

# 소프트웨어공학(7급)

(과목코드 : 143)

2025년 군무원 채용시험

응시번호 :

성명 :

- |  |  |
|--|--|
| <p>1. 소프트웨어 회귀 테스트에 대한 설명으로 가장 적절한 것은?</p> <ul style="list-style-type: none"><li>① 프로그램의 정확성뿐만 아니라 결과물의 품질을 지속적으로 기록하기에 유용하다.</li><li>② 소프트웨어의 버전이 올라갈수록 회귀 테스트를 수행하기 용이해진다.</li><li>③ 회귀 테스트 방식 중 기능 테스트는 개별 함수, 서브루틴, 메소드 등을 동작시켜 보는 활동이다.</li><li>④ 시스템이 과부하 상태에서 어떻게 작동하는지를 검사하는 테스트 방식이다.</li></ul> <p>2. 소프트웨어 제품 품질 특성을 정의하고 품질 평가의 척도를 정의한 국제 표준으로 가장 적절한 것은?</p> <ul style="list-style-type: none"><li>① ISO/IEC 25010</li><li>② ISO/IEC 15504</li><li>③ ISO/IEC 12207</li><li>④ ISO 9001</li></ul> <p>3. PMBOK(Project Management Body of Knowledge) 프로젝트 위험(Risk) 관리 지침에서 정성적 위험 분석 기법으로 가장 적절한 것은?</p> <ul style="list-style-type: none"><li>① 의사 결정 나무(Decision Tree)</li><li>② 금전적 기댓값 분석(Expected Monetary Value Analysis)</li><li>③ 민감도 분석(Sensitivity Analysis)</li><li>④ 확률-영향 매트릭스(Probability and Impact Matrix)</li></ul> | <p>4. 좋은 설계가 되기 위한 조건으로 가장 적절하지 않은 것은?</p> <ul style="list-style-type: none"><li>① 요구분석 명세서의 내용을 설계서에 모두 포함해야 한다.</li><li>② 유지보수가 용이하도록 추적이 가능해야 한다.</li><li>③ 시스템 변경으로 인한 영향이 최소화되도록 국지적이어야 한다.</li><li>④ 설계서는 프로그래머에게 필요하므로 간략하게 작성해야 한다.</li></ul> <p>5. UML에서 시스템의 하드웨어와 소프트웨어의 물리적인 구조를 나타내는 다이어그램으로 가장 적절한 것은?</p> <ul style="list-style-type: none"><li>① 배치 다이어그램(Deployment Diagram)</li><li>② 상태 다이어그램(Statechart Diagram)</li><li>③ 객체 다이어그램(Object Diagram)</li><li>④ 패키지 다이어그램(Package Diagram)</li></ul> <p>6. 소프트웨어 유지보수에 대한 설명으로 가장 적절한 것은?</p> <ul style="list-style-type: none"><li>① 유지보수 비용은 개발 비용에 비해 예측하기 쉽다.</li><li>② 복잡도가 높을수록 소프트웨어의 유지보수 비용이 감소한다.</li><li>③ SMI(Software Maturity Index)가 1에 가까울수록 제품이 안정적이다.</li><li>④ 유지보수 작업은 코딩 중심의 작업으로 개발 단계별로 수행된다.</li></ul> |
|--|--|

7. 다음 모듈 간의 결합도로 가장 적절한 것은?

- 모듈 간의 인터페이스로 배열이나 레코드 등의 자료구조가 전달될 때의 결합도를 의미
- 자료구조 변화는 그것을 조회하는 모든 모듈 뿐만 아니라 변화되는 필드를 실제로 조회하지 않는 모듈에도 영향을 미침

- ① 자료 결합도(Data Coupling)
- ② 스탬프 결합도(Stamp Coupling)
- ③ 외부 결합도(External Coupling)
- ④ 내용 결합도(Content Coupling)

8. 소프트웨어 개발 도구 중 소스 코드의 버전 관리를 위한 도구로 가장 적절하지 않은 것은?

- ① CVS                      ② Git  
③ Mockito                ④ SVN

9. 소프트웨어 하향식 비용 산정 기법으로 가장 적절한 것은?

