

## 1과목 : 소프트웨어 설계

### 1. User Interface 설계 시 오류 메시지나 경고에 관한 지침으로 가장 거리가 먼 것은?

- ① 메시지는 이해하기 쉬워야 한다.
- ② 오류로부터 회복을 위한 구체적인 설명이 제공되어야 한다.
- ③ 오류로 인해 발생 될 수 있는 부정적인 내용을 적극적으로 사용자에게 알려야 한다.
- ④ 소리나 색의 사용을 줄이고 텍스트로만 전달하도록 한다.

#### <문제 해설>

- 1) 직관성(Intuitiveness) : 누구나 쉽게 이해하고, 쉽게 사용할 수 있어야 함.
- 2) 유효성(Effectiveness) : 정확하고 완벽하게 사용자의 목표가 달성될 수 있도록 제작해야 함.
- 3) 학습성(Learnability) : 초보와 숙련자 모두가 쉽게 배우고 사용할 수 있게 제작해야 함.
- 4) 유연성(Flexibility) : 사용자의 인터랙션을 최대한 포용하고, 실수를 방지할 수 있도록 제작해야 함.

[해설작성자 : 기묘]

### 2. 다음 중 애자일(Agile) 소프트웨어 개발에 대한 설명으로 틀린 것은?

- ① 공정과 도구보다 개인과의 상호작용을 더 가치 있게 여긴다.
- ② 동작하는 소프트웨어보다는 포괄적인 문서를 가치 있게 여긴다.
- ③ 계약 협상보다는 고객과의 협력을 가치 있게 여긴다.
- ④ 계획을 따르기보다 변화에 대응하기를 가치 있게 여긴다.

#### <문제 해설>

##### 애자일 방법론 특징

- 1) 프로젝트 요구사항은 '기능' 중심
- 2) 공정과 도구보다 '개인'과 소통을 중요시
- 3) '변화'에 유연하고 신속한 대처
- 4) '고객'과의 피드백을 중요시

[해설작성자 : DaeguYoungsu]

### 3. 소프트웨어 설계에서 요구사항 분석에 대한 설명으로 틀린 것은?

- ① 소프트웨어가 무엇을 해야하는가를 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 출발점이면서 실질적인 첫 번째 단계이다.

#### <문제 해설>

##### 요구 사항 개발 프로세스

도출 - 분석 - 명세 - 확인

[해설작성자 : 골프공]

### 4. 객체지향 기법에서 상위 클래스의 메소드와 속성을 하위 클래스가 물려받는 것을 의미하는 것은?

- ① Abstraction
- ② Polymorphism
- ③ Encapsulation
- ④ Inheritance

#### <문제 해설>

#### 객체지향의 주요 개념(원칙)

##### 캡슐화(Encapsulation)

- 데이터와 데이터를 처리하는 함수를 하나로 묶은 것
- 캡슐화된 객체의 세부 내용이 은폐되어 변경이 발생해도 오류의 파급효과가 적음
- 캡슐화된 객체들은 재사용이 용이함
- 인터페이스가 단순해지고 객체간의 결합도가 낮아짐

##### 상속성(Inheritance)

- 객체지향 프로그래밍에서 이미 정의된 상위 클래스(부모 클래스)의 모든 속성과 연산을 하위 클래스가 상속(물려)받는 것
- 하위 클래스는 상위 클래스로부터 받은 속성과 연산 외에도 새로운 것을 추가 가능
- 클래스의 재사용, 소프트웨어의 재사용을 높이는 중요한 개념

+ 속성과 연산 등을 물려주는 클래스를 상위 클래스 또는 슈퍼 클래스, 물려받는 클래스를 하위 클래스 또는 서브 클래스라고 한다.

##### 다형성(Polymorphism)

- 하나의 메시지에 대해 각 객체가 갖고 있는 고유한 방법대로 응답하는 것을 의미
- 하나의 클래스나 메서드가 다양한 방식으로 동작이 가능한 것을 의미
- 오버로딩과 오버라이딩이 존재

[해설작성자 : comcbt.com 이용자]

### 5. 설계 기법 중 하향식 설계 방법과 상향식 설계 방법에 대한 비교 설명으로 가장 옳지 않은 것은?

- ① 하향식 설계에서는 통합 검사 시 인터페이스가 이미 정의되어 있어 통합이 간단하다.
- ② 하향식 설계에서 레벨이 낮은 데이터 구조의 세부 사항은 설계초기 단계에서 필요하다.
- ③ 상향식 설계는 최하위 수준에서 각각의 모듈들을 설계하고 이러한 모듈이 완성되면 이들을 결합하여 검사한다.
- ④ 상향식 설계에서는 인터페이스가 이미 성립되어 있지 않더라도 기능 추가가 쉽다.

#### <문제 해설>

##### 1. 하향식 통합 테스트 ( Top Down Integration Test )

- a. 깊이 우선 통합법, 넓이 우선 통합법 사용
- b. 테스트 초기 부터 사용자에게 시스템 구조 보여줄 수 있다.
- c. 상위 모듈에서는 tc를 사용하기 어렵다.
- d. 상위 모듈에서 하위 모듈 방향으로 통합하면서 테스트하는 기법
- e. 절차

- 주요 제어 모듈은 작성된 프로그램을 사용하고, 주요 제어 모듈의 종속 모듈들은 스텝 ( stub )로 대체 한다.

- 깊이우선 or 넓이 우선 등의 통합방식에 따라, 하위 모듈인 스텝들이 한 번에 하나씩 실제로 돌로 교체된다.

- 모듈이 통합될 때마다 테스트 실시  
- 새로운 오류가 발생하지 않음을 보증하기 위해 회귀 테스트 실시

##### 2. 상향식 통합 테스트 ( Bottom Up Integration Test )

- a. 하위 모듈에서 상위 모듈 방향으로 통합하면서 테스트 하는 방법
- b. 가장 하위 단계의 모듈부터 통합 및 테스트가 수행되므로 스텝은 필요하지 않다.

c. 하나의 주요 제어 모듈과 관련된 종속 모듈의 그룹인 클러스터 ( cluster ) 은 필요하다.  
 d. 절차

- 하위 모듈을 클러스터로 결합
- 상위 모듈에서 데이터의 입출력을 확인하기 위해, 모듈인 드라이버 작성
- 통합된 클러스터 단위로 테스트
- 테스트가 완료되면 클러스터는 프로그램 구조의 상위로 이동 하여 결합하고, 드라이버는 실제 모듈로 대체 [해설작성자 : mook]

## 6. 자료흐름도(DFD)의 각 요소별 표기 형태의 연결이 옳지 않은 것은?

- ① Process : 원                      ② Data Flow : 화살표  
 ③ Data Store : 삼각형          ④ Terminator : 사각형

<문제 해설>

자료흐름도 4가지 구성요소

- 처리 Process : 원
- 자료흐름 Data Flow : 화살표
- 자료저장소 Data Store : 평행선
- 단말 Terminal : 사각형

[해설작성자 : 양기모픽]

## 7. 소프트웨어 개발에 이용되는 모델(Model)에 대한 설명 중 거리가 먼 것은?

- ① 모델은 개발 대상을 추상화하고 기호나 그림 등으로 시각적으로 표현한다.  
 ② 모델을 통해 소프트웨어에 대한 이해도를 향상시킬 수 있다.  
 ③ 모델을 통해 이해 당사자 간의 의사소통이 향상된다.  
 ④ 모델을 통해 향후 개발될 시스템의 유추는 불가능하다.

<문제 해설>

소프트웨어 개발시 모델은 향후 개발될 시스템을 유추하기 위해서 하는 활동이며, 주로 시스템 개발자가 실행합니다. [해설작성자 : 막가물]

## 8. 다음의 설명에 해당하는 언어는?

객체 지향 시스템을 개발할 때 산출물을 명세화, 시각화, 문서화하는데 사용된다. 즉, 개발하는 시스템을 이해하기 쉬운 형태로 표현하며 분석가, 의뢰인, 설계자가 효율적인 의사소통을 할 수 있게 해준다. 따라서, 개발 방법론이나 개발 프로세스가 아니라 표준화된 모델링 언어이다.

- ① JAVA                                  ② C  
 ③ UML                                    ④ Python

<문제 해설>

Unified Modeling Language

[해설작성자 : comcbt.com 이용자]

## 9. 다음 내용이 설명하는 UML의 도구는?

- 디자인, 사용방법설명, 평가 등을 위해 실제 화면과 유사하게 만든 정적인 형태의 모형
- 시각적으로만 구성 요소를 배치하는 것으로 일반적으로 실제로 구현되지는 않음

- ① 스토리보드(Storyboard)          ② 목업(Mockup)  
 ③ 프로토타입(Prototype)        ④ 유스케이스(Usecase)

<문제 해설>

스토리보드 : 디자이너와 개발자가 최종적으로 참고하는 작업 지침서

상단이나 우측에 제목, 작성자등을 입력하고 좌측에는 UI화면, 우측엔 디스크립션을 기입한다.

프로토타입 : 와이어프레임이나 스토리보드등에 인터랙션을 적용함으로써 실제 구현된 것처럼 테스트가 가능한 동적인 형태의 모형

유스케이스 : 사용자 측면에서의 요구사항으로, 사용자가 원하는 목표를 달성하기 위해 수행할 내용을 기술한다.

[해설작성자 : 코라투]

## 10. 애자일(Agile) 기법 중 스크럼(Scrum)과 관련된 용어에 대한 설명이 틀린 것은?

- ① 스크럼 마스터(Scrum Master)는 스크럼 프로세스를 따르고, 팀이 스크럼을 효과적으로 활용할 수 있도록 보장하는 역할 등을 맡는다.  
 ② 제품 백로그(Product Backlog)는 스크럼 팀이 해결해야 하는 목록으로 소프트웨어 요구사항, 아키텍처 정의 등이 포함될 수 있다.  
 ③ 스프린트(Sprint)는 하나의 완성된 최종 결과물을 만들기 위한 주기로 3달 이상의 장기간으로 결정된다.  
 ④ 속도(Velocity)는 한 번의 스프린트에서 한 팀이 어느 정도의 제품 백로그를 감당할 수 있는지에 대한 추정치로 볼 수 있다.

<문제 해설>

스프린트

- 실제 개발을 2~4주간 진행하는 과정

- 스프린트 백로그에 작성된 Task를 대상으로 작업 시간을 측정 후 담당 개발 자에게 할당합니다.

- Task는 할 일, 진행 중, 완료의 상태로 구성된다.

[해설작성자 : MSKIM]

## 11. UML 다이어그램 중 정적 다이어그램이 아닌 것은?

- ① 컴포넌트 다이어그램          ② 배치 다이어그램  
 ③ 순차 다이어그램                ④ 패키지 다이어그램

<문제 해설>

정적(구조적)다이어그램 : 클래스컴백복패

[해설작성자 : 함초롬다솜]

UML 다이어그램의 분류

정적구조: [ 클래스, 객체, 패키지, 컴포넌트, 복합구조, 배치]  
 동적구조: [ 유스케이스, 상태, 활동, 시퀀스, 통신, 상호작용, 타이밍]

1) 클래스 다이어그램: 객체를 생성하기 위한 도구

2) 객체 다이어그램: 객체 간의 연결관계를 표현

3) 패키지 다이어그램: 객체들의 그룹화된 표현

4) 컴포넌트 다이어그램: 시스템을 구성하는 컴포넌트 사이에 의존관계를 표현

\* 컴포넌트는 1개이상의 클래스로 구현됨.

