

ASSIGNMENT 7 FINAL REVIEW (FOR YOUR REFERENCE ONLY)

Due: 6/7(Friday), 6pm

Spring 2024

Name: _____

1. 두 명의 개발자가 5 개월에 걸쳐 10000 라인의 코드를 개발하였을 때, 월별(man-month) 생산성 측정을 위한 계산 방식으로 가장 적합한 것은? ()

- ① $10000/2$
- ② $10000/(5 \times 2)$
- ③ $10000/5$
- ④ $(2 \times 10000)/5$

2. LOC 기법에 의하여 예측된 총 라인수가 36000 라인, 개발에 참여할 프로그래머가 6 명, 프로그래머들의 평균 생산성이 월간 300 라인일 때 개발에 소요되는 기간을 계산한 결과로 가장 옳은 것은? ()

- ① 5 개월
- ② 10 개월
- ③ 15 개월
- ④ 20 개월

3. 소프트웨어 아키텍처와 관련한 설명으로 틀린 것은? ()

- ① 파이프 필터 아키텍처에서 데이터는 파이프를 통해 양방향으로 흐르며, 필터 이동 시 오버헤드가 발생하지 않는다
- ② 외부에서 인식할 수 있는 특성이 담긴 소프트웨어의 골격이 되는 기본 구조로 볼 수 있다
- ③ 데이터 중심 아키텍처는 공유 데이터 저장소를 통해 접근자 간의 통신이 이루어지므로 각 접근자의 수정과 확장이 용이하다
- ④ 이해 관계자들의 품질 요구사항을 반영하여 품질 속성을 결정한다.

4. 소프트웨어 아키텍처 모델 중 MVC(Model-View Controller)와 관련한 설명으로 틀린 것은?()

- ① MVC 모델은 사용자 인터페이스를 담당하는 계층의 응집도를 높일 수 있고, 여러 개의 다른 UI 를 만들어 그 사이에 결합도를 낮출 수 있다.
- ② 모델(Model)은 뷰(View)와 제어(Controller) 사이에서 전달자 역할을 하며, 뷰마다 모델 서브시스템이 각각 하나씩 연결된다.
- ③ 뷰(View)는 모델(Model)에 있는 데이터를 사용자 인터페이스에 보이는 역할을 담당한다.
- ④ 제어(Controller)는 모델(Model)에 명령을 보냄으로써 모델의 상태를 변경할 수 있다

5. 애자일 기법에 대한 설명으로 맞지 않은 것은?()

- ① 절차와 도구보다 개인과 소통을 중요하게 생각한다
- ② 계획에 중점을 두어 변경 대응이 난해하다
- ③ 소프트웨어가 잘 실행되는데 가치를 둔다
- ④ 고객과의 피드백을 중요하게 생각한다

6. 디자인 패턴을 이용한 소프트웨어 재사용으로 얻어지는 장점이 아닌 것은?()

- ① 소프트웨어 코드의 품질을 향상시킬 수 있다
- ② 개발 프로세스를 무시할 수 있다
- ③ 개발자들 사이의 의사소통을 원활하게 할 수 있다
- ④ 소프트웨어의 품질과 생산성을 향상시킬 수 있다

7. 디자인 패턴을 이용한 소프트웨어 재사용으로 얻어지는 장점이 아닌 것은?()

- ① 소프트웨어 구조 파악이 용이하다.
- ② 객체지향 설계 및 구현의 생산성을 높이는데 적합하다.
- ③ 재사용을 위한 개발 시간이 단축된다.
- ④ 절차형 언어와 함께 이용될 때 효율이 극대화된다.

8. 모듈의 독립성을 높이기 위한 결합도(Coupling)와 관련한 설명으로 틀린 것은?()

- ① 오류가 발생했을 때 전파되어 다른 오류의 원인이 되는 파문 효과(Ripple Effect)를 최소화해야 한다.
- ② 인터페이스가 정확히 설정되어 있지 않을 경우 불필요한 인터페이스가 나타나 모듈 사이의 의존도는 높아지고 결합도가 증가한다.
- ③ 모듈들이 변수를 공유하여 사용하게 하거나 제어 정보를 교류하게 함으로써 결합도를 낮추어야 한다.
- ④ 다른 모듈과 데이터 교류가 필요한 경우 전역 변수(Global Variable)보다는 매개 변수(Parameter)를 사용하는 것이 결합도를 낮추는 데 도움이 된다.

