

2019-계리직-컴퓨터일반-A형-해설-곽후근

1. 동기식 전송(Synchronous Transmission)에 대한 설명으로 옳지 않은 것은?

- ① 정해진 숫자만큼의 문자열을 묶어 일시에 전송한다.
- ② 작은 비트블록 앞뒤에 Start Bit와 Stop Bit를 삽입하여 비트블록을 동기화한다.
- ③ 2,400bps 이상 속도의 전송과 원거리 전송에 이용된다.
- ④ 블록과 블록 사이에 유휴시간(Idle Time)이 없어 전송효율이 높다.

정답 체크 :

(2) 해당 설명은 비동기식 전송에 해당한다.

오답 체크 :

- (1) 동기식 전송에서 비트 스트림은 긴 프레임(frame)으로 합쳐지며, 이 프레임에는 여러 바이트가 들어있다.
- (3) 링크를 통해 이동하는 별도의 비트 수가 작기 때문에 비동기식 전송에 비해 빠르다.
- (4) 블록과 블록 사이에 별도의 유휴시간이 존재하지 않는다.

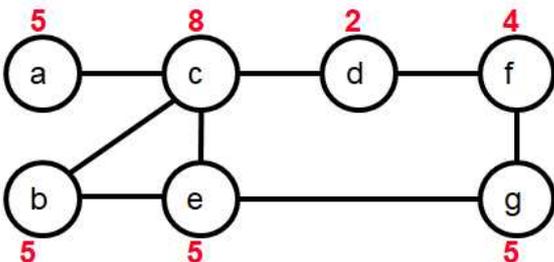
2. 어떤 프로젝트를 완성하기 위해 작업 분할(Work Breakdown)을 통해 파악된, 다음 소작업(activity) 목록을 AOE(Activity On Edge) 네트워크로 표현하였을 때, 이 프로젝트가 끝날 수 있는 가장 빠른 소요시간은?

소작업 이름	소요시간	선행 소작업
a	5	없음
b	5	없음
c	8	a, b
d	2	c
e	3	b, c
f	4	d
g	5	e, f

- ① 13
- ② 21
- ③ 24
- ④ 32

정답 체크 :

(3) $24 = 5+8+2+4+5$: g가 끝나기 위해서는 f, d, c, a가 수행되어야 한다(임계 경로). 나머지 경로에서는 여유 시간이 존재한다. (아래 그림 참고)



3. 다음에 제시된 입력 데이터를 엑셀 서식의 표시 형식 코드에 따라 출력한 결과로 옳은 것은?

입력 데이터 : 1234.5
 표시 형식 코드 : #,##0

- ① 1,234
- ② 1,235
- ③ 1,234.5
- ④ 1,234.50

정답 체크 :

(2) #는 유효 숫자만 표시하고(유효 숫자가 아닌 0은 표시하지 않음), 0은 유효 숫자가 아니어도 이 자리에 무조건 0을 표시한다.
 #,##0은 4자리 숫자이고 주어진 숫자는 4자리 숫자에 소수점을 포함(1234.5)하므로 소수점은 4자리 숫자를 맞추기 위해 반올림이 발생한다(1235).
 1235는 모두 유효 숫자이므로 1,235로 표시된다.

4. 객체지향 소프트웨어 개발 및 UML Diagram에 대한 설명이다. ㉠~㉣에 들어갈 내용을 바르게 짝지은 것은?

- (㉠)은/는 외부에서 인식할 수 있는 특성이 담긴 소프트웨어의 골격이 되는 기본 구조로, 시스템 전체에 대한 큰 밑그림이다. 소프트웨어 품질 요구 사항은 (㉠)을/를 결정하는 데 주요한 요소로 작용한다.
- (㉡)은/는 두 개 이상의 클래스에서 동일한 메시지에 대해 객체가 다르게 반응하는 것이다.
- (㉢)은/는 객체 간의 메시지 통신을 분석하기 위한 것으로 시스템의 동작을 정형화하고 객체들의 메시지 교환을 시각화한다.

㉠	㉡	㉢
① 소프트웨어 아키텍처	다형성	시퀀스 모델
② 유스케이스	다형성	시퀀스 모델
③ 클래스 다이어그램	캡슐화	상태 모델
④ 디자인 패턴	캡슐화	상태 모델

정답 체크 :

(1) 소프트웨어 아키텍처 : 소프트웨어의 구성요소들 사이에서 유기적 관계를 표현하고 소프트웨어의 설계와 업그레이드를 통제하는 지침과 원칙이다.

