**选择**

以下哪一项不属于软件维护的特点 （C）

A.结构化维护与非结构化维护差别巨大

B.维护问题很多

C.维护持续周期长

D.维护代价高昂

2.以下哪一项维护活动占全部维护活动的比例最高 （A）

A.完善性维护

B.改正性维护

C.适应性维护

D.其他维护活动

3.以下哪一项不属于软件再工程过程 （B）

A.文档重构

B.结构重构

C.数据重构

D.代码重构

4.M = P + K \*exp(c-d)中c的含义是 （B）

A.维护人员对软件的熟悉程度

B.复杂程度

C.生产性工作量

D.经验常数

5.用户文档不包括 （C）

A.功能描述

B.操作员指南

C.代码规范

D.使用手册

6.“把今天的方法学应用到昨天的系统上，以支持明天的需求”指的是 （D）

A.可维护性复审

B.软件维护

C.软件测试

D.预防性维护方法

7.以下哪项不影响软件的可修改性 （A）

A. 代码可读性

B. 耦合

C. 内聚

D. 控制域与作用域的关系

8.这项活动不仅从现有程序中恢复设计信息，而且使用该信息去改变或重构现有系统，以提高其整体质量。这项活动描述的是 （C）

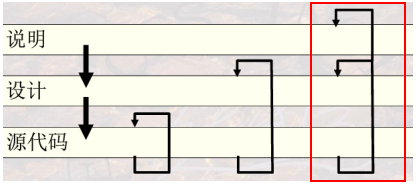
A.文档重构

B.逆向工程

C.正向工程

D.代码重构

下图圈出部分描述的是软件在工程中的哪个活动 （B）



A.文档重构

B.逆向工程

C.正向工程

D.代码重构

10.决定软件可维护性的因素不包括：（D）

A. 可理解性

B. 可测试性

C. 可修改性

D. 可移动性

11.用户文档主要描述（A）

A.系统功能和使用方法

B.系统功能和系统实现

C.系统设计和使用方法

D.系统设计和系统实现

12.以下哪一项不能用来度量维护工作（C）

A. 每次程序运行平均失效的次数

B. 维护每种语言平均花费的人时数

C. 用于维护活动的总人数

D. 不同维护类型所占的百分比

13.以下不属于系统文档描述内容的是（D）

A.系统设计

B.系统实现

C.系统测试

D.系统维护

14.以下说法不正确的是（C）

A.维护一行源代码的代价可能是最初开发该行源代码代价的14~40倍。

B.在完成预防性维护的过程中可以建立起完整的软件配置。

C.利用逆向工程和再工程的工具，可以使所有工作自动化。

D.由于现有的程序版本可作为软件原型使用，重新开发该程序的开发生产率可大大高于平均水平。

15.哪个数据是在保存维护记录中不值得被记录的（D）

A. 程序标识

B. 源语句数

C. 机器指令条数

D. 程序测试时间

16.以下对软件修改报告内容描述有误的一项是（B）

A. 满足维护要求表中提出的要求所需要的工作量

B. 维护要求的人员分工

C. 这项要求的先后次序

D. 与修改有关的事后数据

17.一下那一项不是软件维护任务完成之后的复查所要解决的问题（C）

A. 在当前处境下设计、编码或测试的哪些方面能用不同方法进行

B. 哪些维护资源是应该有而事实上却没有的

C. 对于这项维护工作有什么想法

D. 要求的维护类型中有预防性维护吗

18.M = P + K \*exp(c-d)中P的含义是 （C）

A.维护人员对软件的熟悉程度

B.复杂程度

C.生产性工作量

D.经验常数

19.M = P + K \*exp(c-d)中K的含义是 （D）

A.维护人员对软件的熟悉程度

B.复杂程度

C.生产性工作量

D.经验常数

20.M = P + K \*exp(c-d)中d的含义是 （A）

A.维护人员对软件的熟悉程度

B.复杂程度

C.生产性工作量

D.经验常数

**判断**

1.软件运行维护阶段是软件生命周期的最后一个阶段 （√）

2.软件运行维护阶段的基本任务是保证软件在一个相当长的时期能够正常运行 （√）

3.所谓软件维护就是在软件已经交付使用之后，为了改正错误或满足新的需求而修改软件的过程 （√）

4.软件工程的主要目的就是提高软件的可维护性，减少软件维护所需要的工作量，降低软件系统的总成本。 （√）

5.维护软件的可执行代码比维护软件文档重要。 （×）

6.非结构化维护需要付出很大代价(浪费精力并且遭受挫折的打击)，这种维护方式是没有使用良好定义的方法学开发出来的软件的必然结果。 （√）

7.在过去的几十年中，软件维护的费用稳步上升。 （√）

8.维护的代价高昂仅仅体现在维护费用。 （√）

9.文档是影响软件可维护性的决定因素。 （√）

10.再工程过程一定会在完成特定活动之后终止。 （×）

11.预防性维护实质上是软件再工程。 （√）

12.如果软件的开发途径不好（即没有使用软件工程方法学），而且原来的开发人员不能参加维护工作，那么维护工作量和费用将指数地增长。 （√）

完善性维护，是为了和变化了的环境适当地配合而进行的修改软件的活动。 （×）

