1、计算机语言发展体现了什么特点？与软件工程的关系？

（1）计算机语言发展从低级到高级对硬件要求越来越低；（2）所描述的数据结构越来越复杂；（3）描述程序特征越来越抽象。

软件工程的目标是消除与一段层，随着计算机语言的发展，在软件工程需求阶段使用的是自然语言；在设计阶段使用的是专业语言；在编码阶段使用的是计算机语言。

2、软件测试有哪些阶段，不同阶段对应软件工程中哪些文档？

软件测试分成四个阶段：单元测试、集成测试、系统测试、验收测试；

单元测试是对软件系统中的每一功能模块进行测试，对应于详细设计文档和单元测试用例数据；

集成测试是将各个模块集成在一起对其进行测试，对应于总体设计文档和集成测试用例数据；

系统测试是对整个软件系统进行测试，对应于需求规格说明书和系统测试用例数据；

验收测试是在用户参与下对软件的测试，对应于用户手册、安装手册等文档。

3、传统面向过程方法学和面向对象方法学的比较。

（1）从方法的角度来说，传统面向过程方法学是用结构化的技术来完成软件开发的各项任务；面向对象方法学则是将面向对象思想加入软件开发中，认为数据是主线，将数据和数据操作结合起来的方法学。

（2）从过程的角度来说，传统面向过程的方法学典型的过程模型是瀑布模型；面向对象方法学典型的过程模型是喷泉模型。

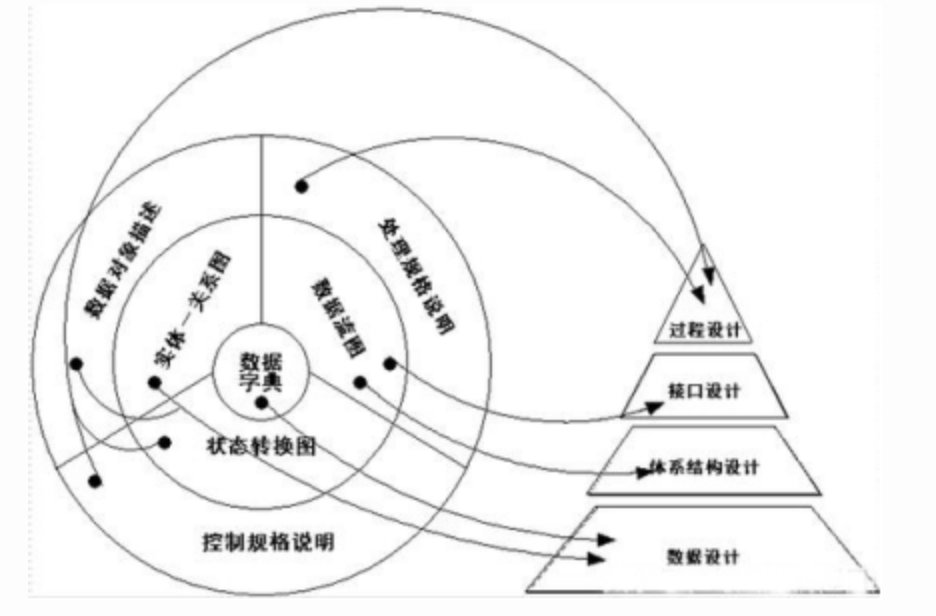
（3）从工具的角度来说，传统面向过程的方法学是使用数据流图、判定表和判定树等工具；面向对象方法学是使用顺序图、类图等工具。

（4）传统面向过程的方法学适用于需求明确、简单的软件系统的开发；面向对象的方法学适用于需求不明确、复杂大型系统的开发。

4、简述分析模型和设计模型，分析模型和设计模型之间的映射关系？（或者问怎样理解“分析是通往设计的桥梁”）

（1）分析模型的核心是数据，还包括实体关系图（E-R图）、数据流图和状态转换图，实体关系图是对数据对象的描述，数据流图是对加工规格的说明，状态转换图是对控制规格的说明。

