进行迭代。软件开发过程每迭代一次，软件开发又前进一个层次

2.简述你对模块独立性的理解。

模块独立性指每个模块只完成系统要求的独立的子功能,并且与其他模块的联系最少且接口简单，模块内部各部分及模块间的关系的一种衡量标准，由内聚和耦合来度量。

模块独立性的重要性：具有独立的模块的软件比较容易开发出来。这是由于能够分割功能而且接口可以简化；独立的模块比较容易测试和维护。这是因为相对说来，修改设计和程序需要的工作量比较小，错误传播范围小，需要扩充功能时能够"插入"模块。总之，模块独立是优秀设计的关键，而设计又是决定软件质量的关键环节。

四．综合题(本大题共\_\_3\_\_题，每题\_\_\_10\_\_\_分，共\_\_\_30\_\_\_\_\_分。)

1、 问题陈述：为更好的管理食堂工作，某公司拟开发食堂管理系统，食堂管理员将所有菜式信息输入到系统中，并提前录入每日菜谱信息到系统中，员工通过系统进行网上订餐，从而得到本人的相关订餐信息，

员工可以通过系统了解每日菜谱信息和自己的消费情况，每月，食堂管理员得到所有员工的消费情况和每个菜式的订餐统计信息（10分）

1. 以结构化方法建立该问题的数据流图(至少2层，即顶层和0层) （7分）
2. 描述数据流图中的主要数据字典（3分）

**软件工程试题二**

一、填空题（每空1分，共25分）

1. 软件生存周期一般可以划分为，**问题定义**、\_\_**可行性研究**\_\_、\_\_**需求分析**\_\_\_、**设计** 、**编码** 、**测试** 和\_ \_**运行与维护**\_。
2. 基于软件的功能划分，软件可以划分成\_\_\_**系统软件**\_\_、\_\_**支撑软件**\_\_、\_\_\_**应用软件**\_\_\_三种。

3、可行性研究，应从  **经济可行性**  、**技术可行性、运行可行性**、 **法律可行性** 和**开放方案可行性** 等方面研究。

4、数据流图的基本四种成分：**数据输入的源点和数据输出汇点**、**加工**、**数据流** 和**数据存储文件**

5、结构化分析方法是面向\_\_\_\_\_\_**数据流**\_\_进行需求分析的方法。结构化分析方法使用\_\_\_\_**数据流图DFD\_**\_\_与\_\_\_\_\_**数据字典DD\_**\_\_来描述。

6、为了在软件开发过程中保证软件的质量，主要采取下述措施：**审查**\_、**复查和管理复审、测试**

7、大型软件测试包括 **单元测试** 、 **集成测试**  、**确认测试**和 **系统测试**  四个步骤。

二、单项选择题（每小题2分，共20分）

1. 软件设计中划分模块的一个准则是（ C ）。
2. 低内聚低耦合 B、低内聚高耦合
   * + 1. 高内聚低耦合 D、内聚高耦合
   1. Jackson设计方法是由英国的M. Jackson提出的，它是一种面向（ C ）的软件设计方法。
3. 对象 B 数据流 C 数据结构 D 控制结构
   1. 试判断下列叙述中，哪 个（些）是正确的(  C   )。

a、软件系统中所有的信息流都可以认为是事务流

b、软件系统中所有的信息流都可以认为是变换流

c、事务分析和变换分析的设计步骤是基本相似的

A、a  B、b   C、c D、b和c

* 1. （ A ）是用户和设计交换最频繁的方法。

1. 原型化方法 B、瀑布模型方法

C、 螺旋模型方法 D、构件组装模型

* 1. 软件工程方法学的目的是：使软件生产规范化和工程化，而软件工程方法得以实施的主要保证是(  C )。

1. 硬件环境 B、软件开发的环境

C软件开发工具和软件开发的环境   D、 开发人员的素质

* 1. 软件测试的目的是？(  D )
     1. 证明软件的正确性
     2. 找出软件系统中存在的所有错误
     3. 证明软件系统中存在错误
     4. 尽可能多的发现软件系统中的错误
  2. 使用白盒测试方法时，确定测试数据应根据（ A ）和指定的覆盖标准。

