第13章 软件项目管理

——SE2017-G01

1. 选择题
2. 代码行技术在已知最小规模（a），最大规模（b），最可能规模（m）这些值平均值后，估计值L等于\_\_\_\_。

A.L=(a+4m+b)/6 B.L=(a+4m+b)/4

C.L=(a+6m+b)/4 D.L=(a+6m+b)/6

1. 以下那个特性不是属于功能点技术中的信息域特性\_\_\_\_。

A.输入项数 B.外部接口数

C.输出项数 D.主文件数

1. 以下哪个估算模型不属于面向KLOC的估算模型\_\_\_\_。

A.Walston\_Felix B.Albrecht&Gaffney

C.Bailey\_Basili D.Doty

1. 以下关于Gantt图的说法错误的是\_\_\_\_\_
2. B.甘特图能够显式地描绘各项作业彼此之间的依赖关系

是历史悠久，应用广泛的制定进度计划的工具

C.进度计划的关键部分不明确

D.计划中有潜力的部分及潜力大小不明确

1. 关于最早时刻EET以下描述错误的是\_\_\_\_。
2. 考虑进入该事件的所有作业

B.对于每个作业都计算它的持续时间与起始事件EET之和

C.选取其中的最小值作为该事件的EET

D.通常第一个事件的EET为0

1. 以下不属于典型组织方式的是\_\_\_\_。

A.民主制程序员组 B.主程序员组

C.多程序员组 D.现代程序员组

1. 下列关于软件质量错误的是\_\_\_\_。
2. 软件只要好用就是质量高
3. 软件与需求不一致就是质量不高

C. 没有满足隐含需求的软件质量值得怀疑

D.没有遵守准则会导致软件质量不高

1. 以下不属于软件质量因素的是\_\_\_\_。

A.可维护性 B.可查询性

C.可移植性 D.健壮性

1. 以下不属于审查过程的是\_\_\_\_。

A.综述 B.讨论

C.返工 D.跟踪

1. 正式技术复审在发现规格说明错误和设计错误方面的有效性为\_\_\_。

A.99% B.25%

C.75% D.50%

1. 以下不属于软件配置管理过程的是\_\_\_\_。

A.版本控制 B.状态报告

C.变化控制 D.标识软件配置中的变化

1. 状态报告不会对应以下哪个问题\_\_\_\_。

A.发生了什么事 B.谁做的这件事

C.这件事在哪里发生的 D.它将影响哪些其他事物

1. 能力成熟度有几个等级\_\_\_\_。

A.3 B.4

C.5 D.6

1. 软件质量保证（SQA）的主要措施不包括\_\_\_\_。

A.程序性能测试 B.基于非执行的测试

C.程序正确性证明 D基于执行的测试

1. 民主制程序员组内部假设有n个成员，可能的通信信道为\_\_\_\_。

A.n B.n\*(n+1)/2

C.n-1 D.n\*(n-1)/2

1. 项目中的小组成员要同时离开公司，项目经理首先应该做什么？
2. 实施风险计划
3. 招募新员工

C. 问老板要钱

D. 修订WBS

1. 软件调试的目的是（）

A 发现软件中隐藏的错误  B 解决测试中发现的错误

C 尽量不发现错误以便早日提交软件  D 证明软件的正确性

1. 软件项目管理是（）一切活动的管理

A需求分析

B软件设计过程

C模块设计

D软件生命周期

1. 以下哪个模式取消主程序员的大部分行政管理工作\_\_\_\_。

A.民主制程序员组 B.主程序员组

C.现代程序员组 D.以上都不对

1. Boehm根据经验支出，软件开发时间最多可以减少到正常时间的\_\_\_\_。

A.50% B.90%

C.75% D.37%

1. 判断题
2. 把产品信息域的5个特性都分类为简单级、平均级或复杂级，根据等级为每个特性分配一个功能点数，然后进行加权运算，如下所示：

result=a1xInp+a2xOut+a3xInq+a4xMaf+a5Inf（a1-a5为信息域系数）

如此算出的result即功能点数FP。

（错）

1. COCOMO2模型给出了3个层次的软件开发工作量估算模型，这3个层次的模型在估算工作量时对软件细节考虑的详尽程度逐级增加。COCOMO2模型可以适用于所有类型的软件和开发环境的工作量估算。

（错）

1. 没有一个普遍适用于所有软件项目的任务集合。

（对）

1. 概括地说，软件质量就是“软件与明确地和隐含地定义的需求相一致的程度”。

（对）

1. 一个项目的人员组织最理想的的状态是：由经验多、技术好、能力强的程序员作为主程序员，同时安排若干人员负责处理主程序员繁琐的事务性工作，再设立一些与主程序员同样优秀的后备程序员随时准备在主程序员发生意外时接替工作。

