

三级信息管理技术分章节考试要点：第二章 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/237/2021_2022__E4_B8_89_E7_BA_A7_E4_BF_A1_E6_c97_237270.htm

第二章 软件工程
自从1968年首次提出软件工程一词以来，软件工程已成为计算机软件的一个重要分支和研究方向。软件工程是指应用计算机科学、数学及管理科学等原理，以工程化的原则和方法来解决软件问题的工程。其目的是提高软件生产率、提高软件质量、降低软件成本。

一、软件工程基本概念

早期的软件主要指程序。程序的开发采用个体工作方式，开发工作主要依赖于开发人员的个人技能和程序设计技巧。当时的软件通常缺少与程序有关的文档，软件开发的实际成本和进度往往与预计的相差甚远，软件的质量得不到保证，开发出来的软件常常不能使用户满意。随着计算机应用的需求不断增长，软件的规模也越来越大，然而软件开发的生产率远远跟不上计算机应用的迅速增长。此外，由于软件开发时缺少好的方法指导和工具辅助，同时又缺少有关的文档，使得大量已有的软件难以维护。上述这些问题严重地阻碍了软件的发展，20世纪60年代中期，人们把上述软件开发和维护中的各种问题称为“软件危机”。1968年在德国召开的NATO会议上，首次提出了“软件工程”一词，希望用工程化的原则和方法来克服软件危机。在此以后，人们开展了软件开发模型、开发方法、工具与环境的研究，提出了瀑布模型、演化模型、螺旋模型、喷泉模型等开发模型，出现了面向数据流方法、面向数据结构的方法、面向对象方法等开发方法，以及一批CASE（computer aided software engineering）工具和环境。

(一) 软件生存周期如同人的一生要经历婴儿期、少年期、老年期直至死亡这样一个全过程一样，任何一个软件产品或软件系统也都要经历软件定义、软件开发、软件维护直至被淘汰这样一个全过程，我们把软件的这一全过程称为软件生存周期。软件定义、软件开发、软件维护等阶段还可分为若干个阶段，每个阶段相对独立又彼此有联系，上一阶段的工作结果是下一阶段工作的依据，下一阶段是上一阶段的进化，它更接近于问题的解。

1. 软件定义 软件定义阶段主要解决的问题是待开发的软件要“做什么”，也就是要确定软件的处理对象、软件与外界的接口、软件的功能和性能、界面以及有关的约束和限制。软件定义阶段通常可分成系统分析、软件项目计划、需求分析等阶段。

(1) 系统分析 这里讲的系统是指计算机系统，包括计算机硬件、软件和使用计算机的人。系统分析的任务是确定待开发软件的总体要求和适用范围，以及与之有关的硬件、支撑软件的要求。系统分析阶段的参加人员有用户、项目负责人、系统分析员。该阶段产生的文档可合并到软件项目计划阶段的文档（项目计划书）中。

(2) 软件项目计划 软件项目计划的任务是确定待开发软件的目标，对其进行可行性分析，并对资源分配、进度安排等做出合理的计划。软件项目计划阶段的参加人员有用户、项目负责人、系统分析员。该阶段所产生的文档有可行性分析报告、项目计划书。

(3) 需求分析 需求分析的任务是确定待开发软件的功能、性能、数据、界面等要求，从而确定系统的逻辑模型。需求分析阶段的参加人员有用户、项目负责人和系统分析员。该阶段产生的文档有需求规约（requirements specification），习惯上称它为需求规格说明书。

2. 软件开发软

件开发阶段主要解决的问题是该软件“怎么做”，包括数据结构和软件结构的设计，算法设计，编写程序，测试，最后得到可交付使用的软件。软件开发阶段通常可分成软件设计、编码、软件测试等阶段。（1）软件设计软件设计通常还可分成概要设计和详细设计。概要设计的任务是模块分解，确定软件的结构、模块的功能和模块间的接口，以及全局数据结构的设计。详细设计的任务是设计每个模块的实现细节和局部数据结构的设计。概要设计阶段的参加人员有系统分析员和高级程序员，详细设计阶段的参加人员有高级程序员和程序员。设计阶段产生的文档有设计规约（design specification），也称为设计说明书，它也可分为概要设计说明书和详细设计说明书。根据需要还可产生数据说明书和模块开发卷宗。（2）编码编码的任务是用某种程序语言为每个模块编写程序。编码阶段的参加人员有高级程序员和程序员，产生的文档有程序清单。（3）软件测试软件测试的任务是发现软件中的错误，并加以纠正。软件测试阶段的参加人员通常由另一部门（或单位）的高级程序员或系统分析员承担，该阶段产生的文档有软件测试计划和软件测试报告。

