소프트웨어공학

문 1. 나선형(spiral) 모형에서 단계별로 수행하는 작업 순서로 옳은 것은?

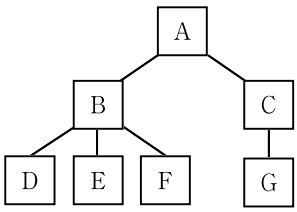
① 위험분석 – 계획 및 정의 – 개발 – 고객평가
② 계획 및 정의 – 위험분석 – 개발 – 고객평가
③ 계획 및 정의 – 개발 – 위험분석 – 고객평가
④ 위험분석 – 계획 및 정의 – 고객평가 – 개발

문 2. 다음 설명에 해당하는 용어는?

> 클래스나 컴포넌트가 제공하는 서비스들을 나타내는 오퍼레이션들의 집합을 나타낸다. 함수들의 시그너처(signature)만 명세할 뿐 함수 구현은 전혀 존재하지 않으며, 속성 값도 존재할 수 없다. 구현은 상속한 클래스에서 하며, 자체는 객체를 생성할 수 없다. 클래스와의 관계는 실체화 관계로 표현한다.

① 추상 클래스
② 액티브 클래스
③ 인터페이스
④ 상태머신

문 3. 다음 컴포넌트 계층 구조의 소프트웨어에 대해 하향식 통합시험을 수행하는 경우, 시험이 이루어지는 통합단위 순서로 옳은 것은?



① A – BC – DEFG – ABCDEFG
② A – BC – DEF – G
③ A – ABC – ABCDEFG
④ D – E – F – G – BDEF – CG – ABCDEFG

문 4. 다음 문제를 해결하는 유지보수 기법은?

> 시스템 사용 중 고객의 이름이 4자 이상 될 경우 (예: 신사임당) 인쇄 중 칸이 부족하여 잘릴 뿐 아니라, 프린터가 작동을 중지하고 원본 프로그램도 멈추는 중대한 문제점이 발생하였다. 4자 이상의 이름을 가진 고객이 많지 않으므로 이런 일은 항상 발생하지는 않지만, 한 달에 1번 정도 일어난다.

① 수정 유지보수(corrective maintenance)
② 적응 유지보수(adaptive maintenance)
③ 완전 유지보수(perfective maintenance)
④ 예방 유지보수(preventive maintenance)

문 5. 기본적인 비즈니스 요구사항, 시스템의 개요를 입력으로 하여 제안된 시스템이 요구 공학과 시스템 개발 프로세스를 수행할 만한 가치가 있는가를 판단하며, 시스템이 현재의 기술과 주어진 예산 및 일정 제한 조건 내에서 구현될 수 있는가 등에 대한 작업을 수행하는 요구 공학 프로세스의 단계로 옳은 것은?

① 타당성 조사(feasibility study)
② 요구사항 추출과 분석(requirements elicitation and analysis)
③ 요구사항 검증(requirements validation)
④ 요구사항 관리(requirements management)

문 6. 소프트웨어 형상관리(SCM)의 기본 요소에 대한 설명으로 옳지 않은 것은?

① 형상항목식별 – 소프트웨어 형상의 조직적 구조를 정의하고 수정이 용이하거나 변경이 발생할 때 추적이 쉽도록 하는 작업이다.
② 형상제어 – 식별된 형상항목의 변경요구를 검토, 승인하여 현재의 베이스라인이 적절히 반영될 수 있도록 하는 작업이다.
③ 형상감사 – 설계된 것이 만들어지고 테스트 한 후 형상항목에 대해 요구사항이 만족하는지 검증하는 작업이다.
④ 형상상태보고 – 모든 문서가 소프트웨어 내용과 정확히 맞는지를 검사하며 소프트웨어 제품 명세 검토, 각 동작에 대한 메모, 원시코드 등의 정보를 지원하는 작업이다.

문 7. UML의 순서(sequence) 다이어그램과 가장 관계가 먼 것은?

① 객체(object) ② 생명선(lifeline)
③ 구획면(swimlane) ④ 실행(activation)

문 8. 다음 설명과 가장 관련있는 객체지향 기법의 원칙은?

> 라인 그래프, 파이 차트, 히스토그램, 키비아트 다이어그램의 4가지 그래프를 그려야하는 응용 프로그램이 있다고 하자. 이 프로그램을 설계하기 위해 일반적인 클래스로 graph라는 클래스를 정의하고 각각의 그래프를 나타내는 클래스들을 graph 클래스의 서브클래스로 정의한다. 그리고 graph 클래스와 각각의 서브클래스에 그래프를 그리는 draw라는 메소드를 정의한다. 그러면 객체는 이들 서브클래스들의 인스턴스인 객체들 어떤 것에도 draw라는 메시지를 보낼 수 있고 메시지를 받은 객체는 적절한 그래프를 생성하기 위해 자기 자신의 draw 메소드를 호출한다.

① 캡슐화(encapsulation)
② 추상화(abstraction)
③ 다형성(polymorphism)
④ 정보은닉(information hiding)

문 9. 객체지향 분석 단계에서 클래스 모델링을 위해 사용하는 CRC 카드 활용에 대한 설명으로 옳지 않은 것은?