5) 복합구조 다이어그램: 컴포넌트나 클래스의 내부 구조를 보여주기 위해 사용 → 내부 연결 형태를 표현

6) 배치 다이어그램: H/W자원에 연결된 S/W컴포넌트 배치를 표현

- 7) 유스케이스 다이어그램: 사용자의 시각에서 소프트웨어의 범위와 기능을 표현  
 8) 상태 다이어그램: 시간에 따라 객체가 변하는 상태를 표현  
 9) 활동 다이어그램: 객체의 동적행위를 활동으로 표현  
 10) 시퀀스 다이어그램: 객체간 상호작용을 시간 순서에 따른 메시지로 표현  
 11) 통신 다이어그램: 객체간 상호작용을 관계형태의 메시지로 표현  
 12) 상호작용 다이어그램: 객체들의 상호작용 관계를 표현  
 13) 타이밍 다이어그램: 객체간 상태변화를 구체적인 시간으로 표현  
 [해설작성자 : SD.신동]

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

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

<문제 해설>  
 노력(M/M)=원시 코드 라인 수(LOC)/(1인당 월 평균 생산 코드 라인 수)  
 =36,000라인/300라인=120M/M

개발 기간=(M/M)/참여 인원=120(M/M)/6명=20개월  
 [해설작성자 : KMJ2022]

개발 소요기간 계산방법  
 총 라인수 36000라인 / 월간 평균 생산성 300라인 = 120  
 120 / 참여 프로그래머 6 = 20  
 20개월  
 [해설작성자 : 양기모픽]

13. 클래스 설계원칙에 대한 바른 설명은?

- ① 단일 책임원칙 : 하나의 클래스만 변경 가능 해야한다.  
 ② 개방-폐쇄의 원칙 : 클래스는 확장에 대해 열려 있어야 하며 변경에 대해 닫혀 있어야 한다.  
 ③ 리스코프 교체의 원칙 : 여러 개의 책임을 가진 클래스는 하나의 책임을 가진 클래스로 대체되어야 한다.  
 ④ 의존관계 역전의 원칙 : 클라이언트는 자신이 사용하는 메소드와 의존관계를 갖지 않도록 해야 한다.

<문제 해설>  
 단일 책임원칙 : 하나의 객체는 하나의 동작만의 책임을 가짐  
 개방-폐쇄의 원칙 : 클래스는 확장에 대해 열려 있어야 하며 변경에 대해 닫혀 있어야 한다.  
 리스코프 교체의 원칙 : 특정 메소드가 상위 타입을 인자로 사용할 때, 그 타입의 하위 타입도 문제 없이 작동해야 함  
 의존관계 역전의 원칙 : 상위 계층이 하위 계층에 의존하는 전통적인 의존관계를 반전(역전)시킴으로써 상위 계층이 하위 계층의 구현으로부터 독립되게 할 수 있음  
 [해설작성자 : 양기모픽]

14. GoF(Gangs of Four) 디자인 패턴에서 생성(Creational) 패턴에 해당하는 것은?

- ① 컴퍼지트(Composite)                      ② 어댑터(Adapter)  
 ③ 추상 팩토리(Abstract Factory)        ④ 옵서버(Observer)

<문제 해설>

생성 : 3. 추상 팩토리(Abstract Factory)  
 구조 : 1. 컴퍼지트(Composite), 2. 어댑터(Adapter)  
 행위 : 4. 옵서버(Observer)

생성패턴  
 - 추상팩토리(Abstract Factory)  
 - 빌더(Builder)  
 - 팩토리메서드(FactoryMethod)  
 - 프로토타입(Prototype)  
 - 싱글톤(Singleton)

구조패턴  
 - 어댑터(Adapter)  
 - 브리지(Bridge)  
 - 컴퍼지트(Composite)  
 - 데코레이터(Decorator)  
 - 파사드(Facade)  
 - 플라이웨이트(Flyweight)  
 - 프록시(Proxy)

행위패턴  
 - 책임 연쇄(Chain of Responsibility)  
 - 커맨드(Command)  
 - 인터프리터(Interpreter)  
 - 이터레이터(Iterator)  
 - 중재자(Mediator)  
 - 메멘토(Memento)  
 - 옵서버(Observer)  
 - 상태(State)  
 - 전략(Strategy)  
 - 템플릿메서드(Template Method)  
 - 방문자(Visitor)  
 [해설작성자 : 양기모픽]

15. 아키텍처 설계과정이 올바른 순서로 나열된 것은?

- ㉠ 설계 목표 설정  
 ㉡ 시스템 타입 결정  
 ㉢ 스타일 적용 및 커스터마이징  
 ㉣ 서브시스템의 기능, 인터페이스 동작 작성  
 ㉤ 아키텍처 설계 검토

- ① ㉠ → ㉡ → ㉢ → ㉣ → ㉤  
 ② ㉤ → ㉠ → ㉡ → ㉣ → ㉢  
 ③ ㉠ → ㉤ → ㉡ → ㉣ → ㉢  
 ④ ㉠ → ㉡ → ㉢ → ㉤ → ㉣

<문제 해설>  
 소프트웨어 아키텍처의 설계과정  
 설계목표설정 - 시스템 타입 결정 - 아키텍처 패턴 적용 - 서브시스템 구체화 - 검토  
 [해설작성자 : 궁디팡팡]

16. 사용자 인터페이스를 설계할 경우 고려해야 할 가이드라인과 가장 거리가 먼 것은?

- ① 심미성을 사용성보다 우선하여 설계해야 한다.  
 ② 효율성을 높이기 위해 설계해야 한다.  
 ③ 발생하는 오류를 쉽게 수정할 수 있어야 한다.  
 ④ 사용자에게 피드백을 제공해야 한다.

<문제 해설>  
 사용자 친화적이게 설계되어야 하기에 사용성이 최우선으로 고려되어야 함

[해설작성자 : 동차합격가즈아]

17. 소프트웨어 설계에서 자주 발생하는 문제에 대한 일반적이고 반복적인 해결 방법을 무엇이라고 하는가?

- ① 모듈 분해                      ② 디자인 패턴
- ③ 연관 관계                      ④ 클래스 도출

<문제 해설>

디자인패턴 : 객체 지향 프로그래밍 설계를 할 때 자주 발생하는 문제들을 피하기 위해 사용되는 패턴.

[해설작성자 : comcbt.com 이용자]

18. 객체지향 분석기법의 하나로 객체 모형, 동적 모형, 기능 모형의 3개 모형을 생성하는 방법은?

- ① Wirfs-Block Method      ② Rumbaugh Method
- ③ Booch Method              ④ Jacobson Method

<문제 해설>

Rumbaugh(럼바우) 방법

모든 소프트웨어 구성 요소를 그래픽 표기법을 이용하여 모델링하는 기법으로, 객체 모델링 기법(Object Modeling Technique) 라고도 한다..분석활동은 객체모델링 -> 동적 모델링 -> 기능 모델링 순으로 통해 이루어진다

Booch(부치) 방법

미시적(Micro) 개발 프로세스와 거시적(Macro) 개발 프로세스를 모두 사용하는 분석 방법으로, 클래스와 객체들을 분석 및 식별하고 클래스의 속성과 연산을 정의한다.

Jacobson 방법

Use Case를 강조하여 사용하는 분석 방법이다.

Coad와 Yourdon 방법

E-R 다이어그램을 사용하여 객체의 행위를 모델링하며, 객체 식별, 구조 식별, 주제 정의, 속성과 인스턴스 연결 정의, 연산과 메시지 연결 정의 등의 과정으로 구성하는 기법

Wirfs-Brock 방법

분석과 설계 간의 구분이 없고, 고객 명세서를 평가해서 설계 작업까지 연속적으로 수행하는 기법

[해설작성자 : ham]

19. 입력되는 데이터를 컴퓨터의 프로세서가 처리하기 전에 미리 처리하여 프로세서가 처리하는 시간을 줄여주는 프로그램이나 하드웨어를 말하는 것은?

- ① EAI                              ② FEP
- ③ GPL                              ④ Duplexing

<문제 해설>

EAI(Enterprise Application Integration): 기업 응용 프로그램 통합으로 기업용 응용 프로그램의 구조적 통합 방안을 가리킴  
FEP(Front-End Processor): 입력되는 데이터를 컴퓨터의 프로세서가 처리하기 전에 미리 처리하여 프로세서가 차지하는 시간을 줄여주는 프로그램이나 하드웨어

GPL(General Public License): 자유 소프트웨어 재단(OSF)에서 만든 자유 소프트웨어 라이선스

Duplexing: 이중화(데이터베이스의 회복 기법 중 가장 간단한 것)

[해설작성자 : 뽕대가리]

20. 객체 지향 개념 중 하나 이상의 유사한 객체들을 묶어 공통된 특성을 표현한 데이터 추상화를 의미하는 것은?

- ① Method                      ② Class
- ③ Field                          ④ Message

<문제 해설>

Method(메서드) : 클래스로부터 생성된 객체를 사용하는 방법, 객체가 메시지를 받아 실행해야 할 객체의 구체적인 연산  
Message(메시지) : 객체 간 상호 작용을 하기 위한 수단, 객체에게 어떤 행위를 하도록 지시하는 방법

Class(클래스) : 특정 객체 내에 있는 변수와 메서드를 정의하는 일종의 틀, 객체 지향 프로그래밍에서 데이터를 추상화하는 단위

Field(필드) : SQL에서 열 또는 속성이라고 불리는 것

[해설작성자 : 기묘]

## 2과목 : 소프트웨어 개발

21. 클린 코드(Clean Code)를 작성하기 위한 원칙으로 틀린 것은?

- ① 추상화 : 하위 클래스/메소드/함수를 통해 애플리케이션의 특성을 간략하게 나타내고, 상세 내용은 상위 클래스/메소드/함수에서 구현한다.
- ② 의존성 : 다른 모듈에 미치는 영향을 최소화하도록 작성한다.
- ③ 가독성 : 누구든지 읽기 쉽게 코드를 작성한다.
- ④ 중복성 : 중복을 최소화 할 수 있는 코드를 작성한다.

<문제 해설>

추상화 : 상위 클래스; 프로그램 특성만 간략하게, 하위 클래스; 세부적 내용 구현

의존성(배제) : 다른 모듈에 미치는 영향을 최소화하여 코드 변경시 영향이 가지 않도록

중복성(최소화) : 코드의 중복을 최소화

가독성 : 누구든지 읽기 쉽게 작성

단순성 : 코드를 단순하게 작성(프로그램을 최소단위로 분리해 작업을 한번에 하나씩 처리하도록)

[해설작성자 : 센딘]

추상화 : 상위 클래스, 메소드, 함수에서는 간략하게 애플리케이션의 특성을 나타냄, 상세 내용은 하위 클래스, 메소드, 함수에서 구현

[해설작성자 : 붙어보자]

22. 단위 테스트에서 테스트의 대상이 되는 하위 모듈을 호출하고, 파라미터를 전달하는 가상의 모듈로 상황식 테스트에 필요한 것은?

- ① 테스트 스텝(Test Stub)
- ② 테스트 드라이버(Test Driver)
- ③ 테스트 스위트(Test Suites)
- ④ 테스트 케이스(Test Case)

<문제 해설>

테스트 스텝 - 제어 모듈이 호출하는 타 모듈의 기능을 단순히 수행하는 도구로, 일시적으로 필요한 조건만을 가지고 있는 테스트용 모듈

테스트 스위트 - 테스트 대상 컴포넌트나 모듈, 시스템에 사용되는 테스트 케이스의 집합

테스트 케이스 - 사용자의 요구사항을 정확히 준수했는지 확인하기 위한 입력 값, 실행조건, 기대결과 등으로 만들어진 테스트 항목의 명세서

[해설작성자 : 김라마]

23. 스택(Stack)에 대한 옳은 내용으로만 나열된 것은?

- ㉠ FIFO 방식으로 처리된다.  
 ㉡ 순서 리스트의 뒤(Rear)에서 노드가 삽입되며, 앞(Front)에서 노드가 제거된다.  
 ㉢ 선형 리스트의 양쪽 끝에서 삽입과 삭제가 모두 가능한 자료 구조이다.  
 ㉣ 인터럽트 처리, 서브루틴 호출 작업 등에 응용된다.

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

<문제 해설>

FIFO-큐 방식

한쪽에서 삽입과 삭제가 모두 이루어 짐

[해설작성자 : 꾸꾸까까]

㉠ 큐 방식

㉡ 데크(Deque)설명

[해설작성자 : ITK]

24. 소프트웨어 모듈화의 장점이 아닌 것은?