（错）

1. 任何软件开发都是迭代过程。

（对）

1. 软件开发时间与从事开发工作的人数总是呈反比关系。

（错）

1. 软件配置管理主要有五项任务：标识、版本控制、变化控制、配置审计和报告。

（对）

1. 一个大项目可以分解为数以百计的小任务（也称为作业），管理人员应该高度关注每个任务的进展情况，一旦任何一个任务拖后，都会导致整个项目的完成日期拖后。

（错）

1. 正式技术复审在发现规格说明错误和设计错误方面非常有效。

（对）

1. 如果一个系统软件的开发时间过短，则开发成功的概率几乎为零。

（对）

1. 用代码行技术估算软件规模时，当程序较小时常用的单位是代码行数（LOC），当程序较大时常用的单位是千行代码数（KLOC）。

（对）

1. 常用的制定进度计划的工具有Gantt图和工程网络，工程网络要比Gantt图更好，我们应尽量使用工程网络来制定进度计划并监督项目进展状况。

（错）

1. 软件工程包括技术和管理两方面的内容，是技术与管理紧密结合的产物。

（对）

1. CMM是用于评价软件机构的软件过程能力成熟度的模型，被应用于许多软件机构内部的过程改进活动中。

（对）

1. 软件维护类似于软件配置管理，在软件项目启动时就开始，一只持续到软件退役后才终止。

（错）

1. 正式技术复审的显著优点是，能够较早发现软件错误，从而可防止错误被传播到软件过程的后续阶段。

（对）

1. 测试可以暴露程序中的错误，对于保证软件可靠性来说是一种完善的技术。

（错）

1. 工程网络是系统分析和系统设计的强有力的工具，它能描绘任务分解情况和每个任务的开始时间和结束时间，它的不足是没法显式描绘各任务间的依赖关系。

（错）

1. 软件项目管理先于任何技术活动之前开始，并且贯穿于软件的整个生命周期之中。

（对）

1. 简答题
2. 动态多变量模型中的P（生产率参数）反映了哪些因素对工作量的影响？

总体过程成熟度及管理水平

使用良好的软件工程实践程度

使用的程序设计语言的级别

软件环境的状态

软件项目组的技术及经验

应用系统的复杂程度

1. 甘特图的三个主要缺点：

