

软件工程名词解释

范围：2005（A、B）、2007、2009、2011、2012、2016、2018、2019、2022 模拟题
括号内为出现次数

（2）统一过程（UP）

用例驱动的、以基本架构为中心的、迭代式和增量性的软件开发过程框架。

（7）项目进度

对特定项目的软件开发周期的刻画。包括对各个离散活动的交互关系的描述，以及对各个活动完成时间及整个项目完成时间的初步估算。

（3）重用

重复采用以前开发的软件系统中具有共性的部件，用到新的开发项目中去。

（1）项目成本

为支持软件开发而购买软件和工具的开支+用于支持需求分析，设计，编码，测试，处理需求变更等等+工作量开支。

（2）需求

是对来自用户的关于软件系统的期望行为的综合描述，涉及系统的对象、状态、功能和约束等。

（2）耦合

两个软件部件之间的相互关联程度。

（1）关联

两个类之间的连线称为关联，表示类的实体之间的关系。

（1）极限编程

极限编程是一种轻量级的软件开发方法论，属于敏捷开发方法。XP 主要特征是要适应环境变化和需求变化，充分发挥开发人员的主动精神。

（3）驱动模块

代替上层模块的调用程序，是调用特定构件并向其传递测试用例的程序。

（2）β 测试

由客户实际进行测试。

（2）软件工程

采用工具、技术等用来解决现实问题的综合过程。

（2）文档缺陷

文档和代码的实际内容不匹配。

（2）分阶段模型

系统被设计成部分提交，用户每次只能得到部分功能，而其他部分处于开发过程中。

（5）派对编程

开发方式是两个程序员共同开发程序，且角色分工明确：一个负责编写程序，一个负责复审和测试，两人定期交换角色。

（3） α 测试

向客户发布系统前，开发者先自己组织团队（或者委托其他团队）测试这个系统（内部测试）。

（3）组合

当一个类是另一个类的部分时，二者关系为组合。

（1）统一开发过程（RUP）

是一个面向对象且基于网络的程序开发方法论，RUP 模型是 IBM 提出的提供支持和包装的 UP 模型。

（3）抽象

基于某种归纳水平的问题描述，他是我们将注意力集中在问题的的关键方面而非细节。

（1）原型

一种部分开发的产品，用来让用户和开发者共同研究，提出意见，为最终产品定型。

（2）功能性需求

描述系统内部功能或系统与外部环境的交互作用。涉及系统输入应对，实体状态变化，输出结果，设计约束与过程约束等。

（2）公共耦合

不用模块访问公共数据。

（3）用例

描述系统外部的执行者与系统的用例之间的某种联系。

（2）路径覆盖测试

借助于程序图设计测试用例的一种白盒方法，测试用例基于流程图来设计。

（1）基线

是指软件文档和其他资料的集合，它们代表了产品在某一时间点的情况（以及其他参考点）。

（2）软件可靠性

软件系统在给定的时间间隔和给定的条件下运行成功的概率。

（1）能力缺陷

系统的处理超过需求描述中的最大负载就会引发压力缺陷。

（2）抛弃式原型

仅用于了解问题、探索可行性，并不打算用来作为将来实际提交系统的一部分，而是用完扔掉。

（1）增量式开发

系统需求按照功能分成若干子系统，开始建造的版本是规模小的、部分功能的系统，后续版本添加包含新功能的子系统，最后版本是包含全部功能的完整系统。

（1）演化型原型

该原型的开发是用于了解问题，并作为将来准备提交的系统的一部分。

（1）面向对象

是一种软件开发方法，它将问题和方法组织成一系列独立的不连续的对象，数据结构和动作都被包括在内。

（2）过载缺陷

软件内部设定的数据结构被填充后溢出，系统不再能完成各种功能。

（1）强度测试

系统在短时间内加载极限负荷，以验证系统性能。（例如：系统在最大设备数量的情况下进行性能测试）

（1）软件可维护性

是指在给定的使用条件（预定的时间间隔、维护程序、维护资源之下进行维护）下，维护活动能被执行的概率。

（1）由底向上测试

集成测试时从模块结构图的最低层开始，由下而上按调用关系逐步添加新模块，组成子系统并分别测试，直到全部模块组装完毕为止。

（3）过程

软件工程是使用工具、技术、过程和模式等用来解决现实问题，提高软件产品质量的综合过程。

（1）能力缺陷

系统活动到达指定的极限时系统性能会变得不可接受。

（2）迭代式开发

迭代开发是统一开发过程(RUP)的关键实践，开发被组织成一系列固定的短期小项目，每次迭代都产生经过测试、集成并可执行的局部系统。每次迭代都具有各自的需求分析、设计、实现和测试。随着时间和一次次迭代，系统增量式完善。

（1）开发模式（今年加了双斜杠）

表示开发软件时特定的方法或哲学。是软件开发的全部过程，活动和任务的结构框架，它能直观的表达的表达式软件开发全过程，明确要完成的主要活动，任务和开发策略。

（1）容量测试

验证系统处理巨量数据的能力。

（1）测试用例

对一项特定的软件产品进行测试任务的描述，体现测试方案、方法、技术和策略。（找不到，百度搜的）

（1）生命周期

软件开发过程描述了软件产品从概念到实现、交付、使用和维护的整个过程，因此，有时把软件开发过程称为软件生命周期。

（2）功能性需求

描述系统内部功能或系统与外部功能的交互作用，涉及系统输入应对、实体状态变化、输出结果、设计约束、过程约束等。根据要求的活动来描述需求的行为。（功能需求定义问题解决方案空间的边界）

（1）设计模式

一种针对单个软件模块或少量模块而给出的一般性解决方案，它提供较低层次的设计决策。它是一个共同的设计结构的关键方面，包括对象和实例，角色和协作，责任分配。

（1）螺旋模型

螺旋模型将瀑布模型和快速原型模型结合起来，强调了其他模型所忽视的风险分析，特别适合于大型复杂的系统。

（2）黑盒测试

测试人员在完全不了解程序内部的逻辑结构和内部特性的情况下，只依据程序的需求规格及设计说明，检查程序的功能是否符合它的功能说明。