- ① 오류의 파급 효과를 최소화한다.  
 ② 기능의 분리가 가능하여 인터페이스가 복잡하다.  
 ③ 모듈의 재사용 가능으로 개발과 유지보수가 용이하다.  
 ④ 프로그램의 효율적인 관리가 가능하다.

<문제 해설>

모듈화의 장점

- 프로그램의 효율적인 관리
- 이해하기 쉬운 소프트웨어
- 소프트웨어 시험, 통합, 수정 시 용이
- 모듈 재사용 가능
- 오류의 파급력 최소화

[해설작성자 : 이케이]

25. 소프트웨어 프로젝트 관리에 대한 설명으로 가장 옳은 것은?

- ① 개발에 따른 산출물 관리  
 ② 소요인력은 최대화하되 정책 결정은 신속하게 처리  
 ③ 주어진 기간은 연장하되 최소의 비용으로 시스템을 개발  
 ④ 주어진 기간 내에 최소의 비용으로 사용자를 만족시키는 시스템을 개발

<문제 해설>

- 프로젝트 관리는 주어진 기간 내에 최소의 비용으로 사용자를 만족시키는 시스템을 개발하기 위한 전반적인 활동
- 프로젝트 관리는 소프트웨어 개발 계획을 세우고 분석, 설계, 구현 등의 작업을 통제하는 것으로 소프트웨어 생명 주기의 전 과정에 걸쳐 진행됨
- 소프트웨어 프로젝트를 성공적으로 수행하기 위해서는 수행할 작업의 범위, 필요한 자원, 수행 업무, 이정표, 비용 추진 일정들을 알아야 함
- 소요인력은 최소화 하는 것이 좋음
- ※ 프로젝트 관리는 제한된 시간과 비용으로 좋은 품질의 시스템을 개발하여 고객에게 제공함

[해설작성자 : 강썸]

26. 정형 기술 검토(FTR)의 지침으로 틀린 것은?

- ① 의제를 제한한다.

- ② 논쟁과 반박을 제한한다.  
 ③ 문제 영역을 명확히 표현한다.  
 ④ 참가자의 수를 제한하지 않는다.

<문제 해설>

### 정형기술검토(FTR)의 지침

- 오류 검출에 초점을 두고 해결책을 나중으로 미룸(제품 검토의 집중성)
- 검토를 위한 자료를 사전에 배포하여 검토하도록 한다(사전 준비성)
- 의견을 제한하되 충분히 받아들인다(의제의 제한성)
- 안건을 세우면 고수한다(안건 고수성)
- 논쟁과 반박을 제한한다(논쟁 반박의 제한성)
- 문제 영역을 공개한다(문제 공개성)
- 참가자의 수를 제한한다(참가 인원의 제한성)
- 발견된 오류는 문서화한다(문서성)

[해설작성자 : 강썸]

27. 소프트웨어 재공학의 주요 활동 중 기존 소프트웨어 시스템을 새로운 기술 또는 하드웨어 환경에서 사용할 수 있도록 변환하는 작업을 의미하는 것은?

- ① Analysis                      ② Migration  
 ③ Restructuring                ④ Reverse Engineering

<문제 해설>

Analysis: 기존 소프트웨어를 분석하여 재공학 대상을 선정하는 것

Migration: 기존 소프트웨어를 다른 운영체제나 하드웨어 환경에서 사용할 수 있도록 변환하는 작업

Restructuring: 기존 소프트웨어를 향상시키기 위하여 코드를 재구성하는 작업(기능과 외적 동작은 변하지 않음)

Reverse Engineering: 기존 소프트웨어를 분석하여 소스코드를 얻어내는 작업(소스코드로 소프트웨어를 만드는 작업의 역작업)

[해설작성자 : KNAPA]

28. 정보시스템 개발 단계에서 프로그래밍 언어 선택 시 고려할 사항으로 가장 거리가 먼 것은?

- ① 개발 정보시스템의 특성                      ② 사용자의 요구사항  
 ③ 컴파일러의 가용성                          ④ 컴파일러의 독창성

<문제 해설>

프로그래밍 언어의 선정 기준

- 친밀감
- 언어의 능력
- 처리의 효율성
- 프로그램 구조
- 프로그램의 길이

[해설작성자 : 지나가던 이용자]

프로그래밍 언어의 선정 기준 :

친밀감, 언어의 능력, 처리의 효율성, 프로그램 구조, 프로그램의 길이, 이식성, 과거의 개발 실적, 알고리즘과 계산상의 난이도, 자료 구조의 난이도, 성능 고려 사항들, 대상 업무의 성격, 소프트웨어의 수행 환경, 개발 담당자의 경험과 지식, 사용자의 요구사항, 컴파일러의 이용 가능성

[해설작성자 : 지나가던 이용자]

29. 소프트웨어 패키징에 대한 설명으로 틀린 것은?

- ① 패키징은 개발자 중심으로 진행한다.  
 ② 신규 및 변경 개발소스를 식별하고, 이를 모듈화하여 상용제품으로 패키징한다.  
 ③ 고객의 편의성을 위해 매뉴얼 및 버전관리를 지속적으로 한다.



- ④ 범용 환경에서 사용이 가능하도록 일반적인 배포 형태로 패키징이 진행된다.

<문제 해설>

소프트웨어 패키징은 소비자 중심으로 진행된다.

[해설작성자 : 주주]

- 모듈별로 생성한 실행 파일들을 묶어 배포용 설치 파일을 만드는 것.
- 개발자가 아니라 사용자 중심으로 진행함.
- 소스 코드는 향후 관리를 고려하여 모듈화하여 패키징함.
- 사용자가 소프트웨어를 사용하게 될 환경을 이해하여, 다양한 환경에서 소프트웨어를 손쉽게 사용할 수 있도록 일반적인 배포 형태로 패키징함.
- 사용자를 중심으로 진행되는 작업이므로 사용자의 편의성 및 실행 환경을 우선적으로 고려해야 함.

[해설작성자 : 해시]

30. 자료 구조의 분류 중 선형 구조가 아닌 것은?

- ① 트리                      ② 리스트  
③ 스택                      ④ 데크

<문제 해설>

-비선형 구조 : 트리, 그래프

-선형 구조 : 스택, 큐, 데크, 리스트

-파일 구조 : 순차파일, 색인파일, 직접파일

[해설작성자 : 정처기만 보면]

31. 아주 오래되거나 참고문서 또는 개발자가 없어 유지보수 작업이 아주 어려운 프로그램을 의미하는 것은?

- ① Title Code              ② Source Code  
③ Object Code            ④ Alien Code

<문제 해설>

Alien code : 개발자가 퇴사를 했거나 현재 사용하지 않는 환경 및 언어로 작성되었거나 프로그램의 관련 문서나 주석이 없는 상태에서 시간이 흐른 뒤 유지보수가 불가능하거나 힘든 프로그램 소스 코드

[해설작성자 : 이케이]

32. 소프트웨어를 재사용함으로써 얻을 수 있는 이점으로 가장 거리가 먼 것은?

- ① 생산성 증가  
② 프로젝트 문서 공유  
③ 소프트웨어 품질 향상  
④ 새로운 개발 방법론 도입 용이

<문제 해설>

\*\*소프트웨어 재사용함으로써 얻을 수 있는 이점들\*\*

1. 개발 시간과 비용 단축
2. 소프트웨어 품질 및 생산성 향상
3. 프로젝트 실패 위험 감소
4. 시스템 구축 방법에 대한 지식 공유

\*\*소프트웨어 재사용함으로써 얻을 수 있는 문제점들\*\*

1. 재사용할 소프트웨어 선정 필요
2. 시스템에 공통적으로 사용되는 요소 발견 필요
3. 프로그램의 표준화 부족
4. 새로운 개발 방법론 도입이 어려움
5. 재사용을 위한 관리 및 지원 부족
6. 기존 소프트웨어에 재사용 소프트웨어를 추가하기 어려움

[해설작성자 : 강썸]

33. 인터페이스 간의 통신을 위해 이용되는 데이터 포맷이 아닌 것은?

- ① AJTML                      ② JSON  
③ XML                      ④ YAML

<문제 해설>

▶JSON(Javascript Object Notation)

속성-값 쌍 또는 “키-값 쌍”으로 이루어진 데이터 오브젝트를 전달하기 위해 인간이 읽을 수 있는 텍스트를 사용하는 개방형 표준 포맷, AJAX에서 많이 사용되고 XML을 대체하는 주요 데이터 포맷

▶ XML(Extensible Markup Language)

HTML의 단점을 보완한 인터넷 언어, SGML의 복잡한 단점을 개선한 특수한 목적을 갖는 마크업 언어

▶ AJAX(Asynchronous Javascript And XML)

자바스크립트를 사용하여 웹 서버와 클라이언트 간 비동기적으로 XML 데이터를 교환하고 조작하기 위한 웹 기술, XMLHttpRequest 객체를 이용해 전체 페이지를 새로 로드하지 않고 필요한 부분만 로드한다.

▶ REST(Representational State Transfer)

- 웹과 같은 분산 하이퍼미디어 환경에서 자원의 존재/상태 정보를 표준화된 HTTP 메서드로 주고받는 웹 아키텍처(리소스, 메서드, 메시지)
- HTTP URI를 통해 자원을 명시하고, HTTP 메서드(POST, GET, PUT, DELETE)를 통해 해당 자원에 대한 생성, 조회, 갱신, 삭제 등의 명령을 적용할 수 있는 분산 하이퍼미디어 시스템을 위한 소프트웨어 아키텍처이다.

[해설작성자 : 강썸]

34. 프로그램 설계도의 하나인 NS Chart에 대한 설명으로 가장 거리가 먼 것은?

- ① 논리의 기술에 중점을 두고 도형을 이용한 표현 방법이다.  
② 이해하기 쉽고 코드 변환이 용이하다.  
③ 화살표나 GOTO를 사용하여 이해하기 쉽다.  
④ 연속, 선택, 반복 등의 제어 논리 구조를 표현한다.

<문제 해설>

NS-Chart(Nassi-Schneiderman Chart)

- 3가지 기본구조만으로 논리를 표현(표준화 가능)
- Flow Chart의 최대 단점인 화살표가 표시되지 않음
- 기본구조의 입구와 출구는 각 하나씩
- 전체적인 알고리즘을 일목요연하게 볼 수 있음

[해설작성자 : polo9332]

35. 순서가 A, B, C, D로 정해진 입력자료를 push, push, pop, push, push, pop, pop, pop 순서로 스택연산을 수행하는 경우 출력 결과는?

- ① B D C A                      ② A B C D  
③ B A C D                      ④ A B D C

<문제 해설>

스택연산 : 입구 및 출구가 하나 A B C D 순으로 입력되었을 때 출력순서는 D C B A 순

push 입력 pop 출력

push A

push A B

pop\_ A / B

push A C / B

push A C D / B

pop\_ A C / B D

pop\_ A / B D C

pop\_ / B D C A

[해설작성자 : .]

### 36. 분할 정복(Divide and Conquer)에 기반한 알고리즘으로

피벗(pivot)을 사용하며 최악의 경우  $\frac{n(n-1)}{2}$  회의 비교를 수행해야 하는 정렬(Sort)은?