A、程序的内部逻辑 B、程序的复杂程度

C、该软件的编辑人员 D、程序的功能

* 1. 软件维护工作的最主要部分是(  C )。
     1. 校正性维护   B、适应性维护 C、完善性维护    D、预防性维护
  2. PDL是（ B ）语言。
     1. 高级程序设计语言 B、伪码式

C、中级程序设计语言 D、低级程序设计语言

三、判断题（正确的在括号内打上“√”，错误的打上“×”。每题2分，共20分）

1. 用黑盒法测试时，测试用例是根据程序内部逻辑设计的。（ × ）
2. 发现错误多的程序模块，残留在模块中的错误也多。（ √ ）
3. 为了加快软件维护作业的进度，应尽可能增加维护人员的数目。（ × ）
4. 软件维护就是改正软件中的错误。（ × ）
5. 质量保证是为了保证产品和服务充分满足消费者要求的质量而进行的有计划,有组织的活动。（ √ ）
6. 软件开发的主要任务是写程序。 （ × ）
7. 测试只能证明程序有错误,不能证明程序没有错误。（ √ ）
8. 模块化程序设计中，模块越小，模块化的优点越明显。一般来说，模块的大小都在10行以下。（ × ）
9. 在编制程序时，首先应该对程序的结构充分考虑，不要急于开始编码，而要像写软件文档那样，很好地琢磨程序具有什么样的功能，这些功能如何安排，等等。（ × ）
10. 程序设计风格指导原则提出，尽量多使用临时变量。（ × ）

四、简答题（每题5分，共20分）

* 1. 什么是软件工程？

软件工程是指导计算机软件开发和维护的工程学科。采用工程的概念、原理、技术和方法来开发与维护软件，把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来。

* 1. 耦合性和内聚性有几种类型? 其耦合度、内聚强度的顺序如何？

答：低：非直接耦合→ 数据耦合→标记耦合→ 控制耦合→外部耦合→ 公共耦合→内容耦合 ：高

强：功能内聚→ 信息内聚→ 通信内聚→ 过程内聚→ 时间内聚→ 逻辑内聚→ 巧合内聚：弱

* 1. 简述需求分析工作可以分成哪四个方面？软件需求分析的有哪三个基本原则？

答：需求分析阶段分成四个方面：对问题的识别、分析与综合、制定规格说明和评审。三个基本原则：必须能够表达和理解问题的数据域和功能域；必须按自顶向下、逐步分解的方式对问题进行分解和不断细化；要给出系统的逻辑视图和物理视图。

4、什么是黑盒测试法？

答：黑盒测试法把程序看成一个黑盒子，完全不考虑程序的内部结构和处理过程，它只检查程序功能是否能按照规格说明书的规定正常使用，程序是否能适当地接收输入数据，产生正确地输出信息。

五、程序设计与算法描述题（共15分）

* 1. 有下列伪码程序:

START

INPUT (M,N)

IF M>=10

THEN X=10

ELSE X=l

ENDIF

IF N>=20

THEN Y=20

ELSE Y=2

ENDIF

PRINT(X,Y)

STOP

设计该程序的语句覆盖和路径覆盖测试用例.

语句覆盖为：

1. M=9(或<l0),N=l9(或<20)；
2. M=l0(或>=10),N=20(或>=20)

路径覆盖的测试用例为

1. M=9,N=19；
2. M=9、N=20；
3. M=l0,N=l9；
4. M=10,N=20;

**软件工程试题三**

## 一、选择题：（每题1分，共20分）（将答案写在题号前的（）中）

（ C ）1.    软件是（ ）。

A. 处理对象和处理规则的描述 B. 程序

C. 程序及其文档 D. 计算机系统

（ B ）2. 软件需求规格说明的内容不应包括（ ）。

A. 主要功能 B. 算法的详细描述

C. 用户界面及运行环境 D. 软件的性能

（ B ）3. 程序的三种基本控制结构是（ ）。

A. 过程、子程序和分程序 B. 顺序、选择和重复

C. 递归、迭代和回溯 D. 调用、返回和转移

( Ｄ ) 4. 面向对象的分析方法主要是建立三类模型，即( )。  
     A) 系统模型、ER模型、应用模型 B) 对象模型、动态模型、应用模型  
      C) Ｅ-Ｒ模型、对象模型、功能模型 D) 对象模型、动态模型、功能模型  
 ( C ) 5. 在E-R模型中，包含以下基本成分( )。  
      A) 数据、对象、实体 B) 控制、联系、对象  
     C) 实体、联系、属性 D) 实体、属性、操作  
 ( A )  6. 各种软件维护的类型中最重要的是( )。  
     A) 完善性维护 B) 纠错性维护 C) 适应性维护 D) 预防性维护  
( B ) 7．软件测试的目标是（ ）。

