소프트웨어공학

문 1. 예측된 총 라인 수(Line of Code)가 50,000 LOC, 개발에 참여할 프로그래머가 5명, 프로그래머들의 평균 생산성이 500 LOC/PM (Person Month)이라고 할 때 개발에 소요되는 기간은?

① 10 개월 ② 15 개월
③ 20 개월 ④ 30 개월

문 2. 다음 설명에 해당하는 용어는?

> 공식적으로 검토 및 동의되었고 추후 개발의 기초가 되며 오직 공식적인 변경 통제 절차에 의해서만 변경될 수 있는 형상항목

① 마일스톤(milestone)
② 베이스라인(baseline)
③ 산출물(deliverable)
④ 유스케이스(use-case)

문 3. 위험(risk)에 관한 설명으로 옳지 않은 것은?

① 위험은 실제로 발생하여 프로젝트 결과에 부정적인 영향을 끼친 문제이다.
② 프로젝트 위험(project risk)은 프로젝트 일정이나 자원에 영향을 미치는 위험으로 경험있는 개발인력이 도중에 그만두는 것이 그 예이다.
③ 제품 위험(product risk)은 개발될 소프트웨어의 품질 혹은 성능에 영향을 미치는 위험으로 구매한 컴포넌트가 예상대로 성능을 내지 못하는 것이 그 예이다.
④ 위험은 발생 가능성과 발생시 프로젝트에 미치는 영향의 정도로 계량화될 수 있다.

문 4. 기능 점수(Function Point)에서 트랜잭션 기능의 복잡도를 결정하는 요소로만 나열한 것은?

① DET, RET
② DET, FTR
③ FTR, RET
④ FTR, VAF

문 5. 요구사항 문서의 점검 활동과 그 내용이 바르게 연결된 것은?

① 일관성 점검 – 문서에 있는 요구사항이 상충되지 않는지 검사
② 추적가능성 점검 – 요구사항 문서의 내용이 여러 의미로 해석되는 모호한 점이 있는지 검사
③ 증명가능성 점검 – 요구사항 문서가 모든 기능을 정의하고 시스템 사용자가 의도한 제약 조건을 모두 포함하는지 검사
④ 실현성 점검 – 미래의 기술을 사용하여 요구사항이 실제로 구현될 수 있는지 검사

문 6. 다음 설명에 해당하는 소프트웨어 설계 원리는?

> 범용적 목적의 단일한 인터페이스보다는 클라이언트별 인터페이스를 다수 갖도록 설계한다.

① CCP(Common Closure Principle)
② OCP(Open-Closed Principle)
③ ISP(Interface Segregation Principle)
④ DIP(Dependency Inversion Principle)

문 7. 주어진 한 시점에서 요구사항에 따라 프로그램이 운영되는 확률을 나타내는 신뢰성 척도를 계산하는 방법은?

① $\frac{1}{(\mathrm{MTTF}+\mathrm{MTTR})}\times 100\%$
② $\frac{\mathrm{MTTF}}{(\mathrm{MTTF}+\mathrm{MTBF})}\times 100\%$
③ $\frac{1}{(\mathrm{MTTF}+\mathrm{MTBF})}\times 100\%$
④ $\frac{\mathrm{MTTF}}{(\mathrm{MTTF}+\mathrm{MTTR})}\times 100\%$

문 8. EVM(Earned Value Management)에서는 BCWP(Budgeted Cost of Work Performed), BCWS(Budgeted Cost of Work Scheduled), ACWP(Actual Cost of Work Performed)를 사용하여 여러 성과 지표를 산출한다. 성과 지표 SPI(Schedule Performance Index)를 계산하는 방법으로 옳은 것은?

① $\frac{\mathrm{BCWP}}{\mathrm{ACWP}}$ ② $\frac{\mathrm{BCWS}}{\mathrm{BCWP}}$
③ $\frac{\mathrm{BCWP}}{\mathrm{BCWS}}$ ④ $\frac{\mathrm{ACWP}}{\mathrm{BCWP}}$

문 9. CMMI 단계적(staged) 표현에서 관리 레벨(Managed Level)을 달성하기 위해 반드시 만족해야 하는 프로세스 영역(Process Area)이 아닌 것은?