- ① Selection Sort                      ② Bubble Sort  
 ③ Insert Sort                      ④ Quick Sort

#### <문제 해설>

- Selection Sort - 배열 내에서 최소값을 찾은 다음 정렬 되지 않은 맨 앞 값과 교환을 하며 정렬을 해 나가는 방법 (정렬 시간 복잡도 모두  $O(n^2)$ )  
 - Bubble Sort - 왼쪽에서 부터 두 데이터를 비교해서 앞에 있는 데이터가 뒤에 있는 데이터 보다 크면 자리를 바꾸는 정렬 알고리즘(정렬 시간 복잡도 모두  $O(n^2)$ )  
 - Insert Sort - 한 개의 값을 추출한 다음 앞으로 비교해서 본인의 자리를 알맞게 찾아가게끔 하는 정렬 방법. 정렬 시간 복잡도는 최상일 경우  $O(n)$ , 평균과 최악일 경우  $O(n^2)$   
 - 병합 정렬 - 또한 분할 정복에 기반한 알고리즘으로 리스트를 1 이하인 상태까지 절반으로 자른 다음 재귀적으로 합병 정렬을 이용해서 전체적인 리스트를 합병하는 정렬 과정.  $O(n \log n)$   
 ※  $n(n-1)/2 = O(n^2)$   
 [해설작성자 : 나태규]

정렬이란? 여러 개의 자료를 순서에 따라 나열하는 방법  
 8개 종류: [선택, 버블, 삽입, 쉘, 퀵, 힙, 이진병합, 버킷정렬]

종류 최상 평균 최악  
 선택 정렬  $O(n)$   $O(n^2)$   $O(n^2)$   
 버블 정렬  $O(n^2)$   $O(n^2)$   $O(n^2)$   
 삽입 정렬  $O(n^2)$   $O(n^2)$   $O(n^2)$   
 쉘 정렬  $O(n)$   $O(n^{1.5})$   $O(n^2)$   
 퀵 정렬  $O(n \log n)$   $O(n \log n)$   $O(n^2)$   
 힙 정렬  $O(n \log n)$   $O(n \log n)$   $O(n \log n)$   
 이진병합 정렬  $O(n \log n)$   $O(n \log n)$   $O(n \log n)$   
 버킷 정렬  $O(dn)$   $O(dn)$   $O(dn)$   
 [해설작성자 : SD.신동]

### 37. 화이트 박스 검사 기법에 해당하는 것으로만 짝지어진 것은?

- ㉠ 데이터 흐름 검사  
 ㉡ 루프 검사  
 ㉢ 동등 분할 검사  
 ㉣ 경계값 분석  
 ㉤ 원인 결과 그래프 기법  
 ㉥ 오류예측 기법

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

#### <문제 해설>

화이트박스 테스트의 종류  
 1. 기초경로 검사  
 2. 제어구조 검사

화이트박스 테스트 검증 기준  
 1. 문장 검증 기준  
 2. 분기 검증 기준

3. 조건 검증기준  
 4. 분기/조건 기준

블랙박스 테스트의 종류  
 1. 동치(동등)분할 검사  
 2. 경계값 분석  
 3. 원인-효과 검사  
 4. 오류 예측 검사  
 5. 비교검사  
 [해설작성자 : JYH]

화이트박스 테스트 종류  
 기초 경로 검사(Basic Path Testing)  
 조건 검사(Condition Testing)  
 루프 검사(Loop Testing)  
 데이터 흐름 검사(Data Flow Testing)  
 [해설작성자 : User]

### 38. 소프트웨어 품질 관련 국제 표준인 ISO/IEC 25000에 관한 설명으로 옳지 않은 것은?

- ① 소프트웨어 품질 평가를 위한 소프트웨어 품질평가 통합모델 표준이다.  
 ② System and Software Quality Requirements and Evaluation으로 줄여서 SQuaRE라고도 한다.  
 ③ ISO/IEC 2501n에서는 소프트웨어의 내부 측정, 외부 측정, 사용품질 측정, 품질 측정 요소 등을 다룬다.  
 ④ 기존 소프트웨어 품질 평가 모델과 소프트웨어 평가 절차 모델인 ISO/IEC 9126과 ISO/IEC 14598을 통합하였다.

#### <문제 해설>

ISO### ISO/IEC 25000  
 - ISO 9126, ISO 14598, ISO 12119의 여러 표준 문서를 통합하고 재구성하여 만든 표준 문서  
 - SQuaRE(Software Quality and Requirement Evaluation)라고도 함  
 - 2500n, 2501n, 2502n, 2503n, 2504n의 다섯가지 분야로 나눌수 있고, 확장 분야인 2505n이 있다.  
 - 2500n (9126-1, 품질 관리 지침)  
     SQuaRE에 대한 개요, 전체에 대한 계획과 관리  
 - 2501n (9126-2, 품질 모형)  
     품질 모델 및 품질 사용 안내  
 - 2503n (9126-3, 품질 측정)  
     매트릭을 통한 측정 방법 제시  
 - 2504n (9126-4, 품질 정의(요구사항))  
     품질 요구사항  
 - 2505n (9126-5, 품질 평가)  
     품질 프로세스에 관한 개요, 관점들의 평가 프로세스  
 [해설작성자 : 강썬]

ISO/IEC 25000  
 - SW 품질 평가 통합 모델  
 - SQuaRE로도 불림  
 - ISO/IEC 9126, ISO/IEC 12119, ISO/IEC 14598의 3개 표준을 통합한 모델

- 2500n : 개요 및 품질관리  
 - 2501n : 품질모델  
 - 2502n : 품질측정  
 - 2503n : 품질요구  
 - 2504n : 품질평가  
 - 2505n : 확장분야  
 [해설작성자 : 내일부터갓생산다]

### 39. 코드 인스펙션과 관련한 설명으로 틀린 것은?

- ① 프로그램을 수행시켜보는 것 대신에 읽어보고 눈으로 확인하는 방법으로 볼 수 있다.
- ② 코드 품질 향상 기법 중 하나이다.
- ③ 동적 테스트 시에만 활용하는 기법이다.
- ④ 결함과 함께 코딩 표준 준수 여부, 효율성 등의 다른 품질 이슈를 검사하기도 한다.

#### <문제 해설>

##### 코드 인스펙션

- 결함 뿐만 아니라 모든 것이 표준대로 되어 있는 지 확인하기 위한 검토
  - 표준이나 명세서에 서술한 내용과 비교하여 편차와 에러를 식별하기 위해 산출물을 근거로 수행하는 검사
  - 정적 테스트에 가까움
- [해설작성자 : 강썸]

### 40. 프로젝트에 내재된 위험 요소를 인식하고 그 영향을 분석하여 이를 관리하는 활동으로서, 프로젝트를 성공시키기 위하여 위험 요소를 사전에 예측, 대비하는 모든 기술과 활동을 포함하는 것은?

- ① Critical Path Method                      ② Risk Analysis
- ③ Work Breakdown Structure              ④ Waterfall Model

#### <문제 해설>

##### 1. Critical Path Method(CPM, 임계 경로법) :

네트워크를 중심의 논리적 구성. 시간과 비용 문제를 취급. 프로젝트를 일정 기일 내에 완성시키고 해당 계획이 원가의 최소값에 의해 보증되는 등의 최적 스케줄을 구하는 관리 방법.

(주로 건설과 설계를 포함하는 복잡한 일에 이용되어 효과를 발휘 함.

##### 3. Work Breakdown Structure(업무 분업 구조) :

성과 목표 완전 달성을 위한 프로그램. 산업 관리 간접 부문의 기술 혁신형 업무-목표를 설정하여 소정 기간, 자원 내에서 달성하는 형태의 업무-를 효과적으로 수행하기 위한 수법.

##### 4. Waterfall Model(폭포수 모델) :

S/W 개발 생명주기에 기반하고 있는 소프트웨어 개발 기법 중 하나.

한 번 떨어지면 다시 거슬러 올라갈 수 없는 폭포수처럼, 각 개발 단계를 확실히 매듭 짓고 다음 단계로 넘어간다는 의미.

[해설작성자 : 썸딘]

### 3과목 : 데이터베이스 구축

### 41. 데이터베이스 설계 단계 중 물리적 설계 시 고려 사항으로 적절하지 않은 것은?

- ① 스키마의 평가 및 정제                      ② 응답 시간
- ③ 저장 공간의 효율화                      ④ 트랜잭션 처리량

#### <문제 해설>

- o 어떤 인덱스를 만들 것인지에 대한 고려
- o 성능 향상을 위한 개념 스키마의 변경 여부 검토
- o 레코드의 크기
- o 파일과 구조 저장을 위한 최소한의 공간
- o 빈번한 질의와 트랜잭션들의 수행 속도를 높이기 위한 고려 사항
- 응답 시간
- 저장 공간의 효율화

#### - 트랜잭션 처리량

[해설작성자 : 용소용소]

스키마의 평가 및 정제는 논리적 설계에서 실행된다.

[해설작성자 : 망굴]

### 42. DELETE 명령에 대한 설명으로 틀린 것은?

- ① 테이블의 행을 삭제할 때 사용한다.
- ② WHERE 조건절이 없는 DELETE 명령을 수행하면 DROP TABLE 명령을 수행했을 때와 동일한 효과를 얻을 수 있다.
- ③ SQL을 사용 용도에 따라 분류할 경우 DML에 해당한다.
- ④ 기본 사용 형식은 “DELETE FROM 테이블 [WHERE 조건];” 이다.

#### <문제 해설>

DELETE문은 테이블내의 튜플들만 삭제, DROP문은 테이블 자체를 삭제

[해설작성자 : ITK]

### 43. 어떤 릴레이션 R의 모든 조인 종속성의 만족이 R의 후보키를 통해서만 만족될 때, 이 릴레이션 R이 해당하는 정규형은?

- ① 제5정규형                      ② 제4정규형
- ③ 제3정규형                      ④ 제1정규형

#### <문제 해설>

- 제1정규형 : 릴레이션에 속한 모든 속성의 도메인이 원자값으로만 구성되어 있으면 제 1 정규형에 속한다.

- 제2정규형 : 제1 정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 완전 함수 종속되면 제2 정규형에 속한다.(부분 함수 종속 제거)

- 제3정규형 : 제2 정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 이행적 함수 종속이 되지 않으면 제3 정규형에 속한다.





- BCNF(보이스/코드) 정규형 : 릴레이션의 함수 종속 관계에서 모든 결정자가 후보키이면 보이스/코드 정규형에 속한다.

- 제4정규형 : BCNF 정규형을 만족하면서 함수 종속이 아닌 다치 종속을 제거해야 만족할 수 있다.

- 제5정규형 : 제4 정규형을 만족하면서 후보키를 통하지 않는 조인 종속을 제거해야 만족할 수 있다.

[해설작성자 : 강썸]

### 44. E-R 모델에서 다중값 속성의 표기법은?

- ① 
- ② 
- ③ 
- ④ 

#### <문제 해설>

마름모 : 관계

사각형 : 개체

선 : 관계-속성 연결

[해설작성자 : comcbt.com 이용자]

### 45. 다른 릴레이션의 기본키를 참조하는 키를 의미하는 것은?

- ① 필드키                      ② 슈퍼키
- ③ 외래키                      ④ 후보키

#### <문제 해설>



\* 키 : 레코드 또는 튜플을 확인하기 위해 이용되는 속성 값

#### 1. 기본키

- \* 유일성과 최소성을 만족하는 속성을 가지는 후보키 중에 지정된 하나의 키
- \* 중복되는 값, Null 값을 가질 수 없음.
- \* 값의 변화가 거의 없고 단순한 후보키가 기본키로 적합.