3.软件维护

软件开发阶段结束后，软件即可交付使用。软件的使用通常要持续几年甚至几十年，在整个使用期间，都可能因为某种原因而修改软件，这便是软件维护。引起修改软件的原因主要有三种：一是在软件运行过程中发现了软件中隐藏的错误而修改软件。二是为了适应变化了的环境而修改软件。三是为修改或扩充原有软件的功能而修改软件。因此软件维护的任务就是为使软件适应外界环境的变化、实现功能的扩充和质量的改善而修改软件。软件维护阶段的参加人员是维护人员，该阶段

产生的文档有维护计划和维护报告。目前，软件生存周期各阶段的划分尚不统一，有的分得粗些，有的分得细些。许多场合软件开发阶段都是从需求分析阶段开始的。本书中，我们也将需求分析看作为软件开发的开始阶段。（二）软件开发模型

为了指导软件的开发，用不同的方式将软件生存周期中的所有开发活动组织起来，形成不同的软件开发模型。常见的软件开发模型有瀑布模型、演化模型、螺旋模型、喷泉模型等。瀑布模型如下图所示，它是1970年由W.Royce提出的。该模型给出了软件生存周期各阶段的固定顺序，上一阶段完成后才能进入到下一阶段，整个过程就像流水下泻，故称之为瀑布模型。图中的虚线部分表示在某一阶段发现错误时，其错误可能是由上一阶段造成的，因此开发过程可能要反馈到上一阶段。

（三）软件开发方法

软件开发过程模型规定软件开发活动的组合应用方式，要保证开发活动的高质量，还需要有相应的软件开发方法作为技术支持。近10年来，软件工作者研制出了许多工程化的软件开发方法，例如70年代初提出的用于编写程序的结构化程序设计方法，确实起到了提高效率，减少错误的效果。但是70年代中期，软件工作者认识到编写程序仅仅是软件开发的一个环节，而合理地建立系统结构比编定程序更为重要。所以研究的重点前移到设计阶段，出现了设计阶段的结构化设计（SD）方法和JACKSON等方法，到了70年代后期，人们又发现事先对用户的要求进行分析更为重要，故又把重点前移到分析阶段。出现了用于分析阶段的结构化分析（SA）方法、结构化分析与设计技术（SADT）等。随着计算机技术的迅速发展，在80年代初期的实时、并发和网络等软件的开发过程中，特别是在第五代计

计算机研究工作中，又提出了面向对象的设计方法。现在流行的方法有多种，它们的适用范围也各不相同。有的适用于一般的数据处理系统，如SA、SD（两者统称为结构化分析与设计方法，即Yourdon方法）、JACKSON方法。有的适用于大型的复杂系统，如SADT技术。有的适用于实时事务处理系统，如FSM方法。有的适用于并发软件系统，如PETRI网方法。作为90年代代表作的面向对象方法，其应用已几乎遍布各个领域。这些方法除了适用范围不同外，方法形成的基础、处理规则和对所开发软件风格的要求等都各有侧重。用什么方法来说明用户的要求、用什么方法来设计软件以及用什么方法对软件进行测试和维护，直接影响所开发软件的质量。（四）软件开发工具早期的软件开发除了一般的程序设计语言外尚缺少工具的支持，致使编程工作量大，质量和进度却难以保证，导致人们将很多的精力和时间花费在程序的编制和调试上。相比之下，在更重要的软件的需求和设计上反而得不到必要的精力和时间投入。软件开发工具的发展促进了软件开发的高速度和高质量。工具的发展是从单项工具的开发逐步走向集成的工具发展的。同时，软件开发方法的有效应用也必须得到相应工具的支持，否则方法将难以有效的实施。工具的完善和发展将促进软件开发的进步和完善。原型化方法的实施基础就是得到了开发工具的支持。快速原型化之所以能够实现的基础就是原型化人员在快速建模时得到了工具的支持，否则原型化方法是无法实施的。（五）软件开发环境软件工程环境或称软件开发环境是全面支持软件开发全过程的软件工具集合。这些软件工具按照一定的方法或模式组合起来，并能支持软件开发生命周期的各个阶段和各项任务的

完成。CASE，即计算机辅助软件工程环境是当前软件开发环境中富于特色的研究工作和发展方向，它的成功将最大限度地降低软件工程的技术难度并使软件开发的质量得到保证。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com