A. 证明软件是正确的 B. 发现错误、降低错误带来的风险

C. 排除软件中所有的错误 D. 与软件调试相同

（ D ）8．软件生命周期中所花费用最多的阶段是（ ）

A．详细设计 B．软件编码 C．软件测试 D．软件维护

（ C ）9．若有一个计算类型的程序，它的输入量只有一个X，其范围是[-1.0, 1.0]，现从输入的角度考虑一组测试用例：-1.001, -1.0, 1.0, 1.001.设计这组测试用例的方法是（ ）

A．条件覆盖法 B．等价分类法 C．边界值分析法 D．错误推测法

（ D ）10、详细设计的基本任务是确定每个模块的( )设计

A．功能 B.调用关系 C.输入输出数据 D.算法

（ A ）11．设函数C（X）定义问题X的复杂程序，函数E（X）确定解决问题X需要的工作量（时间）。对于两个问题P1和P2，如果C（P1）>C（P2）显然E（P1）>E（P2）,则得出结论E（P1+P2）>E（P1）+E（P2）就是：（ ）

A．模块化的根据 B．逐步求精的根据

C．抽象的根据 D．信息隐藏和局部化的根据

（ D ）12．下面几种白箱测试技术，哪种是最强的覆盖准则 （ ）

A．语句覆盖 B．条件覆盖 C．判定覆盖 D．条件组合覆盖

（ A ）13．面向数据流的设计方法把（ ）映射成软件结构。

A．数据流 B．系统结构 C．控制结构 D．信息流

（ A ）14.内聚程度最低的是( )内聚

A.偶然 B.过程 C.顺序 D.时间

（ A ）15.确定测试计划是在( )阶段制定的.

A．总体设计 B.详细设计 C.编码 D.测试

（ D ）16．需求分析的产品是（ ）

A．数据流程图案 B．数据字典 C．判定表 D．需求规格说明书

（ C ）17．数据字典是软件需求分析阶段的最重要工具之一，其最基本的功能是（ ）

A．数据库设计 B．数据通信 C．数据定义 D．数据维护

( D )18.( )引入了“风险驱动”的思想，适用于大规模的内部开发项目。

A．增量模型 B．喷泉模型 C．原型模型 D．螺旋模型

（ D ）19．模块的内聚性最高的是（ ）

A．逻辑内聚 B．时间内聚 C．偶然内聚 D．功能内聚

( D )20.提高测试的有效性非常重要,成功的测试是指( )

A.证明了被测试程序正确无误 B. 说明了被测试程序符合相应的要求

C.未发现被测程序的错误 D.发现了至今为止尚未发现的错误

## 二．判断题（每题1分，共10分）将答案写在题号前的（ ）中，正确用√， 错误用χ。

（ × ）1、开发软件就是编写程序。

（ ×　）２、系统测试的主要方法是白盒法，主要进行功能测试、性能测试、安全性测试及可靠性等 测试。

（ × ）3、编程序时应尽可能利用硬件特点以提高程序效率.

（ × ）4、软件需求分析的任务是建立软件模块结构图。

（ √ ）5、尽可能使用高级语言编写程序

（ × ）6、以结构化分析方法建立的系统模型就是数据流图。

（ × ）7、进行总体设计时加强模块间的联系。

（ × ）8、编码时尽量多用全局变量.

（ √ ）9、用CASE环境或程序自动生成工具来自动生成一部分程序.