다형성 : 같은 이름의 메소드가 클래스 혹은 객체에 따라 다르게 구현되는 것이다. 다형성 사례는 메소드 오버로딩(한 클래스 내에서 같은 이름이지만 다르게 작동하는 여러 메소드)과 메소드 오버라이딩(슈퍼 클래스의 메소드를 동일한 이름으로 서브 클래스마다 다르게 구현)이 있다.

시퀀스 모델 : Instance(객체) 들이 어떻게 상호작용을 하는지를 묘사하는 다이어그램(모델)이다(동적인 특성을 가짐).

오답 체크 :

(2) 유스케이스 : 특정 시스템 혹은 개체 내에서 기능을 표현한다.

(3) 클래스 다이어그램 : Class 관련 요소들의 여러 가지 정적인 관계를 시각적으로 표현한 다이어그램이다(정적인 특성을 가짐).

캡슐화 : 객체를 캡슐로 싸서 내부를 볼 수 없게 하는 것이다. 객체의 가장 본질적인 특징으로 외부의 접근으로부터 객체를 보호한다.

상태 모델 : 특정 개체의 동적인 행위를 상태와 그것들 간의 transition을 통해 묘사하는 다이어그램(모델)이다(동적인 특성을 가짐).

(4) 디자인 패턴 : 여러 가지 문제에 대한 설계 사례를 분석하여 서로 비슷한 문제를 해결하기 위한 설계들을 분류하고, 각 문제 유형 별로 가장 적합한 설계를 일반화해 패턴으로 정립한 것을 의미한다.

5. 공개키 암호방식에 대한 설명으로 옳은 것은?

- ① 송신자는 전송메시지에 대한 MAC(Message Authentication Code)을 생성하고 수신자는 그 MAC을 점검함으로써 메시지가 전송과정에서 변조되었는지 여부를 확인한다.
- ② 송신자는 수신자의 개인키를 이용하여 암호화한 메시지를 송신하고 수신자는 수신한 메시지를 자신의 공개키를 이용하여 복호화한다.
- ③ 송수신자 규모가 동일할 경우, 공개키 암호방식이 대칭키 암호방식보다 더 많은 키들을 필요로 하기 때문에 인증기관이 키 관리를 담당한다.
- ④ 키 운영의 신뢰성을 공식적으로 제공하기 위하여 인증기관은 고객별로 개인키와 키 소유자 정보를 만들고 이를 해당 고객에게 인증서로 제공한다.

정답 체크 :

이의 신청으로 인해 모두 정답 처리되었다.

오답 체크 :

- (1) 원래 정답이었으나 다음과 같이 제가 직접 이의 신청을 하였습니다. "출제자의 의도는 공개키 암호방식을 이용해서 MAC을 만들 수 있다는 것으로 보여집니다. 실제로 대학서적 등에서 보면 공개키를 이용해서 MAC을 만들 수 있습니다(자주 사용하지는 않지만 가능성은 존재합니다). 다만, 2017년 서울시 정보보호론(A형) 3번 문제와 충돌이 발생합니다. 해당 문제에서 MAC은 비대칭키(공개키)를 이용할 수 없다고 하였습니다. 그러므로 해당 문제는 정답이 없다고 사료됩니다."
- (2) 수신자의 공개키를 이용하여 메시지를 암호화하고 수신자의 개인키를 이용하여 메시지를 복호화한다.
- (3) 공개키 암호방식이 대칭키 암호방식보다 더 적은 키들을 필요로 한다. 다만, 공개키가 맞다는 것을 증명하기 위해 인증기관이 공인 인증서를 발급한다.
- (4) 인증기관은 고객별로 공개키와 키 소유자 정보를 만들고 이를 해당 고객에게 공인인증서로 제공한다.

6. 온라인에서 멀티미디어 콘텐츠의 불법 유통을 방지하기 위해 삽입된 워터마킹 기술의 특성으로 옳지 않은 것은?

- ① 부인 방지성
- ② 비가시성
- ③ 강인성
- ④ 권리정보 추출성

정답 체크 :

(1) 워터마킹은 구매자 정보가 들어가지 않기 때문에 부인 방지성을 가지지 않는다.