不能显式的描绘各项作业彼此间的依赖关系；

进度计划的关键部分不明确，难于判断哪些部分应当是主攻和主控对象

计划中有潜力的部分及潜力的大小不明确，往往造成潜力的浪费

1. 人如果希望缩短工期，应该怎么办？

往关键作业中增加资源才会有效果

1. 不在关键路径的作业有一定的机动时间，机动时间计算法方式是：

机动时间=（LET）结束—（EET）开始—持续时间

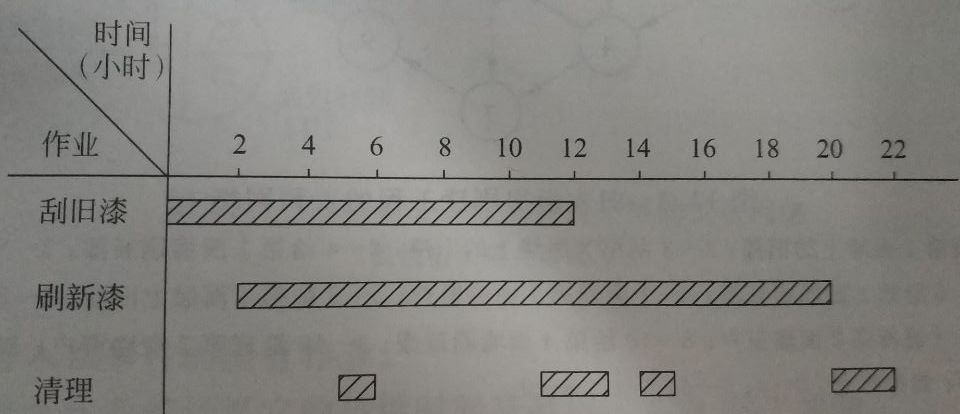
1. 软件质量保证措施中，技术复审的必要性：

能够较早的发现软件的错误，，从而防止错误被传播到软件过程的后续阶段

1. 走查的两种主要方式及特点：
2. 参与者驱动法：参与者按照事先准备好的列表，提出他们不理解的术语和认为不正确的术语。
3. 文档驱动法：文档编写写着向走查组成员仔细解释文档
4. 软件配置管理是在软件的整个生命期内管理变化的一组活动，具体地说，这组活动用来：
5. 标识变化
6. 控制变化
7. 确保适当地实现变化
8. 向需要知道这类信息的人报告变化
9. 能力成熟度的五个等级：
10. 初始级
11. 可重复级
12. 已定义级
13. 已管理级
14. 优化级
15. 论述题
16. 假设有一座陈旧的矩形木板房需要重新油漆，这项工作必须分3步完成，首先刮掉旧漆，然后刷上新漆，最后清除溅在窗户上的油漆。假设一共分配了15名工人去完成这项工作，然而工具有限，只有5把刮旧漆用的刮板，5把刷新漆用的刷子，5把清除溅在窗户上的油漆用的小刮刀。这个矩形木板房总共有4面墙壁，其中第2、4两面墙要比第1、3两面墙的长度长一倍。下表列出了每道工序需要消耗的时间。如何安排才能让工作更有效？请画出该刷漆工程的Gantt图，并写出最短消耗工时。

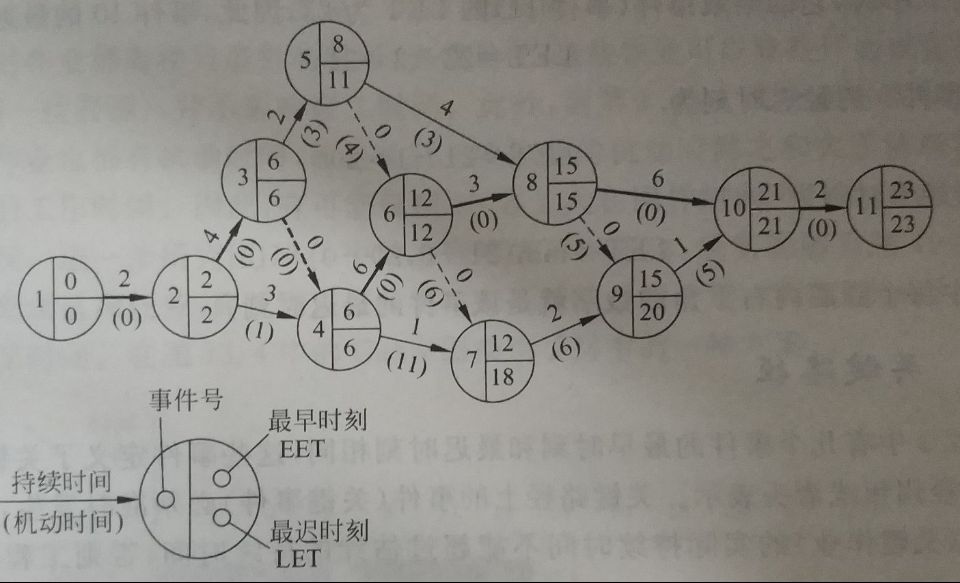
|  |  |  |  |
| --- | --- | --- | --- |
| 墙壁 | 刮旧漆 | 刷新漆 | 清理 |
| 1、3墙 | 2h | 3h | 1h |
| 2、4墙 | 4h | 6h | 2h |