（ × ）10、软件测试是要发现软件中的所有错误。

## 三、简答题：（每题5分，共25分）

1. 软件生命期各阶段的任务是什么？ 软件定义、软件开发、软件运行  
     答：软件生命期分为7个阶段：  
     1、问题定义：要解决的问题是什么

2、可行性研究：确定问题是否值得解，技术可行性、经济可行性、操作可行性

3、需求分析：系统必须做什么

4、总体设计：系统如何实现，包括系统设计和结构设计

5、详细设计：具体实现设计的系统

6、实现：编码和测试

7、运行维护：保证软件正常运行。

2、软件重用的效益是什么？  
     答：1、软件重用可以显著地改善软件的质量和可靠性。

2、软件重用可以极大地提高软件开发的效率。

3、节省软件开发的成本，避免不必要的重复劳动和人力、财力的浪费。

 3、 自顶而下渐增测试与自底而上渐增测试各有何优、缺点？  
     答：①　自顶而下渐增测试

     优点：不需要测试驱动程序，能够在测试阶段的早期实现并验证系统的主要功能，而且能够尽早发现上层模块的接口错误。

     缺点：需要存根程序，底层错误发现较晚。

     ②　自底而上渐增测试

     优点与缺点和自顶而下渐增测试相反。

   4 、 提高可维护性的方法有哪些？  
     答：（1）在软件工程的每一阶段都应该努力提高系统的可维护性，在每个阶段结束

前的审查和复审中，应着重对可维护性进行复审。（2）在需求分析阶段的复审中，应对将来要扩充和修改的部分加以注明。（3）在讨论软件可移植性问题时，要考虑可能要影响软件维护的系统界面。  
（4）在软件设计的复审中，因从便于修改、模块化和功能独立的目标出发，评价软件的结构和过程，还应对将来可能修改的部分预先做准备。（5）在软件代码复审中，应强调编码风格和内部说明这两个影响可维护性的因素。（6） 在软件系统交付使用前的每一测试步骤中都应给出需要进行预防性维护部分的提示。（7）在完成每项维护工作后，都应对软件维护本身进行仔细认真的复审。  （8） 为了从根本上提高软件系统的可维护性，人们正试图通过直接维护软件规格说明来维护软件 ，同时也在大力发展软件重用技术。