14.所谓系统文档，指从问题定义、需求说明到验收测试计划这样一系列和系统实现相关的文档。 （√）

15.维护要求表是一个外部产生的文件，它是计划维护活动的基础。 （√）

16.每一次在拟定进一步的维护计划之前，都需要把软件修改报告提交给变化授权人审查批准。 （√）

17.对于程序模块来说，可以用程序复杂度来度量它的可测试性。 （√）

18.代码复审应该强调编码风格和内容说明文档这两个影响可维护性的因素。 （√）

19.配置复审的目的是保证软件配置的所有成分是完整的、一致的和可理解的，而且为了便于修改和管理已经编目归档了。 （√）

20.软件的逆向工程是分析程序以便在比源代码更高的抽象层次上创建出程序的某种表示的过程，是一个恢复设计结果的过程。 （√）

**简答**

1. 用于维护工作的劳动可分为那两种？简要说明其区别。

用于维护工作的劳动可以分成生产性活动和非生产性活动。

生产性活动是指分析评价、修改设计和编写程序代码等活动。

非生产性活动是指理解程序代码的功能，解释数据结构、接口特点和性能限度等活动

2. 决定软件的可维护性的因素存在哪五点？

软件可理解性、可测试性、可修改性、可移植性、可重用性

3. 软件再工程包括哪几类活动？

库存目录分析、文档重构、逆向工程、代码重构、数据重构、正向工程

4. 请对软件的可维护性进行定义。

软件的可维护性可以定性地定义为：维护人员理解、改正、改动或改进这个软件的难易程度。

5.请举例维护工作中的生产性活动具体有哪些（至少两例）

分析评价、修改设计、编写程序代码等

6.请举例维护工作中的非生产性活动具体有哪些（至少两例）

理解程序代码的功能、解释数据结构、接口特点、性能限度等

7.当改正性维护要求在估量错误阶段的结果是严重错误时，接下来的步骤是什么

在系统管理员的指导下分配人员，并且立即开始问题分析过程

8.当改正性维护要求在估量错误阶段的结果是不严重错误时，接下来的步骤是什么

改正性的维护和其他要求软件开发资源的任务一起统筹安排。

9.简述软件可理解性指的是什么

软件可理解性变现为外来读者理解软件的结构、功能、界河口和内部处理过程的难易程度。

10.简述软件可移植性指的是什么

软件可移植性是指把程序从一种计算环境（硬件配置和炒作系统）转移到另一种计算环境的难易程度。

**论述**

请简要指出软件的四个维护活动并论述其维护的原因目的。

第一项维护活动:改正性维护；改正性维护是为诊断和改正软件系统中潜藏的错误而进行的活动。

第二项维护活动:适应性维护；适应性维护是为适应环境的变化而修改软件的活动。

第三项维护活动:完善性维护；完善性维护是根据用户在使用过程中提出的一些建设性意见而进行的维护活动。

第四项维护活动:预防性维护；预防性维护是为了进一步改善软件系统的可维护性和可靠性，并为以后的改进奠定基础。

2.软件维护可能存在着哪些问题？（答出三点即可）

1.读懂原开发人员写的程序通常相当困难

2.软件人员的流动性，使得软件维护时，很难与原开发人员沟通

3.没有文档或文档严重不足

4.软件设计时，欠考虑软件的可修改性

5.频繁的软件升级，要追踪软件的演化变得很困难，使软件难以修改

3.对软件再工程进行定义。

采用先进的软件工程方法对整个软件或软件中的一部分重新进行设计、编写和测试，以提高软件的可维护性和可靠性，保证系统的正常运行。

4.软件系统文档中的用户文档应该包括哪些内容，并说明这些内容的作用

（1）功能描述，说明系统能做什么

（2）安装文档，说明怎样安装这个系统以及怎样使系统适应特定的硬件配置

（3）使用手册，简要说明如何着手使用这个系统（应该通过丰富的例子说明怎样使用常用的系统功能，还应该说明用户操作错误时怎样恢复和重新启动）

（4）参考手册，详尽描述用户可以使用的所有系统设施以及他们的使用方法，还应该解释系统可能产生的各种出错信息的含义（对参考手册最主要的要求是完整，因此通常使用形式化的描述技术）

（5）操作员指南（如果需要有系统操作员的话），说明操作员应该如何处理使用中出现的各种情况

5.对于一些“老”程序，当初开发过程中没有使用软件工程方法学，使得程序体系结构和数据结构都很差，文档不全，对曾经做过的修改也没有完整的记录。现在为了满足用户的新需求，而以软件工程学为指导进行软件再工程，你认为这样的做法是否合理，请说明理由。

合理。

（1）维护一行源代码的代价可能是最初开发该行源代码代价的14~40倍。

（2）重新设计软件体系结构（程序及数据结构）是使用了现代设计概念，它将对未来的维护可能有很大帮助。

（3）由于现有的程序版本可作为软件原型使用，重新开发该程序的开发生产率可大大高于平均水平。

（4）用户具有较多使用该软件的经验，因此能够很容易地搞清新的变更需求和变更的范围。

（5）利用逆向工程和再工程的工具，可以使所有工作自动化。

（6）在完成预防性维护的过程中可以建立起完整的软件配置。