

1. 테스트의 목표를 수립하고, 테스트의 범위를 선정하는 활동으로 가장 옳은 것은?

- ① 테스트 계획
- ② 테스트 설계
- ③ 테스트 실행
- ④ 테스트 평가

2. 소프트웨어 아키텍처에 대한 설명으로 옳은 것을 <보기>에서 모두 고른 것은?

<보기>

- ㄱ. 품질 속성을 포함한 비기능적인 요구사항이 주요한 아키텍처 설계 목표가 된다.
- ㄴ. 서브시스템으로 분할할 때 서브시스템 간의 결합도는 낮추고 서브시스템 내의 응집도는 높이는 방향으로 설계가 되어야 한다.
- ㄷ. RUP(Rational Unified Process)는 주요 위험 요인들을 파악하고 시스템의 초기 골격 아키텍처를 먼저 구현하여 이를 바탕으로 반복적인 개발을 통해 시스템을 완성해 간다.
- ㄹ. ATAM은 시나리오를 기반으로 하여 아키텍처를 평가하는 방법 중 하나이다.

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

3. <보기>에서 기술한 활동과 관련 있는 국제 표준으로 가장 옳은 것은?

<보기>

소프트웨어를 개발할 때 변경 관리가 필요한 항목을 선정하고 그 항목은 아무나 변경할 수 없도록 적절한 통제를 한다. 변경할 내용을 기록하고 그 기록과 실제 변경된 내용이 일치하는지 확인하는 활동이다.

- ① ISO/IEC/ISO 29119
- ② IEEE 829
- ③ IEEE 1042
- ④ ISO/IEC 9899

4. 국제표준 ISO/IEC 9126과 비교하여 국제표준 ISO/IEC 25010에 추가된 소프트웨어 품질특성 항목으로 가장 옳은 것은?

- ① 사용성(usability)
- ② 호환성(compatibility)
- ③ 유지보수성(maintainability)
- ④ 이식성(portability)

5. 유사한 클래스들 사이의 공유되는 속성과 동작을 묶어주는 추상화 기법으로 가장 옳은 것은?

- ① 집단화(aggregation)
- ② 추상화(abstraction)
- ③ 캡슐화(encapsulation)
- ④ 일반화(generalization)

6. 소프트웨어 개발 계획 단계에서 도출해야 할 중요한 2가지 요소로 가장 옳은 것은?

- ① 비용과 일정
- ② 시간과 일정
- ③ 자원과 도구
- ④ 비용과 자원

7. 객체지향 소프트웨어 개발 프로세스 UP(Unified Process)의 특징으로 가장 옳지 않은 것은?

- ① 반복적(iterative)이고 점진적(incremental)이다.
- ② 유스케이스기반(use-case driven)으로 한다.
- ③ 아키텍처 중심(architecture centric)이다.
- ④ E-R 다이어그램 기반으로 되어 있다.

8. 역공학(reverse engineering)에 대한 설명으로 가장 옳지 않은 것은?

- ① 원시코드를 이용하여 원시코드 이상의 추상화된 표현으로 나타내는 것이며 코드를 분석하는 과정이다.
- ② 수작업으로 하는 것이 어렵기 때문에 자동화 도구를 사용한다.
- ③ 시스템의 성능, 효율성 개선을 목표로 한다.
- ④ 원시코드로부터 시스템 명세서를 개발하는 과정이다.

9. ISO/IEC 9126 품질 모델 중 소프트웨어를 쉽게 변경할 수 있는 특성으로 가장 옳은 것은?

- ① 유지보수성(maintainability)
- ② 재사용성(reusability)
- ③ 신뢰성(reliability)
- ④ 사용 용이성(usability)

10. <보기>와 같이 프로그램을 변경하는 작업으로 가장 옳은 것은?

<보기>

변경 전
<pre>public class Painter { public double calculatePaintRequired(double row, double col, double paintPerUnit) { double area = row * col; double paintRequired = area/paintPerUnit; return paintRequired; } }</pre>
변경 후
<pre>public class Painter { public double calculatePaintRequired(double row, double col, double paintPerUnit) { double area = calculateArea(row, col); double paintRequired = area/paintPerUnit; return paintRequired; } private double calculateArea(double row, double col) { double area = row * col; return area; } }</pre>

- ① 재사용(reuse)
- ② 재공학(reengineering)
- ③ 역공학(reverse engineering)
- ④ 리팩토링(refactoring)

11. <보기>의 설명을 읽고 UML 유스케이스 다이어그램을 작성할 때, A~E에 각각 들어갈 용어로 가장 옳은 것은?

<보기>

설명: 회원이 아닌 고객(비회원)은 물품 조회 만을 할 수 있다. 반면에 회원인 고객은 물품 구매도 가능하다. 그러나 물품 구매를 할 때는 반드시 물품 조회를 실행하여야 한다.