5、简述软件测试要经过哪几个步骤，每个步骤与什么文档有关。

【解答】

测试过程按 4 个步骤进行，即单元测试、集成测试、确认测试和系统测试。

单元测试集中对用源代码实现的每一个程序单元进行测试，与其相关的文档是单元测试计划和详细设计说明书。

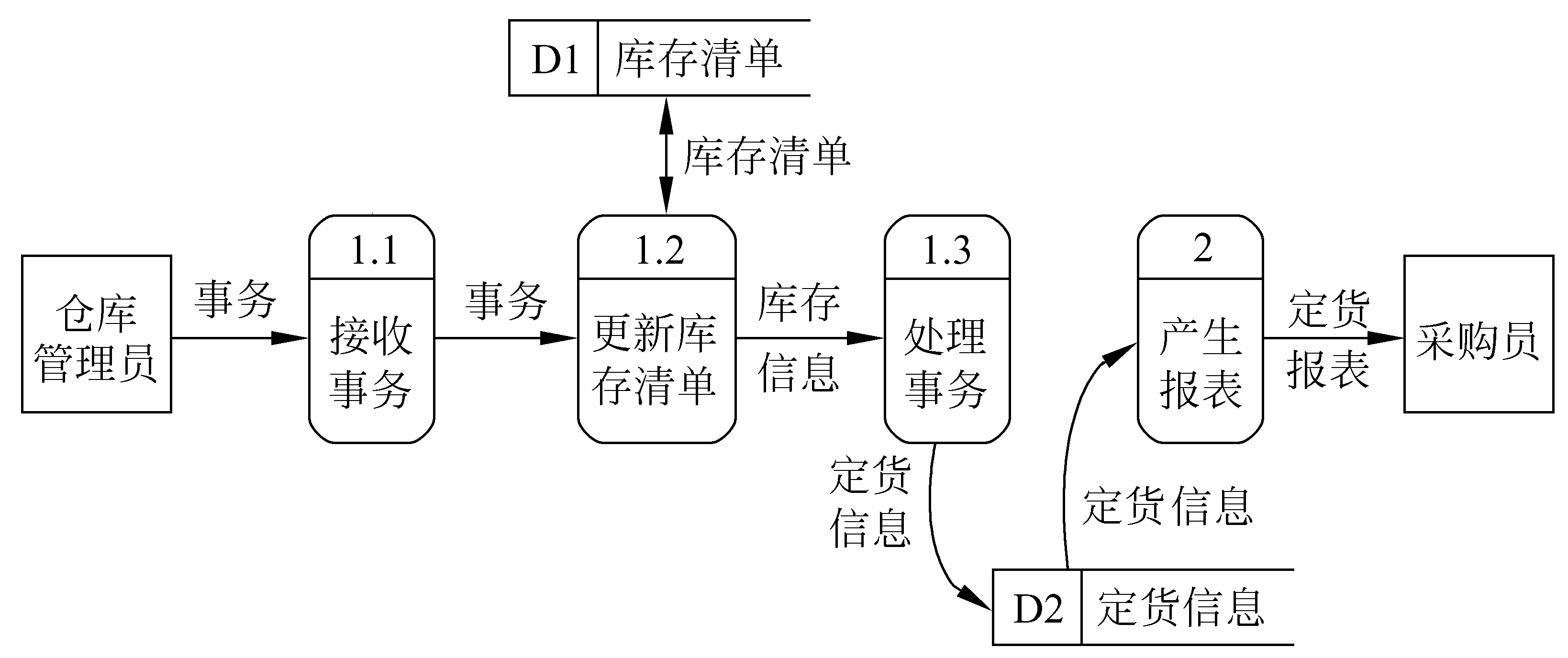
集成测试把已测试过的模块组装起来，主要对与设计相关的软件体系结构的构造进行测试。与其相关的文档是集成测试计划和软件需求说明书。

确认测试则是要检查已实现的软件是否满足了需求规格说明中确定了的各种需求，以及软件配置是否完全、正确。与其相关的文档是确认测试计划和软件需求说明书。

系统测试把已经经过确认的软件纳入实际运行环境中，与其他系统成份组合在一起进行测试。与其相关的文档：用户指南、使用手册等。

## 四、应用题（每题8分，共40分）

1. 假设一家工厂的采购部每天需要一张定货报表，报表按零件编号排序，表中列出所有需要再次定货的零件。对于每个需要再次定货的零件应该列出下述数据：零件编号，零件名称，定货数量，目前价格，主要供应者，次要供应者。零件入库或出库称为事务，通过放在仓库中的CRT终端把事务报告给定货系统。当某种零件的库存数量少于库存量临界值时就应该再次定货。要求：画出系统的数据流图。