- ① 개발 단계별 인원수 기법
- ② 델파이 기법
- ③ LOC(Line of Code) 기법
- ④ 수학적 산정 기법

10. 예방 유지보수에 대한 설명으로 가장 적절한 것은?

- ① 소프트웨어가 새로운 환경이나 요구사항에 맞게 계속 작동하도록 하는 것을 목표로 한다.
- ② 사용자 보고서에 따라 버그를 수정하거나 테스트 중에 발견된 결함을 수정하는 경우에 해당한다.
- ③ 소프트웨어의 기능을 향상하거나 새로운 기능을 추가하여 사용자 요구를 충족시킬 수 있다.
- ④ 코드 리팩토링을 통해 소프트웨어 구조를 개선하거나 성능을 최적화하는 경우에 해당한다.

11. UML의 4+1 뷰 관점에서 각각에 해당하는 다이어그램으로 가장 적절한 것은?

- ① 논리적 관점: 클래스 다이어그램
- ② 프로세스 관점: 유스케이스 다이어그램
- ③ 구현 관점: 객체 다이어그램
- ④ 배치 관점: 컴포넌트 다이어그램

12. 다음 형상 관리의 구성요소로 가장 적절한 것은?

- 소프트웨어 개발 과정에서 변경 사항을 체계적으로 관리하고 승인된 변경 사항만 적용되도록 하는 활동
- 형상 항목에 대한 모든 변경 사항에 대하여 체계적인 방식으로 조정, 평가, 승인, 구현 및 문서화하는 것

- ① 형상식별(Configuration Identification)
- ② 형상통제(Configuration Control)
- ③ 형상감사(Configuration Audit)
- ④ 형상기록(Configuration Status Accounting)

13. CMMI 성숙도 단계 중 레벨 2 관리(managed) 단계에 대한 설명으로 가장 적절하지 않은 것은?

- ① 문서화된 프로세스와 계획이 존재한다.
- ② 작업 산출물들에 대하여 형상 관리가 실시된다.
- ③ 통계적 기법을 사용하여 프로세스를 관리한다.
- ④ 프로젝트가 일정과 예산 내에서 수행된다.

14. 컴포넌트 기반 소프트웨어 개발 방법론으로 가장 적절한 것은?

- ① 스크럼(SCRUM)
- ② 칸반(Kanban)
- ③ 익스트림 프로그래밍(Xtreme Programming)
- ④ 마르미(MaRMI)-RE

15. UML의 다이어그램에 대한 설명으로 가장 적절하지 않은 것은?

- ① 유스케이스 다이어그램은 시스템의 기능적인 측면을 모델링하는 데 적합하다.
- ② 클래스 다이어그램은 시스템의 구조적인 측면을 모델링하는 데 적합하다.
- ③ 시퀀스 다이어그램은 시스템의 동적인 측면을 모델링하는 데 적합하다.
- ④ 액티비티 다이어그램은 객체별로 워크플로우를 모델링하는 데 적합하다.

16. 기능점수(Function Point) 산정 방법에 대한 설명 중 가장 적절하지 않은 것은?

- ① 소프트웨어 규모를 측정하는 방법이다.
- ② 사용자 관점이 아닌 구현 관점의 요구 기능을 정량적으로 산정한다.
- ③ 측정의 일관성을 유지하기 위해 개발 기술, 개발 방법, 품질 수준 등을 고려하지 않는다.
- ④ 소프트웨어의 요구사항 복잡도를 측정한다.

17. 다음 설명하는 코드 악취(Code Smell)로 가장 적절한 것은?

객체의 핵심은 데이터와 이에 사용되는 프로세스를 통합하는 것에 있다. 하나의 메소드가 소속되지 않은 클래스에 더 많이 접근한다면 메소드의 작성이 잘못된 것이다. 잘못 작성된 메소드가 흔히 접근하는 대상은 데이터이기 때문에 자주 접근하는 데이터가 있는 클래스로 메소드가 배치되어야 한다.

- ① 변경의 발산(Divergent Change)
- ② 변경의 분산(Shotgun Surgery)
- ③ 지나친 관여(Inappropriate Intimacy)
- ④ 속성, 조작의 부적절한 관계(Feature Envy)

18. 화이트박스 테스트 기법으로 가장 적절한 것은?