（2）设计模型分成数据设计、体系结构设计、接口设计和过程设计。



映射关系：

（1）数据设计是将实体关系图中描述对象和对象之间的关系，一集数据字典中描述的详细数据内容转为数据结构的定义。

（2）体系结构设计定义软件系统各主要成分的关系，根据数据流图设计数据在模块之间的流向。

（3）接口设计根据数据流图定义软件内部各成分之间、软件与其他协同系统之间及软件与用户之间的交互机制。

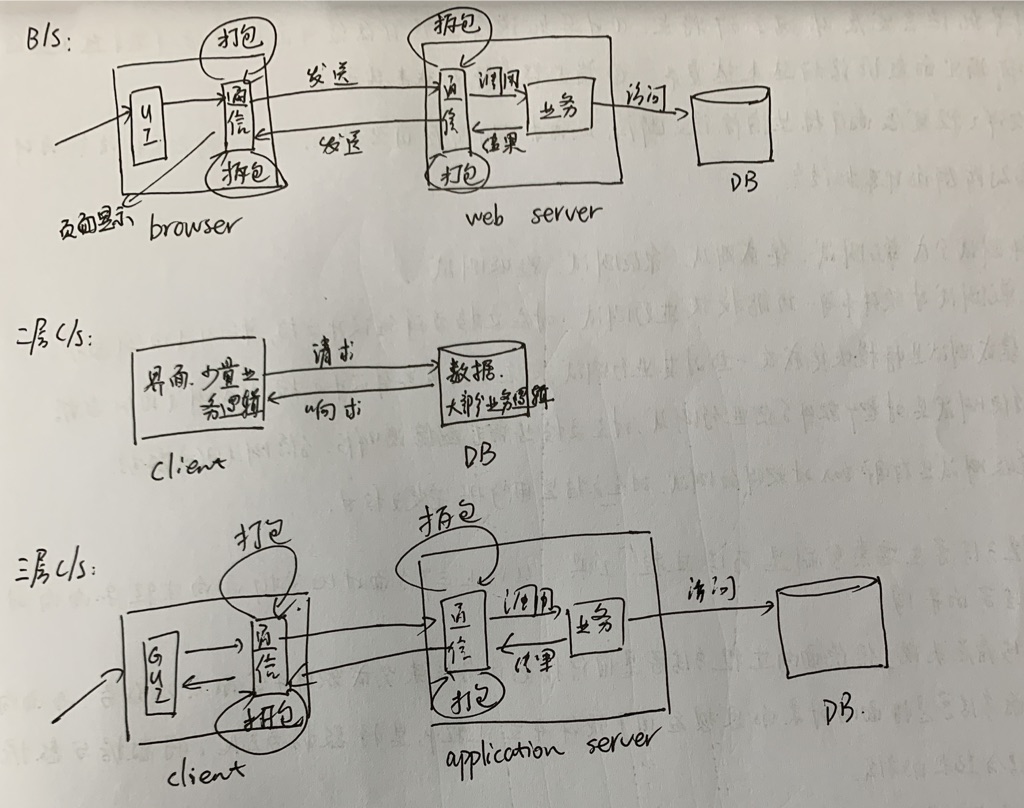
（4）过程设计把结构成分转化为软件的过程性描述，牵扯到数据状态的转换，以及状态变换的方式。

5、比较B/S、二层C/S、三层C/S体系架构。

B/S架构：浏览器／服务器体系架构；客户端使用浏览器，服务器使用web服务器，客户端与服务器之间使用标准化http协议进行通信。

二层C/S架构：使用数据库服务器充当服务器，在数据库编写业务逻辑，客户端主要负责提供操作界面和少量业务逻辑。

三层C/S架构：数据库只用来存储数据，应用服务器用来编写业务逻辑，客户端只提供界面。



6、白盒测试（包括程序图、环路复杂性、基本路径测试及测试用例）见附件

7、黑盒测试（因果图法）见附件

8、模块粒度值随着软件工程的发展逐渐变大，智能性和自主性提高。

9、软件过程与软件过程过型？分析主要的软件过程模型。

软件过程是指开发软件产品的一组活动及其结果。软件过程模型是从一特定的角度对软件过程本质的描述。传统的面向过程的软件过程模型是瀑布模型，典型面向对象的软件过程模型是喷泉模型。瀑布模型是将软件开发分成一系列活动，需求分析、设计、编码、测试和维护，在瀑布模型中，所有的活动都是要在上一步活动完成的情况下才能进行的。

