

소프트웨어공학

문 1. CMMI에서 정의하는 목표는 조직이 달성해야 하는 바람직한 상태의 표현으로서, 각 프로세스 영역에 연관된 특정 목표를 가지며, CMMI는 이러한 목표를 달성하기 위한 방법을 제시하고 있다. 이 방법을 나타내는 CMMI 용어로 옳은 것은?

- ① 관례(practice) ② 성숙도 지침
- ③ 최적화 지침 ④ 목표(target) 프로파일

문 2. 소프트웨어 형상 관리의 활동으로 가장 적절하지 않은 것은?

- ① 형상 관리 항목의 식별 ② 버전(version) 제어
- ③ 형상 항목의 생성 ④ 형상 상태 보고 및 감사

문 3. UML 다이어그램의 설명이 옳지 않은 것은?

- ① 사용사례 다이어그램(use-case diagram) - 시스템의 기능을 모델링
- ② 상태 다이어그램(state diagram) - 클래스 사이의 메시지 교환을 시간의 흐름에 따라 표현
- ③ 클래스 다이어그램(class diagram) - 시스템의 정적인 구조를 나타냄
- ④ 액티비티 다이어그램(activity diagram) - 시스템의 동적 특징을 나타냄

문 4. 요구분석에 대한 설명으로 옳지 않은 것은?

- ① 각 요구사항은 명확하고, 구체적이고, 정확하고, 검증이 가능하도록 정의되고 기술되어야 한다.
- ② 요구사항은 고품질의 소프트웨어를 개발하고 검증할 수 있는 기초를 제공한다.
- ③ 고객과 개발자가 서로 당연한 것으로 인정하는 요구사항은 생략하여도 무방하다.
- ④ 요구사항은 크게 기능적인 요구사항과 성능, 신뢰성, 가용성, 보안성, 안전성 등의 비기능적 요구사항으로 분류된다.

문 5. 다음 중에서 COCOMO 모델에서 사용되는 노력승수값을 구하기 위해서 사용되는 요소가 아닌 것은?

- ① 제품의 특성 ② 컴퓨터의 특성
- ③ 개발 요원의 특성 ④ 사용자의 특성

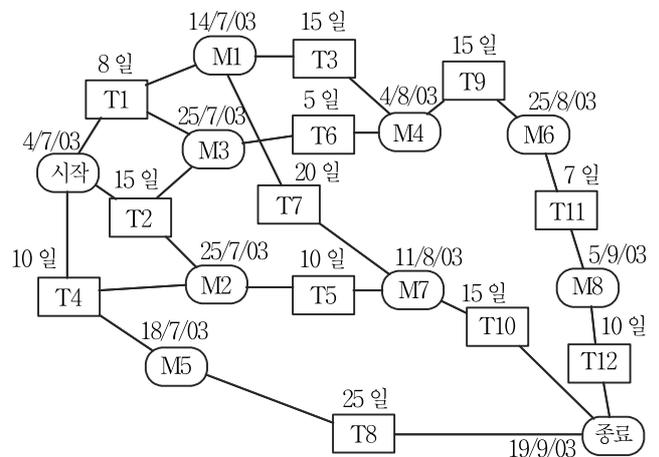
문 6. 소프트웨어에 대한 잘못된 통념에 해당되지 않는 것은?

- ① 소프트웨어는 유연하기 때문에 요구사항의 변경을 소프트웨어에 반영하는 것은 용이하다.
- ② 소프트웨어에 대한 소스 코드를 작성해서 실행을 시키기 전까지는 소프트웨어의 품질을 평가할 수는 없다.
- ③ 소프트웨어공학은 별로 필요없는 덩치 큰 문서를 생성하게 하여 개발을 매우 지연되게 만든다.
- ④ 지체된 프로젝트에 인력을 뒤늦게 투입하는 것은 오히려 프로젝트의 일정을 더욱 지연시킬 수가 있다.

문 7. 최근의 소프트웨어공학의 신기술과 관련된 설명으로 옳지 않은 것은?

