1. **选择题**
2. 以下哪个不是软件危机的主要表现？

A．软件开发进度难以预测 B．软件开发成本难以控制

C．用户对产品功能难以满足 D．软件开发复杂度越来越高

1. 以下哪种模型是线性开发模型

A．喷泉模型 B．螺旋模型

C．增量模型 D．快速原型

1. 软件工程学的目的应该是最终解决软件生产的( )问题。

A．提高软件的开发效率 B．使软件生产工程化

C．加强软件的质量保证 D．消除软件的生产危机

1. 软件危机的主要原因有( )。

①软件本身的特点 ②用户使用不当

③硬件可靠性差 · ④对软件的错误认识

⑤缺乏好的开发方法和手段

A．③④ B．①②④

C．①③ D．①⑤

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( )。

A．硬件环境

B．开发人员的素质

C．软件开发工具和软件开发的环境

D．软件开发的环境

1. 软件就是()
2. 程序+数据结构+文档
3. 程序+算法+文档
4. 程序+文档
5. 程序+文档+算法+数据结构
6. 在软件开发领域,\_\_\_\_已经迅速取代了生命周期方法学。
7. 传统方法学
8. 面向对象方法学
9. 瀑布模型开发方法学
10. 敏捷开发方法学
11. 有几个主要的软件开发模型

A．3

B. 4

C. 5

D. 2

1. 软件工程中的各种方法是完成软件工程项目的技术手段，它们支持软件工程的( )阶段

A．各个

B．前期

C．中期

D．后期

1. 下列关于瀑布模型的描述正确的是（  ）。

A．利用瀑布模型，如果发现问题修改的代价很低

B．瀑布模型的核心是按照软件开发的时间顺序将问题简化

C．瀑布模型具有良好的灵活性

D．瀑布模型采用结构化的分析与设计方法，将逻辑实现与物理实现分开

1. **判断题**

（1）软件危机指的是是指落后的软件生产方式无法满足迅速增长的计算机软件需求，从而导致软件开发与维护过程中出现一系列严重问题的现象 T

（2）传统方法学也称为生命周期方法学或结构化范型。 T

（3）软件生命周期由软件定义、软件开发和运行维护(也称为软件维护)3个时期组成，每个时期又进一步划分成若干个阶段。 T

（4）按照传统的快速原型模型开发软件具有阶段间具有顺序性和依赖性，推迟实现的观点和质量保证的观点。 F

（5）面向对象方法学的优点是降低了软件产品的复杂性，提高了软件的可理解性，简化了软件的开发和维护工作。 T

1. **简答题**
2. 什么是软件危机？

答：软件危机是指在计算机软件开发、使用与维护过程中遇到的一系列严重问题和难题。它包括两个方面：如何开发软件，已满足对软件日益增长的需求；如何维护数量不断增长的已有软件。

1. 软件维护的目的和主要任务是什么?维护的类型有哪些?

答：软件维护是软件生命周期中最后一个、持续时间最长的阶段。

主要任务是在软件已经交付使用之后，为了改正错误或满足新的需要而修改软件，目的是提高软件的可靠性、可用性，延长软件的寿命。维护的类型有：改正性(错误)、适应性(环境)、完善性(功能)、预防性(新产品)。

1. 软件生命周期有几个阶段，各阶段的基本任务是？

答：软件生命周期为7个阶段：

1. 问题的定义：要解决的问题是什么
2. 可行性研究：确定问题是否值得解决，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. 编码和单元测试：编码和测试
7. 运行维护：保证软件正常运行。
8. **论述题：**
9. **请简要概述什么是软件工程学（从软件工程的基本原理，软件生命周期，软件过程等方面回答）**

软件工程是指导计算机软件开发和维护的一门工程学科，采用工程的概念，原理，技术和方法来开发与维护软件。将管理技术和技术方法相结合，经济地开发出高质量的软件并有效地维护它，采用生命周期方法学将软件生命周期分为若干个相对独立的阶段，每个阶段完成一些确定的任务并在每个阶段镜像严格的技术审查和管理复查。软件过程是为了获得高质量的软件产品所需要完成的一系列的任务的框架。其中有瀑布模型，快速原型模型，增量模型等。

1. **请简述瀑布模型(从核心思想，优势等方面回答)**

瀑布模型是一个项目开发架构，开发过程是通过设计一系列阶段顺序展开的，从系统需求分析开始直到产品发布和维护，每个阶段都会产生循环反馈，因此，如果有信息未被覆盖或者发现了问题，那么最好 “返回”上一个阶段并进行适当的修改，项目开发进程从一个阶段“流动”到下一个阶段。瀑布模型核心思想是按工序将问题化简，将功能的实现与设计分开，便于分工协作，即采用结构化的分析与设计方法将逻辑实现与物理实现分开。将软件生命周期划分为制定计划、需求分析、软件设计、程序编写、软件测试和运行维护等六个基本活动，并且规定了它们自上而下、相互衔接的固定次序，如同瀑布流水，逐级下落。