오답 체크 :

- (2) 워터마킹은 스테가노그래피 기술을 사용하기 때문에 비가시성을 가진다.
- (3) 워터마킹은 신호의 중요한 부분에 삽입하여 압축, 노이즈, 변형과 공격에도 추출이 가능해야 하기 때문에 강인성을 가진다.
- (4) 워터마킹은 판매자와 저작권 정보가 들어가기 때문에 권리정보 추출성을 가진다.

7. 프로세스 관리 과정에서 발생할 수 있는 교착상태(Deadlock)를 예방하기 위한 조치로 옳은 것은?

- ① 상호배제(Mutual Exclusion) 조건을 제거하고자 할 경우, 프로세스 A가 점유하고 있던 자원에 대하여 프로세스 B로부터 할당 요청이 있을 때 프로세스 B에게도 해당자원을 할당하여 준다. 운영체제는 프로세스 A와 프로세스 B가 종료되는 시점에서 일관성을 점검하여 프로세스 A와 프로세스 B 중 하나를 철회시킨다.
- ② 점유대기(Hold and Wait) 조건을 제거하고자 할 경우, 자원을 점유한 프로세스가 다른 자원을 요청하였지만 할

당받지 못하면 일단 자신이 점유한 자원을 반납한다. 이후 그 프로세스는 반납하였던 자원과 요청하였던 자원을 함께 요청한다.

③ 비선점(No Preemption) 조건을 제거하고자 할 경우, 프로세스는 시작시점에서 자신이 사용할 모든 자원들에 대하여 일괄할당을 요청한다. 일괄할당이 이루어지지 않을 경우, 일괄할당이 이루어지기까지 지연됨에 따른 성능저하가 발생할 수 있다.

④ 환형대기(Circular Wait) 조건을 제거하고자 할 경우, 자원들의 할당 순서를 정한다. 자원 R_1 가 자원 R_2 보다 먼저 할당되는 것으로 정하였을 경우, 프로세스 A가 R_1 를 할당받은 후 R_2 를 요청한 상태에서 프로세스 B가 R_2 를 할당받은 후 R_1 를 요청하면 교착상태가 발생하므로 운영체제는 프로세스 B의 자원요청을 거부한다.

정답 체크 :

(4) 모든 프로세스에 자원을 순서대로 할당한다

오답 체크 :

(1) 일반적으로 상호배제 조건을 만족하지 않으면, 교착 상태 예방이 불가능하다. 즉, 상호배제가 아닌 공유가 되면 더 큰 문제가 발생한다.

(2) 해당 설명은 비선점 조건을 제거하고자 할 경우를 나타낸다.

(3) 해당 설명은 점유대기 조건을 제거하고자 할 경우를 나타낸다.

8. 순차 파일과 인덱스 순차 파일에 대한 설명으로 옳은 것의 총 개수는?

- ㄱ. 순차 파일에서의 데이터 레코드 증가는 적용된 순차 기준으로 마지막 위치에서 이루어진다.
- ㄴ. 순차 파일에서는 접근 조건으로 제시된 순차 대상 필드 값 범위에 해당하는 대량의 데이터 레코드들을 접근할 때 효과적이다.
- ㄷ. 순차 파일에서의 데이터 레코드 증가는 오버플로우 블록을 생성시키지 않는다.
- ㄹ. 인덱스 순차 파일의 인덱스에는 인덱스 대상 필드 값과 그 값을 가지는 데이터 레코드를 접근할 수 있게 하는 위치 값이 기록된다.
- ㅁ. 인덱스 순차 파일에서는 인덱스 갱신없이 데이터 레코드를 추가하거나 삭제하는 것이 가능하다.
- ㅂ. 인덱스 순차 파일에서는 접근 조건에 해당하는 인덱스 대상 필드 값을 가지는 소량의 데이터 레코드를 순차 파일보다 효과적으로 접근할 수 있다.
- ㅅ. 인덱스를 다중레벨로 구성할 경우, 최하위 레벨은 순차 파일 형식으로 구성된다.

① 2개

② 3개

③ 4개

④ 5개

정답 체크 :

(3) ㄱ. 입력되는 데이터들을 논리적인 순서에 따라 물리적 연속 공간에 순차적으로 기록하는 방식이다.

