**PSP读书笔记**

杨枨老师在软件工程基础第一次课上就要求我们阅读PSP以及TSP，我阅读的第一本有关软件过程的书籍是《人月神话》，第二本是《PMBOK》

《人月神话》结合工业界具体的大型系统开发实例，对软件开发领域的诸多问题提出了思考和见解，对今天的大型软件开发都有启发意义。在第十六章《没有银弹》中提出由于软件的复杂性，一致性，变化性和不可见性，解决软件危机的银弹并不存在。没有任何技术或管理上的进展， 能够独立地许诺十年内使生产率、可靠性或简洁性获得数量级上的进步。

以下是我的PSP学习笔记：

1. **什么是PSP？**

个体软件过程是由美国Carnegie Mellon大学软件工程研究所(CMU/SEI)的Watts s. Humphrey领导开发的，于1995年推出，在软件工程界引起了极大的轰动，可以说是由定向软件工程走向定量软件工程的一个标志。PSP是一种可用于控制、管理和改进个人工作方式的自我改善过程，是一个包括软件开发表格、指南和规程的结构化框架。

个人软件过程(Personal Software Process，PSP)是一种可用于控制、管理和改进个人工作方式的自我持续改进过程，是一个包括软件开发表格、指南和规程的结构化框架。PSP与具体的技术（程序设计语言、工具或者设计方法）相对独立，其原则能够应用到几乎任何的软件工程任务之中。PSP能够说明个体软件过程的原则； 帮助软件工程师作出准确的计划；确定软件工程师为改善产品质量要采取的步骤；建立度量个体软件过程改善的基准；确定过程的改变对软件工程师能力的影响。

1. **PSP的主要内容？**
2. 说明个体软件过程的原则；
3. 帮助软件工程师作出准确的计划；
4. 确定软件工程师为改善产品质量要采取的步骤；
5. 建立度量个体软件过程改善的基准；
6. 确定过程的改变对软件工程师能力的影响。

1. **PSP的作用？**
2. 使用自底向上的方法来改进过程，向每个软件工程师表明过程改进的原则，使他们能够明白如何有效地生产出高质量的软件。
3. 为基于个体和小型群组软件过程的优化提供了具体而有效的途径。其研究与实践填补了CMM的空白。
4. 帮助软件工程师在个人的基础上运用过程的原则，借助于PSP提供的一些度量和分析工具，了解自己的技能水平，控制和管理自己的工作方式，使自己日常工作的评估、计划和预测更加准确、更加有效，进而改进个人的工作表现，提高个人的工作质量和产量，积极而有效地参与高级管理人员和过程人员推动的组织范围的软件工程过程改进。

**第一章 软件工程师的任务**

一个软件工程师的任务就是要在预定的时间和进度下交付高质量的软件产品。要进行有效的软件工程的工作，要从以下三个方面进行：开发出高质量的软件产品；在预期的费用内进行工作；在预定的进度下完成任务。

   在软件行业，没有经过PSP培训的工程师不得不在工作中掌握这种技能。这不仅要付出很大的代价和时间，而且具有越来越大的风险。软件工程师应该计划要做的工作，然后按照这个计划来工作。这就能够在规定的预算和时间内开发出高质量的产品。个体软件过程（PSP）就是为使工程师更好地工作而设计的一个框架。它指出如何估计和计划工作，如何按照这些计划来跟踪自己的性能，以及如何提高程序的质量。

**第二章 时间管理**

时间管理的逻辑原理是：为了制订切实可行的计划,必须对所用的时间进行跟踪；为了检查时间估计和计划的准确性，必须把它们写成文档并在今后与实际情况进行比较；为了制订出更准确的计划，需要知道以前的计划中存在哪些错误，哪些地方可以进行改进；为了管理好时间，首先制订时间分配计划，然后按照计划去做。

按照预先制订的计划进行工作会有两点好处：了解计划中还存在哪些错误，有助于更好地计划下一个项目；按照计划好的方式完成工作。这看起来好像不重要，但是事实上软件工程中许多错误都是由于考虑不周、精心大意或是不注意细节而造成的。

**第三章 时间跟踪第**

**四章 阶段计划与产品计划**

**第五章 产品计划**

计划是按期完成承诺的任务的可靠基础，而且计划也可以在工程师合作开发产品过程中协调他们的工作。个人的产品计划能使第个人互相约定各自的独立任务的期限，以便完全满足所承诺的各项任务。

