

<b>B</b>			
边界值 (P169)	白盒测试 (P174)	bug (P160)	
<b>C</b>			
CMMI (P198)	测试需求 (P118)		
<b>D</b>			
迭代模型 (P37)	订单软件 (P46)	定义软件过程 (P85)	等价类 (P168)
<b>F</b>			
非订单软件 (P46)	风险分析 (P86)	非结构化维护 (P190)	
<b>G</b>			
功能模型 (P93)	构件 (P149)		
<b>H</b>			
合同 (P1)	黑盒测试 (P167)		
<b>J</b>			
基线 (P58)	结构化维护 (P190)		结对编程 (P207)
<b>L</b>			
立项建议书 (P1)	里程碑 (P58)	LOC (P87)	
<b>M</b>			
敏捷文化 (P26)			
<b>R</b>			
软件 (P3)	软件工程 (P5)	软件工程课程 (P6)	软件工程基本原理 (P7)
软件生命周期模型 (P10)	软件测试 (P160)	软件测试的 V 模型 (P166)	软件缺陷 (P160)
软件质量 (P213)	软件配置项 (P208)	软件配置管理的目标 (P208)	任务书 (P1)
<b>S</b>			
审计 (P58)	数据模型 (P94)	三层结构设计 (P118)	
<b>X</b>			
XP 模型 (P41)	需求分析 (P55)	项目 (P217)	
项目管理 (P217)			
<b>Y</b>			
原型模型 (P34)	业务模型 (P93)		
<b>Z</b>			
中间件 (P149)			
<b>其他</b>			
9 个核心流程 (P38)			

## P1:

合同是与固定客户签订的协议书，签订合同后软件公司启动该项目的开发，该软件被称为“订单软件”。

立项建议书是相对“非订单软件”而言的，是相关人员对立项过程的书面描述。

任务书是企业决定开发某个软件时，对此任务的具体部署情况，以书面的形式表达出来，包括正文和附件。

## 分析理解

- 1、论述软件工程在软件行业的作用（P9）
- 2、为什么说面向对象方法能描述无穷的世界？（P12）
- 3、分析面向过程方法的缺点（P11）
- 4、为什么要面向流程进行需求分析？（P22）
- 5、为什么迭代模型中要以业务建模作为第一个核心流程？（P38）
- 6、需求获取的三原则是什么？怎样理解？（P56）
- 7、论述需求分析的任务（P58）
- 8、分析网络操作系统中的三个模型（功能模型、业务模型、数据模型）（P95）
- 9、数据库设计的步骤是什么？他们与软件生命周期各个阶段的关系是什么？（P100）
- 10、论述数据库设计中的原子化理论（P106）
- 11、谈谈对软件设计基本原理的理解（P120）
- 12、以软件中常见的登录功能举例说明软件三层体系结构的设计（P119）
- 13、以软件中常见的登录功能举例说明软件开发中的三个模型（功能模型、业务模型、数据模型）的建模思想
- 14、论述软件维护的最新方法（P192）
- 15、CMMI 的 5 个级别各有哪些特征？（P201）
- 16、简述敏捷过程的开发原则（P207）
- 17、如何理解和实践项目经理对程序员的八项要求？（P219）

## CMMI 的 5 个级别各有哪些特征？

SW-CMM 的 5 个级别分别为：

初始级 (CMM1)：组织内部是人治，是英雄创造历史。

可重复级 (CMM2)：项目管理级，在组织内部重复使用项目管理的经验。

已定义级 (CMM3)：组织级管理，在组织内部已经达到了法律化管理，由项目组级管理发展到组织级管理，13 个 KPA 已制度化和法律化，组织级法律框架健全，工程过程和管理过程已文档化，软件测量数据库已开始建立。

已管理级 (CMM4)：定量管理或数据管理，在组织内部已经达到了量化管理，实现了定量的数据级管理，产品和项目级管理的经验已量化，组织级过程管理已标准化和量化，软件测量数据库已发挥量化管理的作用。

优化级 (CMM5)：组织已经达到了循环优化和与时俱进。

## 如何理解和实践项目经理对程序员的八项要求？

### (1) 团队协作精神的训练和要求

现在的软件开发不再是个人英雄主义打天下的时代，尤其是像微软这样的大软件公司，一个软件都是由几百人甚至几千人共同合作完成的，没有团队精神是无法想象的。

### (2) 数据库和数据结构分析与设计能力的训练和要求

程序员不但要学会看懂数据库和数据结构，而且要逐渐学会分析与设计数据库和数据结构。只有这样，初级程序员才能成长为高级程序员，高级程序员才能逐渐成长为系统分析员。

### (3) 文档习惯的训练和要求

良好的文档是正规研发流程中非常重要的环节，作为程序员，30%的工作时间写技术文档是很正常的，而作为高级程序员和系统分析员，这个比例在 70% 以上。

### (4) 规范化代码编写能力的训练和要求

良好的编写习惯，不但有助于代码的移植和纠错，也有助于不同技术人员之间的协作。代码具有良好的可读性，是程序员的基本工作需求。

### (5) 复用性能力的训练和要求

复用性设计、模块化思维，就是要程序员在完成任何一个功能模块或函数的时候，要多想一些，不要局限在完成当前任务的简单思路上，想想看该模块是否可以脱离这个系统存在，是否可以通过简单的参数修改方式，在其他系统和应用环境下直接引用，这样就能极大地避免重复性的开发工作。

### (6) 测试习惯的训练和要求

程序员在每段程序代码、每个构件或每个子模块完成后都进行认真的测试，就可以尽量将一些潜在的 Bug 问题尽早地发现和解决，这样对整个开发进程将会有很大的促进。

### (7) 学习和总结能力的训练和要求

程序员很容易被淘汰、很容易落伍的职业，因此，程序员必须不断跟进新技术，学习新技能，随时总结，找到自己的不足，逐步提高自己。

### (8) 引导程序员由“丑小鸭”变成“白天鹅”

项目经理要鼓励程序员将编程的实践经验上升到软件的抽象理论，又将软件的抽象理论返回到编程实践。引导与鼓励程序员超过项目经理，使“丑小鸭”变成“白天鹅”。