9. 요구사항 관리 도구의 필요성으로 틀린 것은? ()

- ① 요구사항 변경으로 인한 비용 편익 분석
- ② 기존 시스템과 신규 시스템의 성능 비교
- ③ 요구사항 변경의 추적
- ④ 요구사항 변경에 따른 영향 평가

10. Sequence Diagram(순차다이어그램)과 관련한 설명으로 틀린 것은?()

- ① 객체들의 상호 작용을 나타내기 위해 사용한다
- ② 시간의 흐름에 따라 객체들이 주고 받는 메시지의 전달 과정을 강조한다
- ③ 동적 다이어그램보다는 정적 다이어그램에 가깝다
- ④ Interaction Diagram 의 한 종류로 볼 수 있다

11. 사용자 인터페이스(UI)의 특징으로 틀린 것은?()

- ① 구현하고자 하는 결과의 오류를 최소화한다
- ② 사용자의 편의성을 높임으로써 작업 시간을 증가시킨다.
- ③ 막연한 작업 기능에 대해 구체적인 방법을 제시하여 준다
- ④ 사용자 중심의 상호 작용이 되도록 한다

12. 소프트웨어 설계에서 요구사항 분석에 대한 설명으로 틀린 것은?()

- ① 소프트웨어가 무엇을 해야 하는가를 추적하여 요구사항 명세를 작성하는 작업이다
- ② 사용자의 요구를 추출하여 목표를 정하고 어떤 방식으로 해결할 것인지 결정하는 단계이다.
- ③ 소프트웨어 시스템이 사용되는 동안 발견되는 오류를 정리하는 단계이다
- ④ 소프트웨어 개발의 출발점인면서 실질적인 첫 번째 단계이다

13. 소프트웨어 공학에서 워크스루(Walkthrough)에 대한 설명으로 틀린 것은?()

- ① Use Case 를 확장하여 명세하거나 설계 다이어그램, 원시코드, 테스트 케이스 등에 적용할 수 있다.
- ② 복잡한 알고리즘 또는 반복, 실시간 동작, 병행 처리와 같은 기능이나 동작을 이해하려고 할 때 유용하다.
- ③ 인스펙션(Inspection)과 동일한 의미를 가진다.
- ④ 단순한 테스트 케이스를 이용하여 프로젝트를 수작업으로 수행해 보는 것이다

14. 공학적으로 잘된 소프트웨어(Well Engineered Software)의 설명 중 틀린 것은?()

- ① 소프트웨어는 유지보수가 용이해야 한다.
- ② 소프트웨어는 신뢰성이 높아야 한다.
- ③ 소프트웨어는 사용자 수준에 무관하게 일관된 인터페이스를 제공해야 한다.
- ④ 소프트웨어는 충분한 테스트를 거쳐야 한다

15. 소프트웨어 모델링과 관련한 설명으로 틀린 것은?()

- ① 모델링 작업의 결과물은 다른 모델링 작업에 영향을 줄 수 없다.

- ② 구조적 방법론에서는 DFD(Data Flow Diagram), DD(Data Dictionary) 등을 사용하여 요구사항의 결과를 표현한다.
- ③ 객체지향 방법론에서는 UML 표기법을 사용한다.
- ④ 소프트웨어 모델을 사용할 경우 개발될 소프트웨어에 대한 이해도 및 이해 당사자 간의 의사소통 향상에 도움이 된다

16. UML(Unified Modeling Language)에 대한 설명 중 틀린 것은?()

- ① 기능적 모델은 사용자 측면에서 본 시스템 기능이며, UML에서는 Use case Diagram 을 사용한다.
- ② 정적 모델은 객체, 속성, 연관 관계, 오퍼레이션의 시스템의 구조를 나타내며, UML에서는 Class Diagram 을 사용한다.
- ③ 동적 모델은 시스템의 내부 동작을 말하며, UML에서는 Sequence Diagram, State Diagram, Activity Diagram 을 사용한다.
- ④ State Diagram 은 객체들 사이의 메시지 교환을 나타내며, Sequence Diagram 은 하나의 객체가 가진 상태와 그 상태의 변화에 의한 동작 순서를 나타낸다

17. 유스케이스 다이어그램(Use Case Diagram)에 관련된 내용으로 틀린 것은?()

- ① 시스템과 상호 작용하는 외부 시스템은 액터로 파악해서는 안된다.
- ② 유스케이스는 사용자 측면에서의 요구사항으로, 사용자가 원하는 목표를 달성하기 위해 수행할 내용을 기술한다.
- ③ 시스템 액터는 다른 프로젝트에서 이미 개발되어 사용되고 있으며, 본 시스템과 데이터를 주고받는 등 서로 연동되는 시스템을 말한다.
- ④ 액터가 인식할 수 없는 시스템 내부의 기능을 하나의 유스케이스로 파악해서는 안된다

18. 단위 테스트(Unit Test)와 관련한 설명으로 틀린 것은?()

- ① 구현 단계에서 각 모듈의 개발을 완료한 후 개발자가 명세서의 내용대로 정확히 구현되었는지 테스트한다.
- ② 모듈 내부의 구조를 구체적으로 볼 수 있는 구조적 테스트를 주로 시행한다.
- ③ 필요 데이터를 인자를 통해 넘겨주고, 테스트 완료 후 그 결과값을 받는 역할을 하는 가상의 모듈을 테스트 스텝(Stub)이라고 한다.
- ④ 테스트할 모듈을 호출하는 모듈도 있고, 테스트할 모듈이 호출하는 모듈도 있다.