- ① 컴포넌트기반 소프트웨어공학(CBSE)은 약하게 결합된 독립적인 컴포넌트를 정의, 구현, 합성하여 시스템을 구축하는 재사용기반의 접근방법이다.
- ② 보안 공학(security engineering)은 정보시스템이나 그 내부의 자료를 손상시키려는 악의적 위협(또는 공격)에 대해 방어할 수 있는 시스템을 개발하고 유지보수하는 활동이다.
- ③ 서비스지향아키텍처(SOA)에서 서비스 제공자는 서비스를 설계하고 구현한 후 이들 서비스를 SOAP을 이용하여 정의하고, 서비스 요청자는 서비스 명세와 서비스 제공자를 알기 위해 WSDL을 검색한다.
- ④ 애스펙트(aspect) 지향 소프트웨어 개발은 소위 ‘관심의 분리(separation of concerns)’ 원리를 이용하며, 애스펙트를 ‘cross-cutting concerns’으로 표현하여 이들 관심을 잘 이해하고 재사용하고 독립적으로 수정할 수 있다.

문 8. 프로젝트 일정계획을 위해 액티비티 네트워크가 사용된다. 다음의 액티비티 네트워크를 갖는 프로젝트의 경우 프로젝트가 종료되는데 소요되는 최소의 시간은 얼마인가?



- ① 35일 ② 40일
- ③ 52일 ④ 55일

문 9. 다음 설명에 해당되는 디자인 패턴은?

문제	(서브시스템 안에서와 같이) 이중의 구현이나 인터페이스들에 대해 공통적이고 통합된 인터페이스가 필요하다. 서브시스템 안의 많은 것들이 바람직하지 않은 결합도를 갖거나 서브시스템의 구현이 바뀔 수 있다. 어떻게 해결해야 하는가?
해결책	서브시스템에 대한 단일 접촉 지점을 정의한다. 즉, 외관인 객체는 서브시스템을 감싼다. 이 객체는 하나의 통합된 인터페이스를 제시하고 서브시스템 컴포넌트들과 협력할 책임을 갖는다.

- ① Adapter 패턴 ② Facade 패턴
- ③ Factory 패턴 ④ Singleton 패턴

문 10. 신속한 소프트웨어 개발(Rapid Software Development)에 관한 설명이 옳지 않은 것은?

- ① Agile 방법은 개발 오버헤드를 줄임으로써 소프트웨어를 신속히 생산하도록 하는 반복적 개발방법이다.
- ② 실행가능한, 쓰고 버리는(throw-away) 프로토타입의 장점은 프로토타입의 사용 양식이 최종적으로 인도되는 시스템의 사용 방식과 일치한다는 점이다.
- ③ RAD(Rapid Application Development) 환경은 데이터베이스 프로그래밍언어, 폼 생성 도구, 사무응용으로의 연결 기능을 포함한다.
- ④ XP(eXtreme Programming)는 시험 우선 개발, 리팩토링 및 고객 참여와 같은 사항을 포함한다.

문 11. 시스템 아키텍처에 대한 설명으로 옳지 않은 것은?

- ① 물리적 구성을 기반으로 정의되는 시스템의 상세 설계도이다.
- ② 이해당사자들과의 상호 이해, 협상, 동의 및 의사 교환을 위한 도구이다.
- ③ 프로젝트 초기의 설계 결정으로 시스템 개발 및 유지보수 전반에 걸쳐 지속적인 영향력을 갖는다.
- ④ 시스템에 관련 있는 이해당사자들의 요구사항을 고려하여 정의한다.

문 12. 소프트웨어 재사용을 지원하는 접근법과 거리가 먼 것은?

- ① 디자인 패턴(design pattern)
- ② 컴포넌트 기반 개발(component-based development)
- ③ 제품 라인(product line)
- ④ 클린룸(cleanroom) 방법

문 13. 화이트 박스(white box) 시험 기법에 해당하는 것은?