#### 2. 슈퍼키

- \* 데이터베이스에서 테이블의 행을 고유하게 식별할 수 있는 속성 or 속성의 집합

#### 3. 외래키

- \* 관계형 데이터베이스에서 한 테이블 속성 집합이 다른 테이블의 기본키가 됨.
- \* 데이터베이스 내에 존재하는 테이블들의 관계에서 참조의 무결성을 보장하기 위함임
- \* 중복되는 값, 비어있는 값(Null) 가질 수 있음(<->기본키)

#### 4. 후보키

- \* 유일성과 최소성을 만족하는 속성 or 속성들의 집합
  - \* 슈퍼키 중 최소성을 만족하는 것이 후보키가 됨.
  - \* 후보키(candidate key)는 기본키(primary key)로 사용 할 수 있으며, 후보키가 여러개일 경우에는 하나를 지정해 사용.
  - (지정되지 않은 나머지 후보키 = 대체 키(alternate key))
- [해설작성자 : 센딘]

#### 46. 관계해석에서 '모든 것에 대하여'의 의미를 나타내는 논리 기호는?

- ①  $\exists$                       ②  $\in$   
 ③  $\forall$                       ④  $\subset$

<문제 해설>

A의 'A'를 뒤집어놓은 상태  
 [해설작성자 : 웅실웅실]

- \*  $\exists$ : 존재한다(There exist)
  - \*  $\in$ : t가 r에 속함(  $t \in r$  )
  - \*  $\forall$ : 모든 것에 대하여(for all)
  - \*  $\cup$ : 합집합
- [해설작성자 : 센딘]

#### 47. 다음 릴레이션의 Degree와 Cardinality는?

학번	이름	학년	학과
13001	홍길동	3학년	전 기
13002	미순신	4학년	기 계
13003	강감찬	2학년	컴퓨터

- ① Degree : 4, Cardinality : 3  
 ② Degree : 3, Cardinality : 4  
 ③ Degree : 3, Cardinality : 12  
 ④ Degree : 12, Cardinality : 3

<문제 해설>

카디널리티=행  
 디그리(차수)=열  
 '카행', '디열(차열)'로 암기  
 [해설작성자 : 마산 양덕동 내년 사심]

\* 열차 = 열+차수(degree)

\* 비행기 = (비)+행+기수(cardinality)

[해설작성자 : 센딘]

#### 48. 뷰(View)에 대한 설명으로 틀린 것은?

- ① 뷰 위에 또 다른 뷰를 정의할 수 있다.  
 ② DBA는 보안성 측면에서 뷰를 활용할 수 있다.  
 ③ 사용자가 필요한 정보를 요구에 맞게 가공하여 뷰로 만들 수 있다.  
 ④ SQL을 사용하면 뷰에 대한 삽입, 갱신, 삭제 연산 시 제약 사항이 없다.

<문제 해설>

뷰는 삽입, 갱신, 삭제 연산 시 제약 사항이 있음  
 [해설작성자 : 1]

#### 49. 관계 대수식을 SQL 질의로 옳게 표현한 것은?

$\Pi_{이름}(\sigma_{학과='교육'}(학생))$

- ① SELECT 학생 FROM 이름 WHERE 학과='교육';  
 ② SELECT 이름 FROM 학생 WHERE 학과='교육';  
 ③ SELECT 교육 FROM 학과 WHERE 이름='학생';  
 ④ SELECT 학과 FROM 학생 WHERE 이름='교육';

<문제 해설>

거꾸로 풀면됨 문제에서 학생이 가장 뒤에 있음 - 맨 앞에 생각하는 from문에 삽입 from 학생 그럼 2 or 4 문제에서 학과 = 교육 이니까 2번 답  
 [해설작성자 : 쉽게생각해]

#### 50. 정규화 과정에서 함수 종속이 $A \rightarrow B$ 이고 $B \rightarrow C$ 일 때 $A \rightarrow C$ 인 관계를 제거하는 단계는?

- ① 1NF  $\rightarrow$  2NF                      ② 2NF  $\rightarrow$  3NF  
 ③ 3NF  $\rightarrow$  BCNF                      ④ BCNF  $\rightarrow$  4NF

<문제 해설>

$A \rightarrow B$  이고  $B \rightarrow C$  일 때  $A \rightarrow C$ 를 만족하는 관계를 다른 말로 말하면, 이행적 종속(Transitive Dependency) 관계입니다. 이행적 함수 종속을 제거한다는 뜻은, 2NF(2정규형)에서 3NF(3정규형)가 되는 과정이라고도 볼 수 있겠죠.  
 [해설작성자 : 한달남음]

#### 51. CREATE TABLE문에 포함되지 않는 기능은?

- ① 속성 타입 변경  
 ② 속성의 NOT NULL 여부 지정  
 ③ 기본키를 구성하는 속성 지정  
 ④ CHECK 제약조건의 정의

<문제 해설>

PRIMARY KEY : 테이블의 기본 키를 정의 / 유일하게 테이블의 각 행을 식별  
 FOREIGN KEY : 참조 대상을 테이블로 명시 / 외래 키를 정의 / 열과 참조된 테이블의 열 사이의 외래 키 관계를 적용하고 설정  
 UNIQUE : 테이블 내에서 얻은 유일한 값을 갖도록 하는 속성  
 NOT NULL : 해당 컬럼은 NULL값을 포함하지 않도록 하는 속성  
 CHECK : 개발자가 정의하는 제약조건 / 참(TRUE)이어야 하는 조건을 지정  
 DEFAULT : 해당 필드의 기본값을 설정

2~4번은 해당이 된다..1번은 속성타입 변경이라는 것은  
CREATE TABLE에서 사용하는 것중에는 없다  
[해설작성자 : JSL연수 53기]

## 52. SQL과 관련한 설명으로 틀린 것은?

- ① REVOKE 키워드를 사용하여 열 이름을 다시 부여할 수 있다.
- ② 데이터 정의어는 기본 테이블, 뷰 테이블, 또는 인덱스 등을 생성, 변경, 제거하는데 사용되는 명령어이다.
- ③ DISTINCT를 활용하여 중복 값을 제거할 수 있다.
- ④ JOIN을 통해 여러 테이블의 레코드를 조합하여 표현할 수 있다.

<문제 해설>

REVOKE는 DCL에 해당 하는 권한을 해제하는 명령어  
[해설작성자 : 그렇다네요]

#### DDL(Data Define Language, 데이터 정의어)

>DDL은 SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의하거나 변경 또는 삭제할 때 사용하는 언어이다.

- CREATE: SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의한다.
- ALTER: TABLE에 대한 정의를 변경하는 데 사용한다.
- DROP: SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 삭제한다.

#### DML(Data Manipulation Language, 데이터 조작어)

>DML은 데이터베이스 사용자가 응용 프로그램이나 질의어를 통하여 저장된 데이터를 실질적으로 처리하는데 사용되는 언어이다.

데이터베이스 사용자와 데이터베이스 관리시스템 간의 인터페이스를 제공한다.

- SELECT: 테이블에서 조건에 맞는 튜플을 검색한다.
- INSERT: 테이블에 새로운 튜플을 삽입한다.
- DELETE: 테이블에서 조건에 맞는 튜플을 삭제한다.
- UPDATE: 테이블에서 조건에 맞는 튜플의 내용을 변경한다.

#### DCL(Data Control Language, 데이터 제어어)

> DCL은 데이터의 보안, 무결성, 회복, 병행 수행 제어 등을 정의하는 데 사용되는 언어이다.  
데이터 베이스 관리자가 데이터 관리를 목적으로 사용한다.

- COMMIT: 명령에 의해 수행된 결과를 실제 물리적 디스크로 저장하고, 데이터베이스 조작 작업이 정상적으로 완료되었음을 관리자에게 알려준다.
  - ROLLBACK: 데이터베이스 조작 작업이 비정상적으로 종료되었을 때 원래의 상태로 복구한다.
  - GRANT: 데이터베이스 사용자에게 사용권한을 부여한다.
  - REVOKE: 데이터베이스 사용자의 권한을 취소한다.
- [해설작성자 : bboygg]

## 53. 다음 SQL문의 실행결과로 생성되는 튜플 수는?

SELECT 급여 FROM 사원;

<사원> 테이블

사원ID	사원명	급여	부서ID
101	박철수	30000	1
102	한나라	35000	2
103	김강동	40000	3
104	이구수	35000	2
105	최초록	40000	3

- ① 1
- ② 3
- ③ 4
- ④ 5

<문제 해설>

SQL문은 : SELECT(컬럼 명) FROM (테이블 명)

SQL의 조건이 없으므로 모든 급여가 생성된다..따라서 5개의 튜플이 생성됨.

[해설작성자 : 독학하는자]

## 54. 다음 SQL문에서 사용된 BETWEEN 연산의 의미와 동일한 것은?

SELECT \*  
FROM 성적  
WHERE (점수 BETWEEN 90 AND 95)  
AND 학과 = '컴퓨터공학과';

- ① 점수 >= 90 AND 점수 <= 95
- ② 점수 > 90 AND 점수 < 95
- ③ 점수 > 90 AND 점수 <= 95
- ④ 점수 >= 90 AND 점수 < 95

<문제 해설>

SQL문의 조건은 성적 테이블에 성적(점수)가 90이상 95이하인 학과를 찾아라 이다.

[해설작성자 : 독학하는자]

## 55. 트랜잭션의 상태 중 트랜잭션의 수행이 실패하여 Rollback 연산을 실행한 상태는?

- ① 철회(Aborted)
- ② 부분 완료(Partially Committed)
- ③ 완료(Commit)
- ④ 실패(Fail)

<문제 해설>

ROLLBACK = 데이터베이스 조작 작업이 비정상적으로 종료되었을 때 원래의 상태로 복구 = 철회

[해설작성자 : comcbt.com 이용자]

## 56. 데이터 제어어(DCL)에 대한 설명으로 옳은 것은?

- ① ROLLBACK : 데이터의 보안과 무결성을 정의한다.
- ② COMMIT : 데이터베이스 사용자의 사용 권한을 취소한다.
- ③ GRANT : 데이터베이스 사용자의 사용 권한을 부여한다.
- ④ REVOKE : 데이터베이스 조작 작업이 비정상적으로 종료되었을 때 원래 상태로 복구한다.

<문제 해설>

1. ROLLBACK : 데이터베이스 조작 작업이 비정상적으로 종료되었을 때 원래 상태로 복구한다.
2. COMMIT : 데이터의 보안과 무결성을 정의한다.
3. GRANT : 데이터베이스 사용자의 사용 권한을 부여한다.
4. REVOKE : 데이터베이스 사용자의 사용 권한을 취소한다.

[해설작성자 : 독학하는자]

57. 테이블 R과 S에 대한 SQL에 대한 SQL문이 실행되었을 때, 실행결과로 옳은 것은?

R		S		SQL
A	B	A	B	
1	A	1	A	SELECT A FROM R UNION ALL SELECT A FROM S;
3	B	2	B	

①	1	②	3 2
③	1 3	④	1 3 1 2

<문제 해설>

SQL문의 : SELECT A FROM R 은 R테이블의 A컬럼의 값을 가져오라 라는 뜻  
 UNION ALL : 각 쿼리의 모든 결과를 포함한 합집합 (중복 제거 안함)

[해설작성자 : 독학하는자]

58. 분산 데이터베이스 시스템(Distributed Database System)에 대한 설명으로 틀린 것은?

- ① 분산 데이터베이스는 논리적으로는 하나의 시스템에 속하지만 물리적으로는 여러 개의 컴퓨터 사이트에 분산되어 있다.
- ② 위치 투명성, 중복 투명성, 병행 투명성, 장애 투명성을 목표로 한다.
- ③ 데이터베이스의 설계가 비교적 어렵고, 개발 비용과 처리 비용이 증가한다는 단점이 있다.
- ④ 분산 데이터베이스 시스템의 주요 구성 요소는 분산 처리기, P2P 시스템, 단일 데이터베이스 등이 있다.

<문제 해설>

분산 데이터베이스 주요 구성 요소는 전역, 분할, 할당, 지역 스키마가 있다.

[해설작성자 : 합격기원]

구성요소는 분산처리기, 분산 데이터베이스, 통신 네트워크  
 [해설작성자 : ssub]

분산 데이터베이스의 구성 요소 - '분산'

- 1) 분산 처리기
- 2) 분산 데이터베이스
- 3) 통신 네트워크
- 4) 분산 트랜잭션

분산 데이터베이스의 구조 - 전역, 분할(단편화), 할당, 지역 스키마

[해설작성자 : 한번에 붙고싶어]

59. 테이블 두 개를 조인하여 뷰 V\_1을 정의하고, V\_1을 이용하여 뷰 V\_2를 정의하였다. 다음 명령 수행 후 결과로 옳은 것은?

DROP VIEW V\_1 CASCADE;

- ① V\_1만 삭제된다.
- ② V\_2만 삭제된다.
- ③ V\_1과 V\_2 모두 삭제된다.
- ④ V\_1과 V\_2 모두 삭제되지 않는다.