ㄷ. 순차 파일에서는 별도의 오버플로우 블록이 존재하지 않는다.

ㄹ. 인덱스에는 레코드들의 위치를 찾아가는 색인이 기록된다.

ㅁ. 추가되는 것은 상편이 없고, 삭제가 인덱스를 건드리지 않으면 인덱스를 갱신하지 않아도 된다.

오답 체크 :

(1), (2), (4) ㄴ. 데이터 검색 시 처음부터 순차적으로 검색하기 때문에 대량의 데이터 레코드들을 접근할 때 효과적이지 않다.

비. 인덱스 순차 파일에서는 대량의 데이터 레코드를 순차 파일보다 효과적으로 접근할 수 있다.

스. 최하위는 data file이 아니고 인덱스를 나타내므로 이는 순차 파일 형식이 아니다. (인덱스를 다중 레벨로 구성한다고 했으므로 최하위는 인덱스가 됨)

9. Java 프로그램의 실행 결과로 옳은 것은?

<pre>public class B extends A { int a = 20; public B() { System.out.print("다"); } public B(int x) { System.out.print("라"); } }</pre>	<pre>public class A { int a = 10; public A() { System.out.print("가"); } public A(int x) { System.out.print("나"); } public static void main(String[] a){ B b1 = new B(); A b2 = new B(1); System.out.print(b1.a + b2.a); } }</pre>
--	---

- ① 다라30
- ② 다라40
- ③ 가다가라30
- ④ 가다가라40

정답 체크 :

(3) B b1 = new B(); // B()의 생성자를 호출하기 전에 A()의 생성자(디폴트)를 호출한다. 즉, "가다"를 출력하고, b1.a에 20이 할당된다(B의 객체).

A b2 = new B(1); // B(int x)의 생성자를 호출하기 전에 A()의 생성자(디폴트)를 호출한다. 즉, "가라"를 출력하고, b2.a에 10이 할당된다(A의 객체).

System.out.print(b1.a+b2.a); // "30"을 출력한다.

10. C 언어로 작성된 프로그램의 실행 결과로 옳은 것은?

```

#include <stdio.h>

double h(double *f, int d, double x){
    int i;
    double res = 0.0;
    for(i=d-1; i >= 0; i--){
        res = res * x + f[i];
    }
    return res;
}

int main() {
    double f[] = {1, 2, 3, 4};
    printf("%3.1f\n", h(f, 4, 2));
    return 0;
}

```

- ① 11.0
- ② 26.0
- ③ 49.0
- ④ 112.0

정답 체크 :

(3) $i = 3$; // $res = 4.0$;
 $i = 2$; // $res = 4 \times 2 + 3 = 11.0$;
 $i = 1$; // $res = 11 \times 2 + 2 = 24.0$;
 $i = 0$; // $res = 24 \times 2 + 1 = 49.0$;
 printf("%3.1f\n", 49.0); // 3은 출력폭(49.0은 .을 포함해서 출력폭이 40이나 49는 유효 숫자이므로 자동으로 출력폭이 확장됨), 1은 소수점 이하의 숫자 개수(.0)이다.

11. (가), (나)에서 설명하는 악성 프로그램의 용어를 바르게 짝지은 것은?

(가) 사용자 컴퓨터의 데이터를 암호화시켜 파일을 사용할 수 없도록 한 후 암호화를 풀어주는 대가로 금전을 요구하는 악성 프로그램
 (나) '○○○초대장' 등의 내용을 담은 문자 메시지 내에 링크된 인터넷 주소를 클릭하면 악성 코드가 설치되어 사용자의 정보를 빼가거나 소액결제를 진행하는 악성 프로그램

- | | |
|---------|---------------|
| (가) | (나) |
| ① 스파이웨어 | 트로이목마 |
| ② 랜섬웨어 | 파밍(Pharming) |
| ③ 스파이웨어 | 피싱(Phishing) |
| ④ 랜섬웨어 | 스미싱(Smishing) |

정답 체크 :

(4) 랜섬웨어 : 컴퓨터 시스템을 감염시켜 접근을 제한하고 일종의 몸값을 요구하는 악성 소프트웨어의 한 종류이다. 컴퓨터로의 접근이 제한되기 때문에 제한을 없애려면 해당 악성 프로그램을 개발한 자에게 지불을 강요받게 된다.