① 요구사항 개발(Requirements Development)
② 프로젝트 계획수립(Project Planning)
③ 공급업체 계약관리(Supplier Agreement Management)
④ 형상관리(Configuration Management)

문 10. 다음은 UP(Unified Process)의 어떤 단계를 설명한 것인가?

> 시스템 아키텍처 확립을 위한 설계관련 작업의 비중이 크게 나타나고 프로젝트 계획이 상세히 작성된다.

① 도입 단계(inception phase)
② 정련 단계(elaboration phase)
③ 구축 단계(construction phase)
④ 전이 단계(transition phase)

문 11. 코드 리뷰(Code Review)의 기능을 직접적으로 수행할 수 있는 XP(eXtreme Programming)의 실무관행(Practice)은?

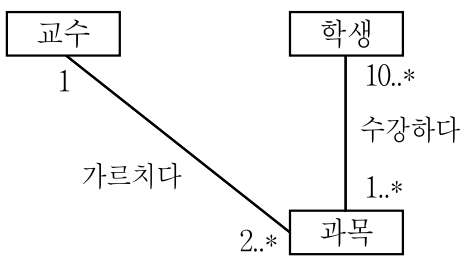
① 단순 설계(simple design)
② 짝 프로그래밍(pair programming)
③ 소규모 릴리즈(small release)
④ 메타포(metaphor)

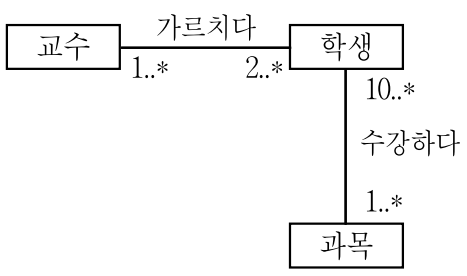
문 12. ISO 12207의 기본 생명주기 프로세스(Primary Life Cycle Process)에 속하지 않는 것은?

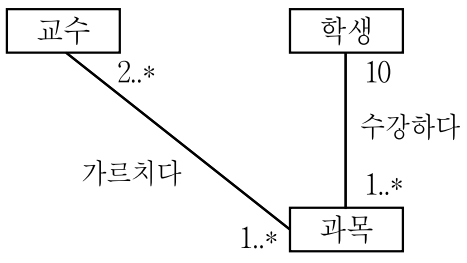
① 획득 프로세스(acquisition process)
② 운영 프로세스(operation process)
③ 형상 관리 프로세스(configuration management process)
④ 유지보수 프로세스(maintenance process)

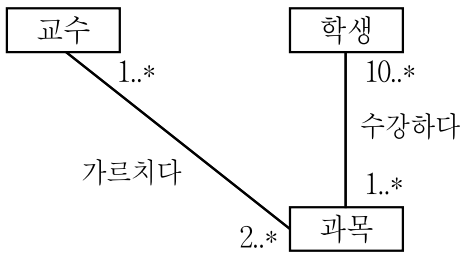
문 13. 다음 사항을 UML 클래스 다이어그램으로 가장 잘 나타낸 것은?

○ '교수'는 적어도 두 '과목' 이상을 가르쳐야 한다.
○ '과목'은 한 명 이상의 '교수'가 가르쳐야 한다.
○ '과목'은 열 명 이상의 '학생'들이 수강해야 한다.
○ '학생'은 한 '과목' 이상을 수강해야 한다.

①


②


③


④


문 14. 분산 시스템 아키텍처의 장점이 아닌 것은?

① 보안성
② 자원 공유
③ 병행성
④ 결함 내성

문 15. 리팩토링(Refactoring)에 대한 설명으로 옳은 것으로만 묶은 것은?