<문제 해설>

CASCADE

- 데이터베이스 내의 임의의 다른 테이블과 PRIMARY KEY 또는 FOREIGN KEY로서의 관계가 여전히 존재하는 경우 PRIMARY KEY가 존재하는 테이블을 마음대로 제거 및 비활성화 할 수 없다.
- 즉, 부모 테이블과 자식테이블 간에 참조 설정이 되어 있을 때 부모 테이블의 제약 조건을 비활성화 시키면서 이를 참조하고 있는 자식 테이블의 제약 조건까지 함께 비활성화시키기 위해 사용한다.
- 또한 부모 테이블만 삭제하고 싶을 경우, 원래는 자식테이블을 먼저 삭제하고 부모를 삭제 한 뒤 자식테이블을 다시 만들어야하는 번거로움이 따른다.
- 하지만 CASCADE를 쓰면 일시적으로 참조관계를 끊을 수 있다.

[해설작성자 : 강썸]

60. 데이터베이스에서 병행제어의 목적으로 틀린 것은?

- ① 시스템 활용도 최대화
- ② 사용자에게 응답시간 최소화
- ③ 데이터베이스 공유 최소화
- ④ 데이터베이스 일관성 유지

<문제 해설>

병행제어의 목적

- 여러 사용자들의 데이터베이스 공동 사용을 최대화
- 사용자의 응답 시간 최소화
- 데이터베이스 시스템의 활용도 최대화
- 데이터베이스의 일관성 유지

[해설작성자 : 강썸]

4과목 : 프로그래밍 언어 활용

61. IP 주소체계와 관련한 설명으로 틀린 것은?

- ① IPv6의 패킷 헤더는 32 octet의 고정된 길이를 가진다.
- ② IPv6는 주소 자동설정(Auto Configuration) 기능을 통해 손쉽게 이용자의 단말을 네트워크에 접속시킬 수 있다.
- ③ IPv4는 호스트 주소를 자동으로 설정하며 유니캐스트(Unicast)를 지원한다.
- ④ IPv4는 클래스별로 네트워크와 호스트 주소의 길이가 다르다.

<문제 해설>

IPv4와 IPv6의 차이는 정보처리기사 뿐만이 아니라 대부분의 IT관련 시험에서 다루고 있음  
 주로 다루는 차이점과 핵심 단어는 아래와 같음

- 1) IPv4  
 ㄱ. 32비트 주소  
 ㄴ. 유니캐스트/멀티캐스트/브로드캐스트 사용  
 2) IPv6  
 ㄱ. 128비트 주소  
 ㄴ. 기존 IPv4의 주소 부족 문제를 해결하기 개발  
 ㄷ. 인증성/기밀성/무결성 지원 (=보안성 강화)  
 ㄹ. 유니캐스트, 애니캐스트, 멀티캐스트 사용  
 [해설작성자 : 단기합격은 유튜브 윤파고]

62. 다음 C언어 프로그램이 실행되었을 때, 실행 결과는?

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
    int arr[2][3]={1,2,3,4,5,6}
    int (*p)[3]=NULL;
    p=arr;
    printf("%d, ", *(p[0]+1) + *(p[1]+2));
    printf("%d", (*(p+1)+0) + (*(p+1)+1));
    return 0;
}
```

- ① 7, 5                      ② 8, 5  
 ③ 8, 9                      ④ 7, 9

<문제 해설>

p[0]=arr[0] -> p[0]+1=arr[0][1]=2  
 p[1]=arr[1] -> p[1]+2=arr[1][2]=6 2+6=8

p=arr[0]의 주소 -> p+1=arr[1] ->  
 (\*(p+1)+0)=arr[1][0]=4  
 위와 같은 계산으로 (\*(p+1)+1)=arr[1][1]=5 4+5=9  
 [해설작성자 : 강썸]

63. OSI 7계층 중 데이터링크 계층에 해당되는 프로토콜이 아닌 것은?

- ① HTTP                      ② HDLC  
 ③ PPP                      ④ LLC

<문제 해설>

1. 응용계층

[해설작성자 : comcbt.com 이용자]

64. C언어에서 두 개의 논리 값 중 하나라도 참이면 1을, 모두 거짓이면 0을 반환하는 연산자는?

- ① ||                      ② &&  
 ③ \*\*                      ④ !=

<문제 해설>

1) || : OR 연산 - 둘 중 하나라도 참이면 1  
 2) && : AND 연산 - 둘 다 참이어야 1  
 3) \*\* : 거듭제곱  
 4) != : 부등(피연산자가 서로 다를 시 true)  
 [해설작성자 : 단기합격은 유튜브 윤파고]

65. IPv6에 대한 특성으로 틀린 것은?

- ① 표시방법은 8비트씩 4부분의 16진수로 표시한다.  
 ② 2<sup>128</sup>개의 주소를 표현할 수 있다.

- ③ 등급별, 서비스별로 패킷을 구분할 수 있어 품질보장이 용이하다.  
 ④ 확장기능을 통해 보안기능을 제공한다.

<문제 해설>

IPv6 : 16비트씩 8부분의 16진수로 표시  
 [해설작성자 : 정처기 공부]

66. JAVA의 예외(exception)와 관련한 설명으로 틀린 것은?

- ① 문법 오류로 인해 발생한 것  
 ② 오동작이나 결과에 악영향을 미칠 수 있는 실행 시간 동안에 발생한 오류  
 ③ 배열의 인덱스가 그 범위를 넘어서는 경우 발생하는 오류  
 ④ 존재하지 않는 파일을 읽으려고 하는 경우에 발생하는 오류

<문제 해설>

문법 오류는 컴파일 시 에러가 발생하기 때문에 프로그램의 정상적인 실행이 불가능하므로 자바에서 정의한 예외 조건에 부합하지 않는다.  
 [해설작성자 : 김라마]

67. TCP/IP 계층 구조에서 IP의 동작 과정에서의 전송 오류가 발생하는 경우에 대비해 오류 정보를 전송하는 목적으로 사용하는 프로토콜은?

- ① ECP(Error Checking Protocol)  
 ② ARP(Address Resolution Protocol)  
 ③ ICMP(Internet Control Message Protocol)  
 ④ PPP(Point-to-Point Protocol)

<문제 해설>

ARP : IP 네트워크 상에서 IP주소를 MAC주소로 변환하는 프로토콜  
 [해설작성자 : 2트를 노려야하나..]

68. 좋은 소프트웨어 설계를 위한 소프트웨어의 모듈간의 결합도(Coupling)와 모듈 내 요소 간 응집도(Cohesion)에 대한 설명으로 옳은 것은?

- ① 응집도는 낮게 결합도는 높게 설계한다.  
 ② 응집도는 높게 결합도는 낮게 설계한다.  
 ③ 양쪽 모두 낮게 설계한다.  
 ④ 양쪽 모두 높게 설계한다.

<문제 해설>

좋은 소프트웨어일수록 결합도는 낮고, 응집도는 높다  
 결합도 : 어떤 모듈이 다른 모듈에 의존하는 정도  
 응집도 : 한 모듈 내부의 처리 요소들이 서로 관련되어 있는 정도(독립적 기능 수행 정도)  
 코딩할 때 모듈을 들자면 모듈을 만들었는데 이 코드 저 코드 엮어서 거미줄 코딩을 한 경우 서로에 대한 의존도가 높으므로 결합도가 높다고 할 수 있고, 하나의 모듈을 만들었는데 독립적 기능을 수행하며 타 코드와 큰 연관성이 없을 경우 응집도가 높다고 할 수 있다..(한마디로 독립적인 모듈이나 의존적인 모듈이냐!)  
 응집도 결합도는 필기/실기 엄청 자주 출제되므로 꼭 잘 모르겠으면, 구글링해서 꼭 제대로 알고 넘어가도록 합시다.  
 [해설작성자 : 단기합격은 유튜브 윤파고]

69. 다음과 같은 형태로 임계 구역의 접근을 제어하는 상호배제 기법은?

```
P(S) : while S <= 0 do skip;  
S := S - 1;  
V(S) : S := S + 1;
```

- ① Dekker Algorithm    ② Lamport Algorithm  
③ Peterson Algorithm    ④ Semaphore

<문제 해설>

1. 데커의 알고리즘

- 프로세스가 두개일때 상호 배제를 보장하는 최초의 알고리즘

- flag와 turn 변수를 사용하여 조정

2. 램퍼드 알고리즘

- 프로세스 n개의 상호 배제 문제를 해결한 알고리즘

- 프로세스에게 고유한 번호를 부여하고, 번호를 기준으로 우선순위를 정하여 우선순위가 높은 프로세스가 먼저 임계 구역에 진입하도록 구현.

3. 피터슨 알고리즘

- 프로세스가 두개일때 상호 배제를 보장, 데커의 알고리즘과 유사하지만 상대방에게 진입 기회를 양보한다는 차이가 있고 보다 더 간단하게 구현됨

4. 세마포어

- 공유된 자원의 데이터 혹은 임계영역 등에 따라 여러 Process 혹은 Thread가 접근하는 것을 막아줌(동기화 대상이 하나 이상)

[해설작성자 : 강썸]

70. 소프트웨어 개발에서 모듈(Module)이 되기 위한 주요 특징에 해당하지 않는 것은?

- ① 다른 것들과 구별될 수 있는 독립적인 기능을 가진 단위(Unit)이다.  
② 독립적인 컴파일이 가능하다.  
③ 유일한 이름을 가져야 한다.  
④ 다른 모듈에서의 접근이 불가능해야 한다.

<문제 해설>

다른 모듈에서의 접근이 불가능한 것은 폐쇄적인 것.

모듈은 폐쇄적이거나 접근이 불가능하게 설계할 필요는 없음.

[해설작성자 : 내일 시험보는 진효식]

71. 빈 기억공간의 크기가 20KB, 16KB, 8KB, 40KB 일 때 기억장치 배치 전략으로 "Best Fit"을 사용하여 17KB의 프로그램을 적재할 경우 내부단편화의 크기는 얼마인가?

- ① 3KB                      ② 23KB  
③ 64KB                    ④ 67KB

<문제 해설>

문제에서 Best Fit을 물을 시 내부단편화가 제일 작은 보기를 찾으면 됨

1) 20KB - 17KB = 3KB

2) 16KB - 17KB = 불가

3) 8KB - 17KB = 불가

4) 40KB - 17KB = 23KB

따라서 1번이 정답

[해설작성자 : 단기합격은 유튜브 윤파고]

72. 다음 C언어프로그램이 실행되었을 때, 실행 결과는?

```
#include <stdio.h>  
#include <stdlib.h>  
int main(int argc, char *argv[]) {  
    int i = 0  
    while(1){  
        if(i==4){  
            break;  
        }  
        ++i;  
    }  
    printf("i = %d", i);  
    return 0;  
}
```

- ① i = 0                      ② i = 1  
③ i = 3                      ④ i = 4

<문제 해설>

int i = 0;    정수형 0을 선언한다

while(1) { . 반복문 1번 실행

if(i==4){ 만약 i가 4랑 같다고 하면)

break; 브레이크문 걸어서 나감

}

++;    그게 아니라면 i를 1씩 증가 시켜라

즉 위에서 반복문 1번 돌리고 조건이 4랑 같을때까지 돌리라고 했으니깐 0 ~4번까지 돌린후 4=4 같으니 탈출 그래서 4

}

printf("i = %d", i);

return 0;

[해설작성자 : 22년1회차 필기 합격생]

while에서 조건식 대신 1을 지정하면 무한히 반복하는 루프가 만들어집니다..i가 4가 되면 break로 while문을 빠져나오게 되면서 i는 4가 됩니다.

73. 다음 JAVA 프로그램이 실행되었을 때, 실행 결과는?