스미싱 : SMS(문자 메시지)와 Phishing의 약자이다. Phishing은 Private Data(개인 정보)와 Fishing(낚시)의 약자이다. 공격자가 문자 메시지에 URL을 보내고, 사용자가 이를 클릭하면 해킹 툴이 스마트폰에 설치되어 개인 정보가 탈취된다.

오답 체크 :

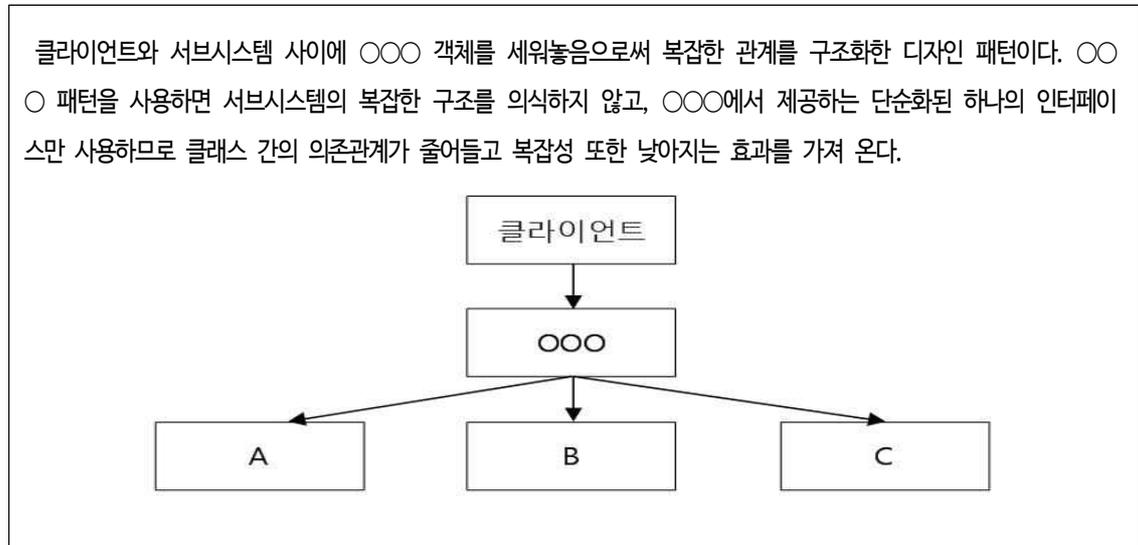
(1) 스파이웨어 : 사용자의 동의 없이 또는 사용자를 속이고 설치되어 광고나 마케팅용 정보를 수집하거나 중요한 개인 정보를 빼내는 악성 프로그램이다.

트로이목마 : 악성 루틴이 숨어 있는 프로그램으로, 겉보기에는 정상적인 프로그램으로 보이지만 실행하면 악성 코드를 실행한다.

(2) 파밍 : Phishing(개인 정보)과 farming(대규모 피해)의 합성어이다. DNS Spoofing과 같이 인터넷 주소창에 방문하고자 하는 사이트의 URL을 입력하였을 때 가짜 사이트(fake site)로 이동시키는 공격 기법이다.

(3) 피싱 : Private data(개인 정보)와 fishing(낚는다)의 합성어이다. 불특정 다수에게 메일을 발송해 위장된 홈페이지로 접속하도록 한 뒤 인터넷 이용자들의 금융정보와 같은 개인정보를 빼내는 사기 기법을 말한다.

12. 다음에서 설명하는 디자인 패턴으로 옳은 것은?



- ① MVC pattern
- ② facade pattern
- ③ mediator pattern
- ④ bridge pattern

정답 체크 :

(2) Façade는 '건물의 앞쪽 정면(전면)'을 의미한다. 몇 개의 클라이언트 클래스와 서브시스템의 클라이언트 사이에 facade라는 객체를 세워놓음으로써 복잡한 관계를 정리(구조화)한 것이다. 모든 관계가 전면에 세워진 facade 객체를 통해서만 이루어질 수 있게 단순한 인터페이스를 제공(단순한 창구 역할)하는 것이다.

