第一部分 管理你的项目：

第一章：交付高质量的产品

即便是质量低劣的软件，其质量要求也要高于其他类型的人类书面文字。编程是一项严谨的工作，所以从事此项工作的专业人员都是在完成对质量要求极为苛刻的任务。唯一的问题是，根据历史发展趋势，未来的软件将远比今天的软件更大、更复杂，这就意味着即使保持目前的缺陷等级，我们在未来的工作也必须达到更高的质量要求。

在软件开发过程中，为了保证质量始终如一，必须遵循以下8个步骤：

1. 确立质量控制的策略、目标和计划；
2. 正确训练、指导和支持开发人员及其团队；
3. 确立和维护软件需求的质量管理过程；
4. 确立和维护软件工程过程的统计控制；
5. 审查、检查并评估所有的产品制品；
6. 评估所有缺陷，加以更正并用以识别、纠正和预防其他类似问题；
7. 确立和维护配置管理和变更控制系统；
8. 持续改进开发过程。

软件产品的质量应当被定义为产品对用户的有用性。

必须要记住，只有得到了清晰的需求，才可能开发出高质量的程序。

软件工程师的工作就是在计划成本和计划进度内交付高质量的产品。软件产品必须满足用户的功能需求，同时也要能可靠、始终如一地完成用户的工作。能完成用户的工作是关键。尽管对于程序的使用者来说软件功能非常重要，但是只有在软件能够运行时，这些功能才可以使用。

在你实现程序的其他任何目标之前，软件自身的大部分缺陷必须是你首先要解决的问题。

你首先要做的是了解这类在开发过程中引入的缺陷，并且尽最大努力防止它们出现。要想做到这一点，你必须熟练运用所使用的编程语言，全面了解你的开发支持系统，并且精通将要开发的这类应用程序。

缺陷是指程序中的错误，例如语法错误、拼写错误、标点符号错误，也就是不正确的程序语句。缺陷是一种客观存在的事物，是可以识别、描述和统计的。

对每一位软件工程师而言，缺陷都是应当高度关注的，不仅因为它们会影响软件产品的使用，而且一般软件公司超过一半的精力都用于发现和修复这些缺陷。由于用于测试的时间非常昂贵，并且很难事先预计，所以缺陷通常是导致项目成本和时间进度出现问题的最主要原因。

当软件程序得到了广泛的应用，特别是以一些设计者事先没有预想到的方式应用时，那些表面看起来十分细微的错误就可能会产生无法预知的后果。特别是对广泛应用的软件系统进行扩展以满足新的需求时，那些潜伏的问题就有可能暴露出来，一个看似微小的缺陷都有可能会引起灾难性的后果。

人们常常把软件质量看做是最终的结果或终点。事实并非如此，这是一段永远都没有终点的旅程。

目标提供了一个任务和焦点，帮助我们确定优先次序并忽略那些不重要的细节。为了完成重要的工作，首先要清晰地知道什么是你要努力去实现的。方向模糊，目标不明确，就是在浪费时间。

目标之所以重要，主要是基于以下两条原因：它们提供了努力的焦点，而且建立了一种优先次序。

要坚持让你的问题得到解决。很多情况下你会发现，其他人遇到的问题和你的几乎完全一样，只是他们羞于开口询问罢了。通过想你的管理者请教，明白无误地了解他们想让你达到的目标是什么，只有这样你才能真正履行好自己的职责。如果他们不能做到这一点，那么在开始工作之前，把你对目标的理解整理出来，然后与他们核对。之后要确信你和他们达成了共识。

第二章：为高质量项目制订计划

计划分为两种类型。第一种的基于时间段的计划，这个时间段可以是日历上的任何一个片段。阶段计划关心在这一段时间内你准备如何利用时间。第二种是基于行动的计划，比如开发一个程序或撰写一份报告。

在任何行业，工作稳定的关键是单位的财务健康，而财务健康的关键在于精确的阶段计划和产品计划。

如果你想成为一位有效率的软件工程师，你就需要知道如何制订计划。制订计划的关键是实践，因此要想得到最佳的实践，从现在就应开始制订计划，并且在你今后所有的项目实践中都要制订计划。

制订产品计划的第一步是清楚地定义要生产的产品是什么。

作为开发人员，我们要为自己的工作负责。在一个不现实的进度下，很难甚至不可能制订一个可行的计划。

由于计划不当而造成的种种问题会使所有人都蒙受损失：客户得到的是产品延期和开销加大，管理者不得不耗费更多资源，而开发人员则名誉扫地。为了成功，公司必须实现他们的承诺。而我们为了履行职责，必须制订精确的计划。

最终得出的计划必须满足特定的要求。就一份计划而言，有五条基本的要求：易于理解、清晰明白、详细具体、精确缜密、准确无误。

与其计划好未来所有工作的细节，不如只粗略划分后期任务，只有在需要时再对其进行完善。

第二部分 管理你的团队：

第3章：高效团队的基本要素

如果团队的进度计划不符合实际情况或者目标盲目乐观，那么在疯狂努力以达到目标的过程中，他们的行为常常可能失去理性。这种现象通常会对项目造成危害，并有可能导致其彻底失败。