```
public class Ape{
    static void rs(char a[]) {
        for(int i = 0; i < a.length; i++)
            if(a[i] == 'B')
                a[i] = 'C';
            else if(i == a.length - 1)
                a[i] = a[i-1];
            else a[i] = a[i+1];
    }

    static void pca(char a[]) {
        for(int i = 0; i < a.length; i++)
            System.out.print(a[i]);
            System.out.println();
    }

    public static void main(String[] args) {
        char c[] = {'A', 'B', 'D', 'D', 'A', 'B', 'C'};
        rs(c);
        pca(c);
    }
}
```

- ① BCDABCA                      ② BCDABCC
- ③ CDDACCC                      ④ CDDACCA

<문제 해설>

rs(c)는 대충 a=c라고 생각하고 rs(char a[])=rs(char c[])  
 c의 문자열 크기는 7이므로 a.length=7

rs()는 i가 7미만일때까지 반복한다는 의미  
 {'A', 'B', 'D', 'D', 'A', 'B', 'C'} 에서 a[0]='A', a[1]='B'  
 이런식으로 이해하면되는데  
 rs 첫번째 if문은 a[i]의 값이'B'인 경우 그 값을 B에서 C  
 로 바꾸라는 의미이고  
 rs 두번째 else if는 i=6일때 a[6]=a[5]로 바꾸라는 의미이고  
 rs 세번째 else 는 1, 2번째 if문에 충족되지 않는 나머지  
 경우는 다 a[i]=a[i+1]을 해주라는 의미

{'A', 'B', 'D', 'D', 'A', 'B', 'C'}에서  
 i=0 이면 A[0]를 A[1]로 변경해서 {'B', 'B', 'D', 'D', 'A',  
 'B', 'C'}  
 i=1 이면 값이 B이기 때문에 C로 변경 {'B', 'C', 'D', 'D',  
 'A', 'B', 'C'}  
 i=2 이면 A[2]를 A[3]으로 변경 {'B', 'C', 'D', 'D', 'A',  
 'B', 'C'} -똑같아서 그대로 방치  
 i=3 이면 A[3]를 A[4]로 변경 {'B', 'C', 'D', 'A', 'A', 'B',  
 'C'}  
 i=4 이면 A[4]를 A[5]로 변경 {'B', 'C', 'D', 'A', 'B', 'B',  
 'C'}  
 i=5 이면 값이 B이기 때문에 C로 변경 {'B', 'C', 'D', 'A',  
 'B', 'C', 'C'}  
 i=6 이면 A[6]를 A[5]로 변경 {'B', 'C', 'D', 'A', 'B', 'C',  
 'C'}  
 pca(c)로 와서 char a[] 그대로 출력

정답은 {'B', 'C', 'D', 'A', 'B', 'C', 'C'} = 2번  
 [해설작성자 : 영남이공대 소프트웨어콘텐츠과]

1. 치환하는 문제임으로 a[i] == 'B' 일때 C로 치환하는  
 부분이 2번째 C를 확인.
  2. 첫번째 치환값이 얼마인지만 확인하면 금방 풀립니  
 다..else부분에 a[0] = a[1] 값이므로 B
  3. 따라서 B로 시작하고 2번째 6번째가 C인걸 찾으면 금  
 방 풀립니다
- [해설작성자 : 벼락치기 TTT]

74. 개발 환경 구성을 위한 빌드(Build) 도구에 해당하지 않는  
 것은?

- ① Ant                                      ② Kerberos
- ③ Maven                                      ④ Gradle

<문제 해설>

Build = 작성된 Source Code를 실제 기기에서 실행 할 수  
 있는 소프트웨어로 변환하기 위한 과정  
 1) Ant = 아파치 재단에서 개발한 자바의 공식적인 빌드 도구  
 2) Kerberos = 컴퓨터 네트워크 인증 암호화 프로토콜로  
 3) Maven = 아파치 재단에서 개발, Ant 대안으로 개발되었음  
 4) Gradle = Ant, Maven 의 보완으로 개발된 빌드 도구(안드  
 로이드 스튜디오 주 빌드 도구)  
 [해설작성자 : 단기합격은 유튜브 윤파고]

75. 3개의 페이지 프레임 을 갖는 시스템에서 페이지 참조 순  
 서가 1, 2, 1, 0, 4, 1, 3 일 경우 FIFO 알고리즘에 의한  
 페이지 교체 의 경우 프레임의 최종 상태는?

- ① 1, 2, 0                                      ② 2, 4, 3
- ③ 1, 4, 2                                      ④ 4, 1, 3

<문제 해설>

[1]      (1)      [4]  
          [2]                      [1]  
                                  [0]      [3]  
 [해설작성자 : 정처기 공부]

1  
 12  
 1 //  
 120  
 204  
 041  
 413 최종  
 [해설작성자 : 오하요하이]

76. 다음 C언어 프로그램이 실행되었을 때, 실행 결과는?

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
    char str1[20] = "KOREA";
    char str2[20] = "LOVE";
    char* p1=NULL;
    char* p2=NULL;
    p1=str1;
    p2=str2;
    str1[1]=p2[2];
    str2[3]=p1[4];
    strcat(str1, str2);
    printf("%c", *(p1+2));
    return 0;
}
```

- ① E                      ② V  
 ③ R                      ④ O

<문제 해설>  
 str1[1]=str2[2] => str1의 KOREA중 O가 V로 변경  
 str1= KVREA  
 str2[3]=str1[4] =>str2의 LOVE중 E가 A로 변경  
 str2=LOVA

p1+2= str1[2]= R  
 [해설작성자 : 영남이공대 소프트웨어콘텐츠과]

77. 다음 Python 프로그램이 실행되었을 때, 실행 결과는?

```
a=100
list_data = ['a', 'b', 'c']
dict_data = {'a':90, 'b':95}
print(list_data[0])
print(dict_data['a'])
```

- ① 

a  
90

                      ② 

100  
90

  
 ③ 

100  
100

                      ④ 

a  
a

<문제 해설>  
 list\_data에 a가 아니라 'a'이기 때문에 문자 그대로 a로 출력  
 되고  
 dict\_data는 90으로 출력됨  
 [해설작성자 : 영남이공대 소프트웨어콘텐츠과]

78. C언어에서 정수 변수 a, b에 각각 1, 2가 저장되어 있을  
 때 다음 식의 연산 결과로 옳은 것은?

```
a < b + 2 && a << 1 <= b
```

- ① 0                      ② 1  
 ③ 3                      ④ 5

<문제 해설>  
 연산자 우선순위 문제  
 a < (b+2) -> true  
 a << 1 = 00000001 (1) 에서 좌측시프트 00000010 (2)/  
 2 <= b = true  
 && - and 연산은 둘다 참일경우 1을 반환  
 [해설작성자 : 1트에 합격했지롱]

79. 다음 Python 프로그램이 실행되었을 때, 실행 결과는?

```
a = ["대", "한", "민", "국"]
for i in a:
    print(i)
```

- ① 

대한민국

                      ② 

대한  
민국

  
 ③ 

대

                      ④ 

대대대대

<문제 해설>  
 for (반복변수) in [리스트]  
 결국 리스트에 있는 값 인덱스 기반으로 하나 씩 출력  
 [해설작성자 : 정처기 공부]

파이썬은 print()하면 자동으로 개행된다  
 [해설작성자 : 영국인]

80. UNIX 시스템의 셸(shell)의 주요 기능에 대한 설명이 아닌  
 것은?

- ① 사용자 명령을 해석하고 커널로 전달하는 기능을 제공  
 한다.  
 ② 반복적인 명령 프로그램을 만드는 프로그래밍 기능을  
 제공한다.  
 ③ 셸 프로그램 실행을 위해 프로세스와 메모리를 관리한  
 다.  
 ④ 초기화 파일을 이용해 사용자 환경을 설정하는 기능을  
 제공한다.

<문제 해설>  
 프로세스, 메모리 관리는 커널의 기능  
 [해설작성자 : 합격기원]

#### 5과목 : 정보시스템 구축관리

81. 소프트웨어 생명주기 모델 중 나선형 모델(Spiral Model)  
 과 관련한 설명으로 틀린 것은??

- ① 소프트웨어 개발 프로세스를 위험 관리(Risk  
 Management) 측면에서 본 모델이다.  
 ② 위험 분석(Risk Analysis)은 반복적인 개발 진행 후 주  
 기의 마지막 단계에서 최종적으로 한 번 수행해야 한  
 다.  
 ③ 시스템을 여러 부분으로 나누어 여러 번의 개발 주기  
 를 거치면서 시스템이 완성된다.

- ④ 요구사항이나 아키텍처를 이해하기 어렵다거나 중심이 되는 기술에 문제가 있는 경우 적합한 모델이다.

<문제 해설>

점진적으로 개발 과정이 반복되므로 위험분석 또한 계획 수립 이후 추가적인 위험 분석이 가능함  
 [해설작성자 : 김라마]

82. 정보시스템과 관련한 다음 설명에 해당하는 것은?

- 각 시스템 간에 공유 디스크를 중심으로 클러스터링으로 묶여 다수의 시스템을 동시에 연결할 수 있다.
- 조직, 기업의 기간 업무 서버 등의 안정성을 높이기 위해 사용될 수 있다.
- 여러 가지 방식으로 구현되며 2개의 서버를 연결하는 것으로 2개의 시스템이 각각 업무를 수행하도록 구현하는 방식이 널리 사용된다.

- ① 고가용성 솔루션(HACMP)  
 ② 점대점 연결 방식(Point-to-Point Mode)  
 ③ 스텝스넷(Stuxnet)  
 ④ 루팅(Rooting)

<문제 해설>

고가용성 솔루션(HACMP:High Availability Cluster Multi Processing)

- AIX를 기반으로 한 IBM의 High Availability Solution
- Resource의 중복 또는 공유를 통해 Application의 보호를 가능하게 해줌
- 같은 Data를 공유하거나 동시에 access하는 node들에서 여러 개의 application을 실행하게 해줌
- 두대 이상의 시스템을 하나의 Cluster로 묶어 Cluster내의 한 시스템에서 장애가 발생할 경우 다른 시스템이 장애가 발생한 시스템의 자원을 인수할 수 있도록 하여 서비스의 중단을 최소화 할 수 있도록 도와주는 솔루션

점대점 연결 방식(Point-to-Point Mode)

- 네트워크에 있어 물리적으로는 중개 장치를 통과하지 않고 한 지점에서 다른 지점으로 직접 가는 채널
- 두 스테이션간을 별도의 회선을 사용하여 1 대 1로 연결.
- 전용회선이나 공중 전화 회선을 이용.
- 회선 구성이 간단하고 대용량 전송에 유리.
- 별도의 회선과 포트에 따른 높은 설치비용

스텝스넷(Stuxnet)

- 2010년 6월에 발견된 웜 바이러스
- 윈도우를 통해 감염, 지멘스산업의 SW 및 장비를 공격

루팅(Rooting)

- 모바일 기기에서 구동되는 안드로이드 운영체제상에서 최상위 권한 (루트 권한)을 얻음으로 해당 기기의 생산자 또는 판매자 측에서 걸어 놓은 제약을 해제하는 행위
- [해설작성자 : 강썸]

83. 위조된 매체 접근 제어(MAC) 주소를 지속적으로 네트워크로 흘려보내, 스위치 MAC 주소 테이블의 저장 기능을 혼란시켜 더미 허브(Dummy Hub)처럼 작동하게 하는 공격은?

- ① Parsing                      ② LAN Tapping  
 ③ Switch Jamming          ④ FTP Flooding

<문제 해설>

- Parsing

- 하나의 프로그램을 런타임 환경(예를 들면, 브라우저 내 자바스크립트 엔진)이 실제로 실행할 수 있는 내부 포맷으로 분석하고 변환하는 것을 의미함

- LAN Tapping

- 처음 들어보는 용어이고, 찾아도 제대로 된 정의가 나오지 않으나 Lan+Tapping으로 해석한다면 LAN신호를 직접 자신에게 끌어오는 방식의 공격정도로 해석 가능함

- Switch Jamming

- 스위치의 기능이 방해 받아 정상 동작을 하지 못해 스위치가 더미 허브처럼 작동 하게 되는 것
- Switch + Jamming(방해)

- FTP(SYN) Flooding

- TCP의 3 Way Handshake 취약점을 이용한 DoS 공격으로 다량의 SYN패킷을 보내 백로그를 가득 채우는 공격

Flooding 이라고 칭하는 경우가 많음

- 통상적으로 위의 공격법을 TCP SYN Flooding 이라고 칭하는 경우가 많음
- FTP프로토콜을 사용한 서버에 다량의 SYN 패킷을 보내 마비시키는것을 FTP Flooding이라고 볼 수 있음.