19. 소프트웨어 테스트와 관련한 설명으로 틀린 것은? ()

- ① 화이트박스 테스트는 모듈의 논리적인 구조를 체계적으로 점검할 수 있다.
- ② 블랙박스 테스트는 프로그램의 구조를 고려하지 않는다.

- ③ 테스트 케이스에는 일반적으로 시험 조건, 테스트 데이터, 예상 결과가 포함되어야 한다.
- ④ 화이트박스 테스트에서 기본 경로(Basis Path)란 흐름 그래프의 시작 노드에서 종료 노드까지의 서로 독립된 경로로 사이클을 허용하지 않는 경로를 말한다.

20. 테스트 드라이버(Test Driver)에 대한 설명으로 틀린 것은? ()

- ① 시험대상 모듈을 호출하는 간이 소프트웨어이다.
- ② 필요에 따라 매개 변수를 전달하고 모듈을 수행한 후의 결과를 보여줄 수 있다.
- ③ 상향식 통합 테스트에서 사용된다.
- ④ 테스트 대상 모듈이 호출하는 하위 모듈의 역할을 한다.

21. 화이트박스 테스트와 관련한 설명으로 틀린 것은?()

- ① 화이트박스 테스트의 이해를 위해 논리 흐름도(Logic-Flow Diagram)를 이용할 수 있다.
- ② 테스트 데이터를 이용해 실제 프로그램을 실행함으로써 오류를 찾는 동적 테스트(Dynamic Test)에 해당한다.
- ③ 프로그램의 구조를 고려하지 않기 때문에 테스트 케이스는 프로그램 또는 모듈의 요구나 명세를 기초로 결정한다.
- ④ 테스트 데이터를 선택하기 위하여 검증 기준(Test Coverage)을 정한다.

22. 소프트웨어 형상 관리(Configuration Management)에 관한 설명으로 틀린 것은? ()

- ① 소프트웨어에서 일어나는 수정이나 변경을 알아내고 제어하는 것을 의미한다.
- ② 소프트웨어 개발의 전체 비용을 줄이고, 개발 과정의 여러 방해 요인이 최소화되도록 보증하는 것을 목적으로 한다.
- ③ 형상 관리를 위하여 구성된 팀을 "Chief Programmer Team"이라고 한다.
- ④ 형상 관리의 기능 중 하나는 버전 제어 기술이다.

23. 제품 소프트웨어의 형상 관리 역할로 틀린 것은 ()

- ① 형상 관리를 통해 이전 리버전이나 버전에 대한 정보에 접근 가능하여 배포본 관리에 유용
- ② 불필요한 사용자의 소스 수정 제한
- ③ 프로젝트 개발 비용을 효율적으로 관리
- ④ 동일한 프로젝트에 대해 여러 개발자 동시 개발 가능.

24. 형상 관리의 개념과 절차에 대한 설명으로 틀린 것은?()

- ① 형상 식별은 형상 관리 계획을 근거로 형상 관리의 대상이 무엇인지 식별하는 과정이다.
- ② 형상 관리를 통해 가시성과 추적성을 보장함으로써 소프트웨어의 생산성과 품질을 높일 수 있다.

- ③ 형상 통제 과정에서는 형상 목록의 변경 요구를 즉시 수용 및 반영해야 한다. --
- ④ 형상 감사는 형상 관리 계획대로 형상 관리가 진행되고 있는지, 형상 항목의 변경이 요구사항에 맞도록 제대로 이뤄졌는지 등을 살펴보는 활동이다

1. Consider a simple program that gets an input of *Date of Birth* from the user and generates an output of its age. Define a set of test cases for applying Blackbox testing on this program.

test case 1.

test case 2.

test case ____

2. What are the types of errors that are not found through Blackbox testing but found through Whitebox testing? Be specific.

3. (Software Quality Management) Error Index (EI)
A. Interpret the following metric for computing Error Index.

$$EI = \sum (i \times PI_i) / PS$$

B. Explain how to compute the Phase Index (PI).

4. (Project Management) Function Point-based Estimation

A. Interpret the following formulae for Function Point in details.

$$FP = Count_Total \times [0.65 + 0.01 \times \sum(F_i)], \quad i = 1 \text{ to } 14$$

B. What are the pros and cons of FP-based estimation?

Pros:

Cons:

5. Use Case/Object/Dynamic Model 간의 Consistency Explain.