오답 체크 :

(1) MVC에서 모델은 애플리케이션의 정보(데이터)를 나타내며, 뷰는 텍스트, 체크박스 항목 등과 같은 사용자 인터페이스 요소를 나타내고, 컨트롤러는 데이터와 비즈니스 로직 사이의 상호동작을 관리한다.

(3) Mediator은 '중재자', '조정자', '중개인'을 의미한다. 부동산 중개사, 비행기의 이착륙을 통제하는 관제탑, 중고물건을 사고파는 사이트처럼 중간에서 연결하고 통제하는 역할이다.

(4) bridge는 '무엇인가를 연결 한다'를 의미한다. 두 장소를 연결하는 역할을 한다. 기능의 클래스 계층과 구현의 클래스 계층을 연결

하고, 구현부에서 추상 계층을 분리하여 각자 독립적으로 변형할 수 있게 해준다. 구현과 인터페이스(추상화된 부분)를 분리할 수 있고, 추상화된 부분과 실제 구현 부분을 독립적으로 확장할 수 있다. 해당 패턴은 기능과 구현(추상화)의 분리되었음에 유의한다.

13. SQL의 명령을 DDL, DML, DCL로 구분할 경우, 이를 바르게 짝지은 것은?

	<u>DDL</u>	<u>DML</u>	<u>DCL</u>
①	RENAME	SELECT	COMMIT
②	UPDATE	SELECT	GRANT
③	RENAME	ALTER	COMMIT
④	UPDATE	ALTER	GRANT

정답 체크 :

(1) DDL은 테이블을 생성하고 변경 및 제거하는 기능을 제공한다. 데이터 구조를 정의하는 질의문이고, 데이터베이스를 처음 생성하고 개발할 때 주로 사용하고 운영 중에는 거의 사용하지 않는다. DDL의 종류는 다음과 같다.

- CREATE: 데이터베이스 객체(table or view)를 생성한다.
- ALTER: 데이터베이스 객체를 변경한다.
- DROP: 데이터베이스 객체를 삭제한다.
- RENAME: 데이터베이스 객체의 이름을 변경한다.
- TRUNCATE: 데이터베이스를 객체를 삭제하고 새로 만든다.

DML은 테이블에 새 데이터를 삽입하거나, 테이블에 저장된 데이터를 수정/삭제/검색하는 기능을 제공한다. 데이터베이스의 운영 및 사용과 관련해 가장 많이 사용하는 질의문이다. DML의 종류는 다음과 같다.

- SELECT: 사용자가 테이블이나 뷰의 내용을 읽고 선택한다.
- INSERT: 데이터베이스 객체에 데이터를 입력한다.
- UPDATE: 기존 데이터베이스 객체에 있는 데이터를 수정한다.
- DELETE: 데이터베이스 객체에 있는 데이터를 삭제한다.

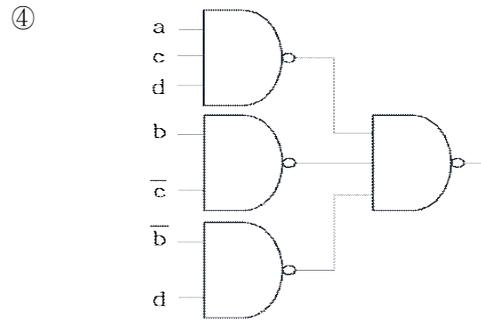
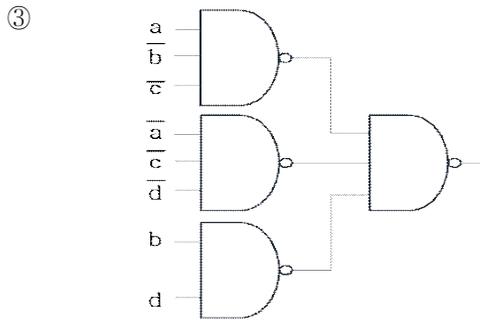
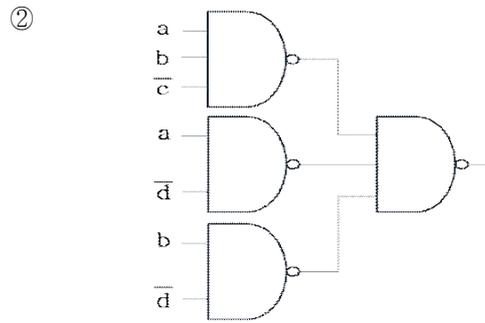
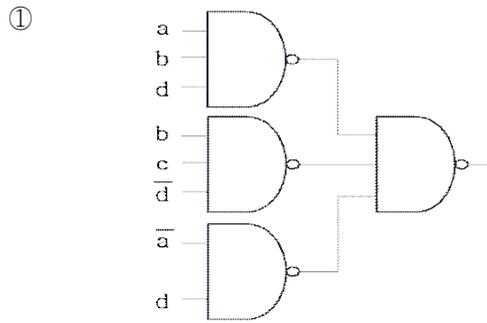
DCL은 보안을 위해 데이터에 대한 접근 및 사용 권한을 사용자별로 부여하거나 취소하는 기능을 제공한다. DCL의 종류는 다음과 같다.