- ① 동등 분할(Equivalence Partitioning)
- ② 경계값 분석(Boundary Value Analysis)
- ③ 조건 검사(Condition Coverage)
- ④ 상태 전이 다이어그램(State Transition Diagram)

19. 소프트웨어 요구분석 명세서 작성 시 주의사항으로 가장 적절하지 않은 것은?

- ① 개발자가 설계와 코딩에 효과적으로 사용할 수 있도록 작성한다.
- ② 개발할 소프트웨어의 주요 기능적 요구사항을 위주로 기술한다.
- ③ 사용자가 읽기 쉽고 이해할 수 있도록 작성한다.
- ④ 품질에 대한 우선순위를 명시한다.

20. 다음 소프트웨어 개발 프로세스 모델로 가장 적절한 것은?

- 독립적인 서비스를 제공하는 소프트웨어 모듈을 조합해 재사용함으로써 개발 생산성과 품질을 높이고 시스템 유지보수 비용을 절감할 수 있는 개발 방법론
- 재사용성을 높이기 위해서는 결합도를 낮추고 응집도를 높이는 설계 전략이 요구됨

- ① 폭포수 모델
- ② 프로토타이핑 모델
- ③ 컴포넌트 기반 개발 모델
- ④ 애자일 개발 모델

21. 소프트웨어 상위 설계로 가장 적절하지 않은 것은?

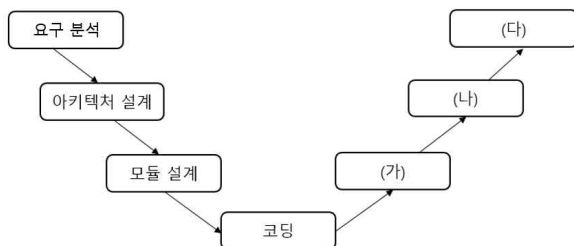
- ① 모듈 설계
- ② 데이터 설계
- ③ 사용자 인터페이스 설계
- ④ 아키텍처 설계

22. 다음 설명하는 디자인 패턴으로 가장 적절한 것은?

- 객체 간의 상호작용을 캡슐화하는 디자인 패턴으로 프로그램의 실행 중 행위를 바꾸는 방법인 행동 패턴(Behavior Pattern)에 속한다.
- M:N의 관계에서 M:1의 관계로 복잡도를 떨어뜨려 유지보수 및 확장성을 유리하게 하는 디자인 패턴이다.

- ① 브리지 패턴(Bridge Pattern)
- ② 어댑터 패턴(Adapter Pattern)
- ③ 데코레이터 패턴(Decorator Pattern)
- ④ 중재자 패턴(Mediator Pattern)

23. 다음 소프트웨어 개발 단계에 따른 V모델에서 (가), (나), (다)에 들어갈 용어들을 가장 적절하게 나열한 것은?

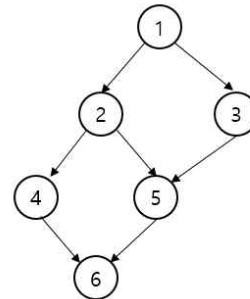


- ① (가) 단위 테스트 (나) 통합 테스트 (다) 시스템 테스트
- ② (가) 단위 테스트 (나) 시스템 테스트 (다) 통합 테스트
- ③ (가) 시스템 테스트 (나) 단위 테스트 (다) 통합 테스트
- ④ (가) 시스템 테스트 (나) 통합 테스트 (다) 단위 테스트

24. 서비스 지향 아키텍처(SOA)에 대한 설명으로 가장 적절한 것은?

- ① 웹서비스 개념 이후에 출현하였으며 웹서비스를 포괄한다.
- ② 사용자는 서비스 구현의 세부 정보를 알아야 한다.
- ③ 서비스는 느슨한 결합을 통해 서비스 간의 의존성을 최소화한다.
- ④ 서비스는 정적으로 바인딩되어 상호 운용성을 높이게 된다.

25. 다음 그래프에 대한 McCabe의 순환 복잡도로 가장 적절한 것은?



- ① 2
- ② 3
- ③ 4
- ④ 5