答：

软件工程试题四

## 一、选择题

1．软件是一种 产品。

A．物质 B. 逻辑√ C. 工具 D. 文档

2．软件产品的生产主要 \_\_\_ \_ \_。

A. 研制√ B. 再生产 C. 开发 D. 生产

3．软件工程是一门\_\_ \_学科。

A. 工程性 B. 原理性 C. 综合性√ D. 理论性

4．类是面向对象方法中的一个重要概念，类的描述分为：名称、 和操作。

A. 角色 B. 用例 C. 方法 D. 属性√

5．下列描述中正确的是\_\_\_ \_\_。

A. 软件工程的主要是解决软件开发中的技术问题

B. 软件工程主要解决软件产品的生产率问题

C. 软件工程的主要思想是强调在软件开发过程中需要运用工程化的原则√

D. 软件工程只是解决软件项目的问题.

6．需求分析最终结果是产生 。

A．项目开发计划 B．可行性分析报告 C．设计说明书 D．需求分析说明书√

7．需求分析中，开发人员要从用户那里解决最重要的问题是 。

A．要让软件做什么√ B．要给软件提供哪些信息

C．要求软件工作效率怎样 D．要让该软件具有何种结构

8．分层DFD是一种比较严格又易于理解的描述方式，它的顶层图描述了系统的 。

A．软件的作者 B．输入与输出√ C．细节 D．绘制的时间

9．数据字典中，一般不包括 条目。

A．源点与终点√ B．数据存储 C．数据流 D．加工

10．软件需求分析是保证软件质量的重要步骤，它的实施应该是在 。

A．编码阶段 B．软件设计阶段√ C．软件定义过程 D．软件开发全过程

11．软件的主要功能和结构是在 阶段确定的。

A．测试 B．编程 C．分析设计√ D．维护

12．需求分析最终结果是产生 。

A．需求分析说明书√ B．可行性分析报告C．项目开发计划 D．设计说明书

13．可行性研究主要从 方面进行研究。

A．技术可行性，经济可行性，操作可行性B．经济可行性，系统可行性，时间可行性

C．经济可行性，系统可行性，操作可行性D．技术可行性，经济可行性，社会可行性 √

14． 可行性分析研究的目的是 。

A．项目值得开发与否√ B．开发项目 C．争取项目 D．规划项目

15． 需求分析中开发人员要从用户那里了解 。

A．输入的信息 B．用户使用界面 C．软件做什么√ D．软件的规模

16.需求分析阶段研究的对象是 。

A．软硬件要求 B．系统要求 C．分析员要求 D．用户需求√

17．系统流程图用于可行性分析中的 的描述。

A．目标系统 B．当前逻辑模型 C．当前运行系统√ D．新系统

18．数据字典是用来定义 中的各个成分的具体含义的。

A．数据流图√ B．功能结构图 C．系统结构图 D．流程图

19．需求分析是 。

A．软件生命周期的开始 B．由系统分析人员完成

C．由开发人员和系统分析人员完成 D．软件开发任务的基础性工作√

20．数据字典是对数据定义信息的集合，它所定义的对象都包含于 。

A．软件结构 B．程序框图 C．数据流图√ D．方框图

21．结构化设计方法在软件开发中，用于 。

A． 测试用例设计 B．程序设计 C．概要设计√ D．详细设计

22. 在进行软件结构设计时应遵循的最主要的原理是 。

A．模块独立√ B．抽象 C．模块化 D．信息隐藏

23.通信内聚是指 。

A．把需要同时执行的动作组合在一起形成的模块

B．模块内所有元素共同完成一个功能，缺一不可

C．一个模块中各个处理元素都密切相关于同一功能且必须顺序执行

D．各处理使用相同的输入数据或者产生相同的输出数据√

24.软件的结构化设计方法中，一般分为概要设计和详细设计两阶段，其中详细设计主要是要建立 。

A．软件模块√ B．软件过程 C．软件模型 D．软件结构

25.软件详细设计阶段的任务是 。

A．功能设计 B．算法设计√ C．调用关系设计 D．输入/输出设计

## 二、填空题

1．软件工程是一门 综合性交叉 学科，计算机学着重于 理论和科学 ，

软件工程着重于 设计实现软件系统 。

2．需求分析阶段产生的最重要的文档之一是 需求规格说明书 。

3．SA方法中的主要描述工具是 DFD图 与 DD 字典 。

4．数据字典中有4类条目，分别是 数据流 ， 数据项 ， 数据存储 ， 加工 。

5．项目可行性分析分为 经济可行性 ， 技术可行性 ， 社会可性性 。

6．软件概要设计阶段产生的重要的文档之一是 概要设计说明书 。

7．软件结构是以 模块 为基础而组成的，一种控制层次结构。

8．反映软件结构的基本形态特征是 深度 ， 宽度 和 扇入 ， 扇出 。

9.数据流图一般可分为 变换 型和 事务 型两类。

10.程序流程图（PFC）又称为 程序框图 ，它是软件开发者最熟悉的一种 算法 表达工具。

11.在 UML 用例图中，椭圆表示 用例 ，方框表示 系统边界 ，小人状图案表示 执行者 。

## 三、概念题

1．软件生存周期：指软件从提出开发要求开始，直到该软件报废为止的整个时期。把整个生存周期划分为若干阶段，使得每个阶段有明确的任务，使规模大、结构和管理复杂的软件开发变得容易控制和管理。（P7）

2．数据字典（DD）：数据字典是用来定义数据流图中的各个成分的具体含义的。它以一种准确的、无二义性的说明方式为系统的分析、设计及维护提供了有关元素的一致的定义和详细的描述。（P26）