유스케이스 다이어그램:

- | | A | B | C | D | E |
|---|-----|-----|----|-------|-------|
| ① | 고객 | 비회원 | 회원 | 물품 조회 | 물품 구매 |
| ② | 고객 | 비회원 | 회원 | 물품 구매 | 물품 조회 |
| ③ | 비회원 | 회원 | 고객 | 물품 조회 | 물품 구매 |
| ④ | 비회원 | 회원 | 고객 | 물품 구매 | 물품 조회 |

12. PMBOK 프로젝트 관리 지침에서 제시하는 프로젝트 관리 관점 중에서 프로젝트의 요구사항을 만족하여 수행 목표를 달성하기 위한 활동을 다루는 것은?

- | | |
|---------|---------|
| ① 일정 관리 | ② 품질 관리 |
| ③ 비용 관리 | ④ 위험 관리 |

13. 다음은 '항공권 예약 시스템'의 요구 사항을 나열한 것이다. 기능적 요구 사항에 대한 설명으로 가장 옳지 않은 것은?

- ① 항공편, 탑승객, 예약을 입력하는 방법을 결정해야 한다.
- ② 티켓과 리포트에 어떤 정보를 표시할지 결정해야 한다.
- ③ 여행사와 고객이 데이터베이스에 접근할 때 어떤 정보를 얻을 수 있는지 결정해야 한다.
- ④ 자주 탑승하는 고객을 서비스하기 위해 시스템을 확장할 수 있도록 설계해야 한다.

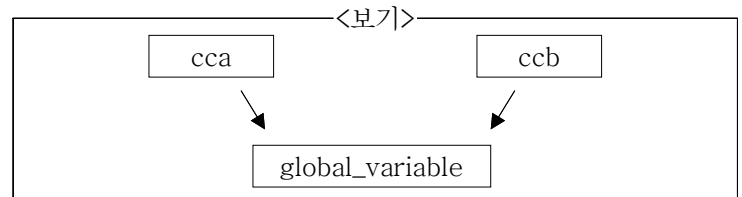
14. 클래스 다이어그램의 특징으로 가장 옳지 않은 것은?

- ① 클래스들의 인터페이스를 나타낸다.
- ② 클래스는 객체의 집합이 가지는 속성과 동작을 추상적으로 나타낸다.
- ③ 시스템 동적 구조를 모델링한다.
- ④ 속성값의 타입을 정하는 것은 시스템 정의에서 중요한 일이 아니므로 객체 설계 단계까지 미뤄도 된다.

15. 객체지향 분석에서 기능 모델링에 대한 설명으로 가장 옳지 않은 것은?

- ① 객체 모델링, 동적 모델링에 이어 시스템을 기술하는 세 번째 단계이다.
- ② 시스템에 요구되는 객체들의 구조와 관계가 정립된다.
- ③ 동적 모델링에서 나타난 동작들이 어떠한 기능으로 이루어져 있는지 나타낸다.
- ④ UML 다이어그램 중 액티비티(activity) 다이어그램을 사용한다.

16. 두 모듈 간의 상호작용 정도를 나타내는 모듈 결합도에서 <보기>와 같이 두 모듈들이 같은 전역 데이터를 접근하는 결합도로 가장 옳은 것은?



- ① 데이터 결합도(data coupling)
- ② 내용 결합도(content coupling)
- ③ 공통 결합도(common coupling)
- ④ 제어 결합도(control coupling)

17. 소프트웨어 시험 기법에서 블랙 박스 시험 기법의 특징으로 가장 옳지 않은 것은?

- ① 동등 분할(equivalence partitioning)
- ② 조건 검사(condition coverage)
- ③ 경계값 분석(boundary value analysis)
- ④ 원인-결과 그래프(cause-effect graphing)

18. 2017년 발표된 웹 애플리케이션 보안 취약점 OWASP TOP 10에 포함되지 않은 것은?

- ① 크로스 사이트 스크립팅(XSS)
- ② 인젝션
- ③ 보안 설정 미흡
- ④ 안전한 암호화 저장

19. 발주자가 요청한 최종 결과물의 일부 기능을 사전에 점검해 볼 수 있는 소프트웨어 생명주기 모델로 가장 옳은 것은?

- ① 폭포수 모델(waterfall model)
- ② 나선형 모델(spiral model)
- ③ 프로토타이핑 모델(prototyping model)
- ④ V 모델(V model)

20. 소프트웨어 개발 생명주기 모델 중 나선형 모델(spiral model)의 단계를 순서대로 바르게 나열한 것은?

- ① 위험 분석 - 고객 평가 - 계획 및 정의 - 개발
- ② 위험 분석 - 계획 및 정의 - 개발 - 고객 평가
- ③ 계획 및 분석 - 위험 분석 - 고객 평가 - 개발
- ④ 계획 및 정의 - 위험 분석 - 개발 - 고객 평가