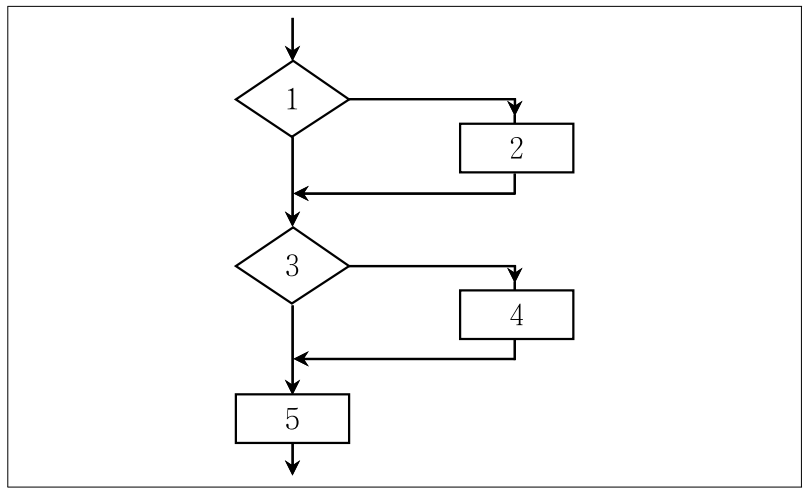
가. 소프트웨어의 디자인을 개선시킨다.
나. 기능을 지속적으로 추가하는 작업이다.
다. 소프트웨어를 이해하기 쉽게 만들고, 프로그램을 빨리 작성하게 도와준다.
라. 겉으로 보이는 동작의 변화없이 내부구조를 변경하는 것이다.
마. 버그를 수정하는 작업이다.

① 가, 나, 다
② 가, 다, 라
③ 나, 다, 마
④ 나, 라, 마

문 16. CPU나 메모리와 같은 컴퓨터 부품의 가격은 다양한 이유로 가격 결정 정책이 자주 변경될 수 있다. 이를 위해 가격 결정 정책을 클래스로 캡슐화하여 쉽게 변경 및 추가할 수 있는 설계패턴은?

① Visitor
② Observer
③ Template Method
④ Strategy

문 17. 다음과 같은 제어흐름을 가진 프로그램을 시험할 때, 제시한 커버리지 기준과 그 기준을 100 % 만족하기 위해 필요한 프로그램 경로들이 바르게 짝지어지지 않은 것은? (단, 제어흐름도의 모든 프로그램 경로는 실행 가능하다)



① 문장 커버리지(statement coverage) : 1→2→3→5, 1→3→4→5
② 분기 커버리지(branch coverage) : 1→2→3→5, 1→3→4→5
③ 경로 커버리지(path coverage) : 1→2→3→4→5, 1→3→5, 1→2→3→5, 1→3→4→5
④ 기본 경로 시험(basis path testing) : 1→3→5, 1→2→3→5

문 18. 다음 C 프로그램이 조건 커버리지(condition coverage)를 100 % 만족하기 위한 테스트 데이터 집합은? (단, short-circuit evaluation은 수행하지 않는다)

```
void foo(int x, int y, int z) {
    if (x>10 && y==10) z=5;
    if (x==10 || z>3) z=z+10;

    printf("%d", z);
}
```

① (x: 10, y: 10, z:10), (x: 20, y: 10, z: 3)
② (x: 10, y: 20, z: 0), (x: 20, y: 20, z: 20)
③ (x: 20, y: 10, z: 3), (x: 20, y: 20, z: 20)
④ (x: 20, y: 10, z:10), (x: 10, y: 20, z: 0)

문 19. 여러 객체들이 서로 메시지를 주고받는 상호 작용을 특정 객체 안에 캡슐화하여 서로의 존재를 모르는 상태에서도 메시지를 주고받으며 협력할 수 있도록 하는 설계패턴은?

① Singleton
② Mediator
③ Command
④ Decorator

문 20. 프로젝트 관리를 위한 도구 및 기법이 아닌 것은?

① Use-case Diagram
② PERT Chart
③ GANTT Chart
④ Work Breakdown Structure