3．DFD图： Data Flow Diagram 数据流图，是SA方法中用于表示系统逻辑模型的一种工具，它以图形的方式描绘数据在系统中流动和处理的过程，由于它只反映系统必须完成的逻辑功能，所以它是一种功能模型。（P22）

4．用例模型：用例模型主要是由用例、用例描述和用例图组成，用来描述系统的外部特征。它表示了从系统的外部用户（即执行者或角色）的观点看系统应该具备什么功能。它说明系统实现设什么功能，一幅用例图包含的模型元素：系统、执行者、用例与用列之间关系。

5. Alpha测试Alpha测试是由一个用户在开发环境下进行的测试，也可以是公司内部的用户在模拟实际操作环境下进行的受控测试，Alpha测试不能由程序员或测试员完成。

Alpha测试的目的是评价软件产品的功能、可使用性、可靠性、性能和支持。测试可以从软件产品编码结束之后开始，或在模块（子系统）测试完成后开始，也可以在确认测试过程中产品达到一定的稳定和可靠程度之后再开始。

## 四、简答题

1．软件危机主要表现在那些方面？解决软件危机的途径有那些？

答：软件危机主要表现在： 1）软件开发无计划性 2）软件需求不充分

3）软件开发过程无规范 4）软件产品无测评手段

解决软件危机的途径有：

1） 软件开发是智力劳动；

2） 是一种组织良好、管理严密、各类人员协同配合、共同完成的工程项目。

3） 充分吸取几十年来人类从事计算机硬件研究和开发的经验教训。

4） 使用适当的软件工具，支持软件开发的过程

总之，消除软件危机，既要有技术措施(方法和工具)，又要有必要的组织管理措施。软件工程正是从管理和技术两方面研究如何更好地开发和维护计算机软件的一门新兴学科。

2．什么是需求分析？该阶段的基本任务是什么？

答：需求分析指：开发人员要准确理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的形式功能规约（需求规格说明）的过程。

需求分析的基本任务是：

要准确地定义新系统的目标，为了满足用户需要，回答系统必须“做什么”的问题。在可行性研究和软件计划阶段对这个问题的回答是概括的、粗略的。主要任务是：

1）问题识别

(1)功能需求：所开发的软件必须具备什么样的功能，这是最重要的。

(2)性能需求：开发软件的技术性能指标，如存储容量、运行时间等限制。

(3)环境需求：软件运行时所需要的软、硬件(如机型、外设、操作系统、数据库管理系统等)的要求。

(4)用户界面需求；人机交互方式、输入输出数据格式等等。

另外还有可靠件、安全件、保密件、可移梢件、可维护性等方面的需求，这些需求般通过双方交流、调查研究来获取，并达到如同的理解。

2）分析与综合，导出软件的逻辑模型

分析人员对获取的需求，进行—致性的分析检查，在分析、综合中逐步细化软件功能并划分成各个子功能。这里也包括对数据域进行分解，并分配到各个于功能上，以确定系统的构成及主要成分，并用图文结合的形式，建立起新系统的逻辑模型。

3）编写文档

(1)编写“需求规格说明书”，把双方共同的理解与分析结果用规范的方式描述出来作为今后各项工作的基础。

(2)编写初步用户使用于册，着重反映被开发软件的用户功能界面和用户使用的具体要求，用户手册能强制分析人员从用户使用的观点考虑软件。

(3)编写确认测试计划，作为今后确认和验收的依据。

(4)修改完善软件开发计划。在需求分析阶段对待开发的系统有了更进一步的了解，所以能更准确地估计开发成本、进度及资源要求，因此对原计划要进行适当修正。

3. 什么是结构化维护？结构化维护的特点是什么？

若采用软件工程的方法开发软件，则各阶段都有相应的文档，容易进行维护工作，这是一种结构化的维护。

有软件工程思想开发的软件具有各个阶段的文档，这对于理解和掌握软件功能、性能、系统结构、数据结构、系统接口和设计约束有很大作用。进行维护活动时，首先从评价需求说明开始，搞清楚功能、性能上的改变，然后对设计说明文档进行评价，对设计说明文档进行修改和复查；根据设计的修改，再进行程序的变动；其后根据测试文档中的测试用例进行回归测试；最后，把修改后的软件再次交付使用。这对于减少精力、减少花费、提高软件维护效率有很大作用。