- ① 동등 분할(equivalence partitioning)
- ② 구조 시험(structure testing)
- ③ 경계값 분석(boundary value analysis)
- ④ 원인결과 그래프(cause-effect graph)

문 14. UML의 클래스 다이어그램에서 가장 표현하기 어려운 개념은?

- ① 일반화(generalization)
- ② 전체 부분(whole-part)
- ③ 분할(fork)과 결합(join)
- ④ 연관(association)

문 15. 객체지향개발방법론인 UP(Unified Process)에서 정제된 비전, 핵심 아키텍처의 반복적인 구현, 대부분의 요구사항 및 범위의 식별 등의 작업이 수행되는 단계는?

- ① construction 단계
- ② elaboration 단계
- ③ inception 단계
- ④ transition 단계

문 16. 정적 분석(static analysis)이란 ‘프로그램을 실행시키지 않고 프로그램 텍스트의 구문을 분석해서 결함과 이상을 발견하는 작업’이다. 정적 분석에 대한 설명이 옳지 않은 것은?

- ① 제어 흐름 분석 - 여러 개의 출구나 입구를 갖는 반복문과 도달할 수 없는 코드를 식별
- ② 데이터 사용 분석 - 초기화하지 않고 사용된 변수, 중간에 값 배정 없이 두 번 기록된 변수, 선언만 하고 사용되지 않는 변수를 식별
- ③ 인터페이스 분석 - 입력 변수와 출력 변수 사이의 종속성을 식별
- ④ 경로 분석 - 프로그램 상의 모든 가능한 경로를 식별하고, 그 경로에서 수행되는 문장을 나열

문 17. 시스템 개발단계 중 일반적으로 사용자 인수시험 계획이 준비되는 단계는?

- ① 요구사항 정의
- ② 타당성 조사
- ③ 구현계획 수립
- ④ 구현 후 검토

문 18. 소프트웨어 유지보수와 관련된 설명으로 옳은 것을 모두 고르면?

ㄱ. 역공학은 높은 추상도를 가진 표현에서 낮은 추상도 표현을 추출하는 작업이다.

ㄴ. 형상 관리는 프로그램 인도 후 이루어진다.

ㄷ. 일반적으로 소프트웨어 유지보수 비용은 소프트웨어 개발 비용의 25%를 차지한다.

ㄹ. 유지보수의 기술 향상을 위해 소프트웨어 척도를 사용한다.

ㅁ. 베이스라인의 설정은 형상 관리에서 일어나는 중요한 작업 중 하나이다.

- ① ㄱ, ㅁ
- ② ㄹ, ㅁ
- ③ ㄱ, ㄹ, ㅁ
- ④ ㄴ, ㄷ, ㄹ

문 19. 프로젝트 일정계획 수립에 사용되는 표기법별 용도에 대한 설명으로 옳지 않은 것은?

- ① 바 차트(bar chart) - 작업(activity)에 대한 책임자 및 작업의 시작과 종료 시점을 표현
- ② 액티비티 네트워크(activity network) - 작업(activity)들 간의 종속성(dependency)을 표현
- ③ 작업 분해 구조(work breakdown structure) - 프로젝트 완성에 필요한 소작업들의 계층적 목록을 나타냄
- ④ 플로우차트(flowchart) - 단위 업무(task) 데이터의 흐름을 나타냄

문 20. 객체지향 개념에서 캡슐화에 대한 설명으로 옳지 않은 것은?

- ① 캡슐화를 하면 객체들 사이의 결합도가 높아지고, 객체의 응집도가 향상된다.
- ② 객체의 명세와 실체를 분리하는 기법으로 캡슐화된 객체들의 재사용성을 향상시킬 수 있다.
- ③ 객체의 상세한 내용을 객체 외부로부터 숨기고, 단순히 메시지만으로 객체와의 상호작용을 하게 하는 것을 캡슐화라고 한다.
- ④ 인터페이스를 단순화시킬 수 있고, 프로그램 변경에 대한 오류의 파급효과가 적다.