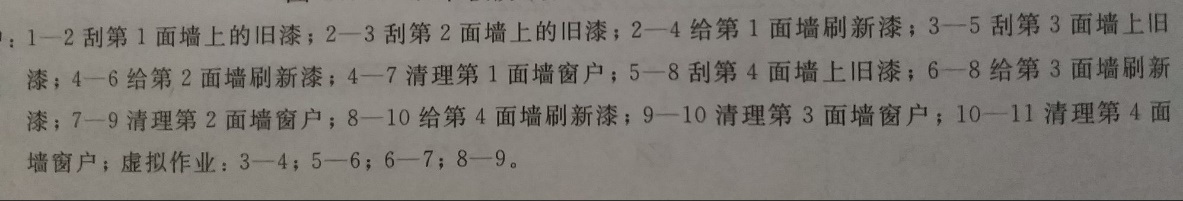
答案：



最短消耗22工时。

1. 请画出上一题中项目的工程网络，标出持续时间、最早时刻、最迟时刻。并写出该工程网络的关键路径。





关键路径：1-2-3-4-6-8-10-11

1. 请论述软件质量的要点，如何判断一个软件的质量高低。

答案：

1. 软件需求是度量软件质量的基础，与需求不一致就是质量不高。
2. 指定的开发标准定义了一组指导软件开发的准则，如果没有遵守这些准则，肯定导致软件质量不高。
3. 如果软件满足明确描述的需求，却不满足隐含地需求，那么软件的质量仍然是值得怀疑的。
4. 软件项目组有哪3种典型的组织方式？试分析每个方式的优劣处。

答：

1. 民主制程序员组

优点：

* 1. 组员们对发现程序错误持积极的态度，这种态度有助于更快速的发现错误，容易产生高质量的代码。
  2. 组员们享有充分民主，小组有高度凝聚力，组内学术空气浓厚，有利于攻克技术难关。

缺点：

1. 如果组内成员多数技术水平不高，或是缺乏经验的新手，这种组织方式将缺乏权威指导开发工程的进行，组员间缺乏必要的协调，最终可能导致工程失败。
2. 主程序员组

优点：

* 1. 专业化，每名成员仅完成他们受过专业训练的那些工作。
  2. 层次性，主程序员指挥每名组员工作，并对项目全面负责。

缺点：

1. 不切实际，除了主程序员外的人员难以找到。
2. 小组成员为了业绩不愿意发现错误。
3. 现代程序员组

优点：

* 1. 将民主制程序员组和主程序员组的优点相结合。
  2. 弥补了民主制程序员组和主程序员组的缺点。

1. 请论述功能点技术中有哪五大信息域，给出功能点的详细估算步骤及计算过程（不需要列举出14个技术因素）。

参考公式：

技术复杂性因子计算公式 ：TCF=0.65+0.01XDI，其余公式需在答案中写出。

英文参考：

UFP：未调整功能点数，

TCF：技术复杂性因子

DI：技术因素对软件规模的综合影响程度。

FP：功能点数

答案：五大信息域：输入项数Inp，输出项数Out，查询数Inq，主文件数Maf，外部接口数Inf（英文简写可不加）

计算过程：

* + - 1. 计算未调整的功能点数UFP：把产品信息域的5个特性都分类为简单级、平均级或复杂级，根据等级为每个特性分配一个功能点数，然后进行加权运算，如下所示：

UFP=a1xInp+a2xOut+a3xInq+a4xMaf+a5Inf（a1-a5为信息域系数）

* + - 1. 计算技术复杂性因子TCF：度量14种技术因素对规模的影响程度，为每个因素分配一个0-5的影响度，最后求和得到综合影响程度DI，再由题目给出的公式得到技术复杂性因子TCF