[해설작성자 : 강썸]

84. 다음 내용이 설명하는 스토리지 시스템은?

- 하드디스크와 같은 데미터 저장장치를 호스트 버스 어댑터에 직접 연결하는 방식
- 저장장치와 호스트 기기 사이에 네트워크 디바이스 없이 직접 연결하는 방식으로 구성

- ① DAS                              ② NAS  
 ③ BSA                              ④ NFC

<문제 해설>

직접 연결 저장장치(Direct-attached storage,DAS)

Network Attached Storage. 네트워크 결합 스토리지  
 근거리 무선 통신(NFC)

[해설작성자 : 58]

85. 취약점 관리를 위해 일반적으로 수행하는 작업이 아닌 것은?

- ① 무결성 검사  
 ② 응용 프로그램의 보안 설정 및 패치(Patch) 적용  
 ③ 중단 프로세스 및 닫힌 포트 위주로 확인  
 ④ 불필요한 서비스 및 악성 프로그램의 확인과 제거

<문제 해설>

중단된 프로세스와 닫힌 포트가 아니라, 활성화된 프로세스와 열린 포트를 중심으로 확인해야 취약점이 관리될 수 있음.

[해설작성자 : 동차할격가즈아]

86. 소프트웨어 생명주기 모델 중 V 모델과 관련한 설명으로 틀린 것은?

- ① 요구 분석 및 설계단계를 거치지 않으며 항상 통합 테스트를 중심으로 V 형태를 이룬다.  
 ② Perry에 의해 제안되었으며 세부적인 테스트 과정으로 구성되어 신뢰도 높은 시스템을 개발하는데 효과적이다.  
 ③ 개발 작업과 검증 작업 사이의 관계를 명확히 들어내

높은 폭포수 모델의 변형이라고 볼 수 있다.

- ④ 폭포수 모델이 산출물 중심이라면 V 모델은 작업과 결과의 검증에 초점을 둔다.

<문제 해설>

V 모델(V-model)은 소프트웨어 개발 프로세스로 폭포수 모델의 확장된 형태 중 하나로 볼 수 있다..아래 방향으로 선형적으로 내려가면서 진행되는 폭포수 모델과 달리, 이 프로세스는 오른쪽 그림과 같이 코딩 단계에서 위쪽으로 꺾어서 알파벳 V자 모양으로 진행된다

검증(Verification)단계

1. 요구사항 분석
2. 시스템 설계
3. 아키텍처 설계
4. 모듈 설계

[해설작성자 : 58]

87. 블루투스(Bluetooth) 공격과 해당 공격에 대한 설명이 올바르게 연결된 것은?

- ① 블루버그(BlueBug) - 블루투스의 취약점을 활용하여 장비의 파일에 접근하는 공격으로 OPP를 사용하여 정보를 열람
- ② 블루스나프(BlueSnarf) - 블루투스를 이용해 스펀처럼 명함을 익명으로 퍼뜨리는 것
- ③ 블루프린팅(BluePrinting) - 블루투스 공격 장치의 검색 활동을 의미
- ④ 블루재킹(BlueJacking) - 블루투스 장비사이의 취약한 연결 관리를 악용한 공격

<문제 해설>

블루버그: 블루투스 장비사이의 취약한 연결 관리를 악용한 공격

블루스나프(블루스나핑): 블루투스의 취약점을 활용하여 장비의 파일에 접근하는 공격으로 OPP를 사용하여 정보를 열람  
 블루재킹: 블루투스를 이용해 스펀처럼 명함을 익명으로 퍼뜨리는 것

\*OPP(Obex Push Protocol)- 블루투스 장치끼리 인증 없이 정보를 간편하게 교환하기 위하여 개발됨

[해설작성자 : 호랑해]

88. DoS(Denial of Service) 공격과 관련한 내용으로 틀린 것은?

- ① Ping of Death 공격은 정상 크기보다 큰 ICMP 패킷을 작은 조각(Fragment)으로 쪼개어 공격 대상이 조각화 된 패킷을 처리하게 만드는 공격 방법이다.
- ② Smurf 공격은 멀티캐스트(Multicast)를 활용하여 공격 대상이 네트워크의 임의의 시스템에 패킷을 보내게 만드는 공격이다.
- ③ SYN Flooding은 존재하지 않는 클라이언트가 서버별로 한정된 접속 가능 공간에 접속한 것처럼 속여 다른 사용자가 서비스를 이용하지 못하게 하는 것이다.
- ④ Land 공격은 패킷 전송 시 출발지 IP주소와 목적지 IP주소 값을 똑같이 만들어서 공격 대상에게 보내는 공격 방법이다.

<문제 해설>

멀티캐스트 x

브로드캐스트 o

[해설작성자 : 토크포테이토]

89. 다음 설명에 해당하는 시스템은?

- 1990년대 David Clock이 처음 제안하였다.
- 비정상적인 접근의 탐지를 위해 의도적으로 설치해 둔 시스템이다.
- 침입자를 속여 실제 공격당하는 것처럼 보여줌으로써 크래커를 추적 및 공격기법의 정보를 수집하는 역할을 한다.
- 쉽게 공격자에게 노출되어야 하며 쉽게 공격이 가능한 것처럼 취약해 보여야 한다.

- ① Apache
- ② Hadoop
- ③ Honeypot
- ④ MapReduce

<문제 해설>

Hadoop(하둡): 오픈 소스를 기반으로한 분산 컴퓨팅 플랫폼, 대형 스토리지, 빅데이터 관련

MapReduce : 대용량 데이터를 분산처리하기 위한 목적으로 개발된 프로그래밍 모델

구글에 의해 고안

임의의 순서로 분산 처리후 다시 합침

[해설작성자 : 부실합니다 보충해 주세요]

Apache : 월드 와이드 웹 컨소시엄(W3C)에서 사용하고 아파치 소프트웨어 재단에서 관리 및 운영하는 서버용 오픈소스 소프트웨어

Hadoop : 오픈 소스를 기반으로 한 분산 컴퓨팅 플랫폼

MapReduce : 대용량 데이터를 분산 처리하기 위한 목적으로 Google에 의해 고안된 프로그래밍 모델

[해설작성자 : 일요일 시험이당]

90. 다음이 설명하는 IT 기술은?

- 컨테이너 응용프로그램의 배포를 자동화하는 오픈소스 엔진이다.
- 소프트웨어 컨테이너 안에 응용 프로그램들을 배치시키는 일을 자동화해 주는 오픈 소스 프로젝트이자 소프트웨어로 볼 수 있다.

- ① StackGuard
- ② Docker
- ③ Cipher Container
- ④ Scytale

<문제 해설>

- StackGuard

- Stack 상에 일정한 주소번지에 프로그램이 선언한 canary를 심어 두어, 스택의 변조 된 경우에, canary를 체크하여 프로그램이 비정상적으로 종료 시키는 기법

- Docker

- 컨테이너 응용프로그램의 배포를 자동화 하는 오픈소스 엔진

- SW 컨테이너 안의 응용 프로그램들을 배치시키는 일을 자동화해 주는 오픈소스 프로젝트 이자 소프트웨어

- Cipher Container

- 자바에서 암호화 복호화 기능을 제공하는 컨테이너

- Scytale

- 암호화 기법으로 단순히 문자열의 위치를 바꾸는 방법

[해설작성자 : 강썸]

91. 간트 차트(Gantt Chart)에 대한 설명으로 틀린 것은?

- ① 프로젝트를 이루는 소작업 별로 언제 시작되고 언제 끝나야 하는지를 한 눈에 볼 수 있도록 도와준다.

- 기출문제 해설의 최강 자격증 기출문제 전자문제집 CBT : [www.comcbt.com](http://www.comcbt.com) 통해서 실시간으로 변경됩니다.



- 수험 처리)
- 논리적인 802.11 MAC 부계층과 물리적인 특성을 연결하는 역할
  - 802.11 MAC 부계층이 물리적 특성에 관계없이 동작하도록 함
  - Traffic Distributor
    - 네트워크 통신 간에 트래픽을 분배해주는 솔루션
  - DPI
    - 네트워크에서 전송되는 패킷의 헤더와 페이로드 내 정보를 분석하는 콘텐츠 내용 분석 기술
    - 네트워크 보안, 관리, 콘텐츠 관리 등이 목적

\*단어의 정의를 몰라도 하나씩 뜯어보면 Deep(내부) Packet Inspection(분석)으로 유추 가능  
 [해설작성자 : 강썸]

#### 100. 소프트웨어 개발 방법론의 테일러링(Tailoring)과 관련한 설명으로 틀린 것은?

- ① 프로젝트 수행 시 예상되는 변화를 배제하고 신속히 진행하여야 한다.
- ② 프로젝트에 최적화된 개발 방법론을 적용하기 위해 절차, 산출물 등을 적절히 변경하는 활동이다.
- ③ 관리 측면에서의 목적 중 하나는 최단기간에 안정적인 프로젝트 진행을 위한 사전 위험을 식별하고 제거하는 것이다.
- ④ 기술적 측면에서의 목적 중 하나는 프로젝트에 최적화된 기술 요소를 도입하여 프로젝트 특성에 맞는 최적의 기법과 도구를 사용하는 것이다.

<문제 해설>

테일러링 (Tailoring) : 프로젝트 상황 특성에 맞게 정의된 소프트웨어 개발 방법론 절차, 사용기법 등을 수정 및 보완하는 작업

[해설작성자 : 땅혁]

본 해설집의 저작권은 [www.comcbt.com](http://www.comcbt.com)에 있으며 카페, 블로그등 개인적 활용 이외에 문서의 수정 및 금전적 이익을 취하는 일체의 행위를 금지 합니다.

전자문제집 CBT PC 버전 : [www.comcbt.com](http://www.comcbt.com)

전자문제집 CBT 모바일 버전 : [m.comcbt.com](http://m.comcbt.com)

기출문제 및 해설집 다운로드 : [www.comcbt.com/xe](http://www.comcbt.com/xe)

전자문제집 CBT란? 인터넷으로 종이 없이 문제를 풀고 자동 채점하는 프로그램으로 워드, 컴활, 기능사 등의 상설검정에서 사용하는 실제 프로그램 방식입니다.

해설을 제공하며 PC 버전 및 모바일 버전 완벽 연동 교사용/학생용 관리기능도 제공합니다.

오답 및 오탈자가 수정된 최신 자료와 해설은 전자문제집 CBT 에서 확인하세요.

1	2	3	4	5	6	7	8	9	10
④	②	③	④	④	③	④	③	②	③
11	12	13	14	15	16	17	18	19	20
③	④	②	③	①	①	②	②	②	②
21	22	23	24	25	26	27	28	29	30
①	②	③	②	④	④	②	④	①	①
31	32	33	34	35	36	37	38	39	40
④	④	①	③	①	④	①	③	③	②
41	42	43	44	45	46	47	48	49	50
①	②	①	③	③	③	①	④	②	②
51	52	53	54	55	56	57	58	59	60
①	①	④	①	①	③	④	④	③	③
61	62	63	64	65	66	67	68	69	70
①	③	①	①	①	①	③	②	④	④
71	72	73	74	75	76	77	78	79	80
①	④	②	②	④	③	①	②	②	③
81	82	83	84	85	86	87	88	89	90
②	①	③	①	③	①	③	②	③	②
91	92	93	94	95	96	97	98	99	100
④	②	④	①	③	①	④	④	④	①