① 클래스의 기능성(responsibility)을 카드에 기재한다.
② CRC 기법의 특징은 클래스의 책임을 가지는 팀 멤버들에게 해당 카드를 배포하는 것이다.
③ CRC 카드를 팀이 활용할 경우, 멤버들 간의 상호작용으로 속성이나 메소드 등이 클래스에서 누락되었는지를 밝혀준다.
④ CRC 카드 활용법은 팀 멤버들이 관련 업무영역에 충분한 경험을 갖고 있지 않더라도 적절한 것이 장점이다.

문 10. 다음과 같은 프로그램이 있을 때 이 프로그램을 화이트 박스 테스트하려 한다. 테스트 종료 조건으로 경로 커버리지(path coverage)를 사용한다고 할 때, 테스트 데이터의 최소 개수는?

```
if x > 3 then
    x = x + 2
    if x > 9 then
        x = x * 2
    else
        x = x + 5
    end
else
    x = x + 4
end

if x > 8 then
    x = x + 7
else
    x = x * 3
end
```

① 3　　② 4
③ 5　　④ 6

문 11. 객체지향 개발 프로세스인 RUP(Rational Unified Process)에 대한 설명으로 옳지 않은 것은?
① 반복적 개발 프로세스이다.
② 비즈니스 관점보다는 기술적인 관점에 초점을 맞춘다.
③ 고객의 요구사항을 명확하게 문서화하고 그 요구사항의 변경을 추적한다.
④ 구축(construction)단계에서도 요구사항을 기술한다.

문 12. ISO 12207의 '기본 생명주기 프로세스(primary life cycle process)'에 포함되지 않는 것은?
① 획득 프로세스(acquisition process)
② 운영 프로세스(operation process)
③ 형상관리 프로세스(configuration management process)
④ 유지보수 프로세스(maintenance process)

문 13. 소프트웨어 아키텍처 모델 개발의 중요성과 가장 관련이 적은 것은?
① 시스템 이해
② 확장 준비
③ 단위테스트
④ 재사용성

문 14. CMM과 CMMI의 차이에 대한 설명으로 옳은 것은?
① CMM은 소프트웨어 개발 프로세스의 성숙도를 다루고 CMMI는 하드웨어, 네트워크 등 CMM에서 제외된 부분의 성숙도를 다룬다.
② CMM은 아날로그 타입이고 CMMI는 디지털 타입의 성숙도를 다룬다.
③ CMM과 CMMI는 같은 내용을 다루고 단지 CMM의 최신 버전을 CMMI라고 한다.
④ CMM은 소프트웨어 개발 프로세스의 성숙도를 다루고 CMMI는 소프트웨어, 시스템, 프로덕트를 포함하는 세 분야를 통합 평가하는 모델이다.

문 15. 소프트웨어 개발기간 동안 발생하는 이벤트와 위험(risk)을 구분하기 위해 살펴보아야 할 내용으로 가장 관련이 적은 것은?
① 이벤트를 발생시킨 주체
② 이벤트와 관련된 손실
③ 이벤트가 발생할 가능성
④ 결과를 변경시킬 수 있는 정도

문 16. 소프트웨어 프로젝트의 소요기간과 비용추정에 대한 설명으로 옳지 않은 것은?
① COCOMO는 먼저 완성될 시스템의 규모를 추정하고 이를 준비된 식에 대입하여 소요인원·월(Man - Month)을 예측한다.
② 기능점수방법은 원시코드의 구현에 이용되는 프로그래밍 언어에 독립적이다.
③ 기능점수방법은 경험 중심적이기 때문에 비용을 산정하려면 단위 시간당 프로그래머의 생산성을 기능점수로 표현한 생산성 메트릭이 있어야 한다.
④ LOC 기반 추정과 기능점수 기반 추정은 사용 정보만 다를 뿐 동일한 결과를 내므로 선택적으로 사용이 가능하다.

문 17. 소프트웨어 재공학(reengineering) 프로세스 모델에 포함되지 않는 것은?
① 커스터마이징(customizing)
② 재고 분석(inventory analysis)
③ 코드 재구축(code restructuring)
④ 순방향 엔지니어링(forward engineering)

문 18. 요구분석 명세서의 평가기준에 대한 설명으로 옳은 것은?
① 무결성 – 요구분석은 모호한 점이 없도록 간결하고 명쾌하게 작성해야 한다.
② 일관성 – 요구분석서 안에 서로 모순되는 부분이 없어야 한다.
③ 명확성 – 사용자의 요구를 오류없이 완벽하게 반영하고 있어야 한다.
④ 추적가능성 – 개발된 시스템이 요구분석에 기술된 내용과 일치해야 한다.

문 19. 소프트웨어 설계에 대한 설명으로 옳지 않은 것은?
① 각 모듈은 단순한 인터페이스를 갖도록 한다.
② 모듈의 팬 입력(fan-in)은 다른 모듈을 직접 제어하는 개수로서 한 모듈에서 호출하는 모듈의 개수이다.
③ 팬 출력(fan-out)의 수는 7±2 정도가 적절하다.
④ 정보은닉(information hiding) 개념에 따라 모듈을 설계한다.

문 20. 바람직한 프로그램 코드에 대한 설명으로 옳지 않은 것은?
① 다양한 오류 처리를 위한 간결한 구조를 위해서 예외처리를 효과적으로 사용하는 것이 필요하다.
② 소스코드 작성단계에서 발생한 설계변경은 설계문서에 반영할 필요는 없다.
③ 임시 변수의 사용은 피하는 것이 좋다.
④ 빠른 프로그램보다 먼저 명료한 프로그램을 작성하는 것이 좋다.