4.数据流图的作用？它的优缺点？其中的符号表示什么含义？

数据流图是用来表示系统的功能的工具。它表示系统的逻辑模型描述了数据流在系统中流动的情况，它是一种功能模型，它用来定义系统的功能，它不能反映“如何”实现。

数据流图只描述了系统的分解，没有表达出每个数据流、数据存储和加工的具体含义。

数据流图的作用主要有以下几条：

(1)系统分析员用这种工具可以自顶向下分析系统信息流程。 (2)可在图上画出需要计算机处理的部分。 (3)根据数据存贮，进一步作数据分析，向数据库设计过渡。 (4)根据数据流向，定出存取方式。 (5)对应一个处理过程，用相应的语言、判定表等工具表达处理方法。

箭头表示数据流、圆圈表示加工、双横线表示存储、方框表示数据的原点和终点。

5．详细设计的基本任务是什么？

答：详细设计又称过程设计，“怎样实现”这个软件系统，对系统中的每个模块给出足够详细的过程性描述。具体任务如下：

(1)为每个模块进行详细的算法设计，用某种图形、表格、语言等工具将每个模块处 理过程的详细算法描述出来。

(2)为模块内的数据结构进行设计，对于需求分析、概要设计确定的概念性的数据类

型进行确切的定义。

(3)对数据库进行物理设计，即确定数据库的物理结构。物理结构主要指数据库的存储记录格式、存储记录安排和存储方法，这些都依赖于具体所使用的数据库系统。

(4)其他设计

根据软件系统的类型，还可能要进行以下设计：

①代码设计。为了提高数据的输入、分类、存储、检索等操作，节约内存空间，对数据库中的某些数据项的值要进行代码设计。 ②输入／输出格式设计。 ③人机对话设计。对于一个实时系统，用户与计算机频繁对话，因此要进行对话方式、内容、格式的具体设计。

(5)编写详细设计说明书。 (6)评审。对处理过程的算法和数据库的物理结构都要评审。

6.什么是面向数据流的设计方法？它有哪些策略？

面向数据流的设计方法也称结构化设计方法(SD)，它与结构化分析(SA)相衔接，它按一定的设计策略将数据流图转换成软件的模块层次结构。有两种设计策略：

(1)事务型分析设计：一个大的复杂的系统分解成较小的，相对简单的子系统，这些子系统彼此之间相对独立一些，而高层数据流图的数据处理往往反映这些子系统的功能，有平行分别处理的特点，因此，高层数据流图的转换通常可作为事务型处理，把一个加工逻辑看成是一类特定的事务，把它们分别映射成一个模块，最高层模块为系统模块，通过对输入初始命令的判断决定调用哪个模块。这种事务型分析设计的策略也用于较低层数据流图向软件结构图的转换。

(2)变换型分析设计：变换型数据流图具有主要的处理功能及实现这项处理功能所需要的输入数据流和经过处理后产生的输出数据流。确定了第三部分，高层模块就可分解出三个从属于它的新模块，分别执行输入、变换、输出功能。变换分析设计一般用于对较低层数据流图向软件结构图的转换。

软件结构图设计好后，还需要为每个模块提供必要的说明，如功能说明、接口说明等，IPO图是常采用的图形工具，软件结构图和各个模块的IPO图结合在一起才能较完整地描绘软件系统在总体上对需求功能的实现。

7．什么是用例建模？如何进行用例建模？

答：用例建模是直接面向用户的，主要以需求陈述为基本依据，有关系统的业务边界、使用对象等，是构造系统用例模型的基本元素。

用例建模的步骤如下：

·从几方面识别系统的执行者，包括需要从系统中得到服务的人、设备和其他软件系统等；

·分析系统的业务边界或执行者对系统的基本业务需求，并将其作为系统的基本用例；

·分析基本用例，将基本用例中具有一定独立性的功能，特别是具有公共行为特征的功能分解出米，将其作为包含用例供基本用例使用； ·分析基本用例功能以外的其他功能，将其作为扩展用例供基本用例进行功能扩展； ·分析并建立执行者与用例之间的通信关系。