10、CASE（Computer-Aided Software Engineering）计算机辅助软件工程，CASE是支持软件工程实施的一系列工具的集合。低端的CASE工具是支持实现和测试的工具；高端的CASE是支持分析和设计的工具。

11、需求阶段获取的主要内容

物理环境、接口、用户或人的因素、功能、文档、数据、资源、安全性、质量保证

12、SOA主要内容及作用。

SOA又成为面向服务的体系结构，是传统软件体系结构结合了软件构件技术而演化而来的一种新的风格的体系结构。SOA提供了一种面向服务、大粒度、松耦合以及动态绑定的分布式应用系统的构建方法。SOA是通过调用一些可以独立运行、松耦合的模块来完成所需要的系统功能；各个模块的运行独立于具体的实现平台，模块之间通过接口通信进行交互。

13、Web Services及其Web Services软件工程。

Web Services是构件技术在网络环境下的演化，是一种能够被描述并通过网络发布、发现和调用的自包含、自描述、松散耦合的软构件。Web Services采用标准协议，用接口定义、方法调用的方式来进行使用，解决面向Web的分布式计算问题。

Web Services的优点：①协议的通用性，即采用标准的Internet协议；②完全的平台和语言独立性，即Web Services采用XML数据格式进行通信和交互，不需要以往构件那样在特定的框架或模式下运行；③具有更高的灵活性。

14、基线以及基线的特点？

基线是软件工程活动从一个环节转入另一个环节时对阶段产品或组件的标识。基线的特点是：有唯一的标识；具体的内容；经过审批；严格地控制变更。

15、构件（组件）。

构件是指语义完整、语法正确和有可重用价值的单位软件，是软件重用过程中可以明确辨识的系统；结构上，它是语义描述、通讯接口和实现代码的复合体。

构件模型的三个流派：

OMG（Object Management Group，对象管理集团）的CORBA（Common Object Request Broker Architecture，通用对象请求代理结构）

Sun的EJB（Enterprise Java Bean）

Microsoft的DCOM（Distributed Component Object Model，分布式构件对象模型）。

16、软件体系结构及其风格？

软件体系结构为软件系统提供了一个结构、行为和属性的高级抽象，由构成系统的元素的描述、这些元素的相互作用、指导元素集成的模式以及这些模式的约束组成。软件体系结构不仅指定了系统的组织结构和拓扑结构，并且显示了系统需求和构成系统的元素之间的对应关系，提供了一些设计决策的基本原理。

软件体系结构风格是描述某一特定应用领域中系统组织方式的惯用模式。体系结构风格反映了领域中众多系统所共有的结构和语义特性，并指导如何将各个模块和子系统有效地组织成一个完整的系统。经典体系结构风格包括：数据流风格、调用／返回风格、独立构件风格、C/S风格、B/S风格、三层C/S风格等。

17、常见的软件过程模型包括瀑布模型、原型模型、增量模型、螺旋模型、迭代模型、喷泉模型等等。

18、（1）模块是由边界元素限定的相邻程序元素的序列，模块是构成程序的基本构件。

（2）模块化就是把程序划分成独立命名且可独立访问的模块，每个模块完成一个子功能，将这些模块集成起来构成一个整体，就能完成指定功能满足用户的要求。

（3）抽象是将事物中相似的部分集中概括起来，暂时忽略它们之间的差异。

（4）信息隐藏就是使一个模块中的信息对于不需要这些信息的模块来说是不可访问的。

（5）模块独立是模块话、抽象、信息隐藏和局部化的结果。开发具有功能独立且与其他模块之间没有过多的相互作用的模块，就可以做到模块独立。

（6）耦合是对一个软件结构不同模块间相互连接的程度；内聚是一个模块内各元素彼此结合的紧密程度。