**第六章 产品规模**

**第七章 管理好时间**

管理时间的步骤：决定如何使用时间；制订时间安排表；对照制订的安排表跟踪使用时间的方式；决定应该改变什么以使自己的行动达到所作安排的要求。

**第八章 契约的管理**

许多软件开发的进度和计划中存在的主要问题就是：管理人员要把这些进度和计划看成是类似于合同的契约，而软件工程师则并不把它看成是个人的契约。对于带有合同性质的契约，在它生效之前，两个或是更多的人必须在预期的行动上达成一致。

    一个真正的契约既是个人的承诺也有合同的效应，并且它需要双方或多方在如下问题上达成明确和自愿的一致。

**第九章 进度管理**

**第十章 项目计划**

**第十一章 软件开发过程**

    一个作业的每个步骤或阶段都有明确的入口准则，只有满足这些条件，才可以开始进入这个阶段。同样它们也有出口准则，只有满足这些条件，才可以结束这个阶段。过程的步骤定义要完成的任务以及如何完成这些任务，因为软件工程师的过程质量在很大程度上决定了他们的工作质量和效率，所以过程的设计和管理在软件工程中是非常重要的。

    个体软件过程（PSP）是一个用以帮助软件工程师测量和改进工作方式的框架。PSP有两个目标，一个是帮助软件工程师开发程序，另一个是告诉你如何利用过程来改善工作方式。

**第十二章 缺陷**

软件质量是要满足用户要求并且要能可靠而稳定地完成用户的工作。这就要求你开发的软件完全没有或几乎没有缺陷。软件缺陷是软件产品中的不正确的东西。缺陷是由人为的错误引起的。因为查找和修复缺陷的代价很大，所以，最有效的方法是工程师能及时发现和修复引入的缺陷。

**第十三章 缺陷查找技术**

在PSP中主要的缺陷排除方法是个人代码复查。先打印出程序清单，然后逐行进行复查，尽可能地修复发现的每一个缺陷。最有效的方法是，找出以前的程序中在编译和测试阶段引起最大问题的那些缺陷类型。因为人们往往容易犯同样的错误，所以缺陷数据有肋于在第一次编译和测试前找到全部或绝大部分程序的缺陷。经验表明，当工程师在第一次编译前仔细地复查了他们的代码时，能减少编译时间，大约相当于10%的开发时间，并且可以节省更多的测试时间。

**第十四章 代码复查检查表**

检查表包括一系列规程式的步骤，并要求你精确地遵循这些步骤去做。当人们需要严格地按照说明去做某件重要事情时，经常使用检查表。如果想发现和改正程序的每一个缺陷，就必须遵照一个精确的规程。检查表可以帮助确保遵循这个规程。检查表也是一个构思的来源。当按照检查表去做时，就知道如何进行代码复查。检查表包括了个人的经验。通过不断地使用和改进个人检查表，可以更好地发现在程序中的缺陷。

**第十五章 缺陷预测**

缺陷个数不稳定是有以下几个原因造成的。首先使经验问题，个人的技能是在不断提高的。开始编程序时，要面临着很多以前没有碰到过的问题。往往不能确定有些过程和函数是如何执行的，可能是语言的结构不清楚或者可能会遇到新的编译器或编程环境的问题。这些问题都会引起开发时间和缺陷路的波动。有了经验后，你将逐渐克服这些问题，犯的错误就减少了。这既减少缺陷的总数又减少缺陷数目的波动。

缺陷路波动的第二个原因是个体过程不稳定。当开始学习写程序时你也同时开始学习使用新的过程和方法。你的过程将随着实际的经验不断的发展，这就会引起完成不同程序任务的时间和引入缺陷的数据的波动。

**第十六章 缺陷排除的经济效益**

保证工作质量的原则是在第一次就要开发出合格的产品。

**第十七章 设计缺陷**

**第十八章 产品质量**

**第十九章 过程质量**

程序的质量取决于过程的质量，而过程的质量又取决于人们的工作方式。

**第二十章 个人对质量的承诺**

    软件应用在许多有迫切需要的领域，软件缺陷可能引起的经济损失，甚至造成人身伤害。当你的软件是飞机飞行、汽车驾驶、空中交通管理、控制工厂运行或控制电厂的系统的一个部分时，软件系统中的缺陷可能造成无法计算甚至可能是灾难性的后果。每个领域的软件应用都遵循着同样的趋势。