- GRANT: 데이터베이스 객체에 권한을 부여한다.
- DENY: 사용자에게 해당 권한을 금지한다.
- REVOKE: 이미 부여된 데이터베이스 객체의 권한을 취소한다.
- COMMIT: SQL 명령을 데이터베이스에 반영한다(실행한다).
- ROLLBACK: SQL 명령을 취소한다.

14. ㉠과 ㉡에 들어갈 용어로 바르게 짝지은 것은?

(㉠)은/는 구글에서 개발해서 공개한 인공지능 응용프로그램 개발용 오픈소스 프레임워크이다. 이 프레임워크를 사용할 때 인공지능 소프트웨어가 이미지 및 음성을 인식하기 위해서는 신경망의 (㉡) 모델을 주로 사용한다.

- | | |
|---------|---------|
| ㉠ | ㉡ |
| ① 텐서플로우 | 논리곱 신경망 |
| ② 알파고 | 퍼셉트론 |



정답 체크 :

(3) $ab'c' + a'c'd' + bd = ((ab'c')'(a'c'd')'(bd))'$ (드모르간 법칙)

ab \ cd	00	01	11	10
00	1	0	0	0
01	1	1	1	0
11	0	1	1	0
10	1	1	0	0

a'c'd' (points to 00, 01, 10 row)
 bd (points to 11, 10 column)
 ab'c' (points to 00, 01, 10 row)

16. 다음은 숫자를 처리하는 C 프로그램이다. 프로그램에서 ㉠과 ㉡에 들어갈 내용과 3 2 1 4를 입력하였을 때의 출력결과를 바르게 짝지은 것은? (단, 다음 프로그램에 문법적 오류는 없다고 가정한다.)

```

#include <stdio.h>
#include <stdlib.h>

void a (int n, int *num) {
    for (int i = 0; i < n; i++)
        scanf("%d", &(num[i]));
}

void c(int *a, int *b) {
    int t;
  
```

```

        t = *a; *a = *b; *b = t;
    }
    void b(int n, int *lt) {
        int a, b;
        for (a = 0; a < n-1; a++)
            for (b = a + 1; b < n; b++)
                if (lt[a] > lt[b]) c ( ㉠ , ㉡ ) ;
    }
    int main() {
        int n;
        int *num;
        printf("How many numbers?");
        scanf("%d", &n);
        num = (int *)malloc(sizeof(int) * n);
        a(n, num);
        b(n, num);
        for (int i = 0; i < n; i++)
            printf("%d ", num[i]);
    }

```

- | ㉠ | ㉡ | 출력 결과 |
|---------|-------|---------|
| ① lt+a | lt+b | 1 2 3 4 |
| ② lt+a | lt+b | 1 2 4 |
| ③ lt[a] | lt[b] | 4 3 2 1 |
| ④ lt[a] | lt[b] | 4 2 1 |

정답 체크 :
 (2) scanf("%d", &n); // 3이 입력되므로 n에 3이 대입된다.
 a(n, num); // 3개의 숫자(2, 1, 4)를 입력받는다.
 b(n, num); // 입력받은 3개의 숫자를 버블 정렬을 통해 오름 차순으로 정렬한다(1, 2, 4). 이때, lt+a와 lt[a]는 동일한 의미이다.
 printf("%d ", num[i]); // 1, 2, 4를 출력한다.

17. 엑셀 시트를 이용해 수식을 실행한 결과, 값이 나머지와 다른 것은?

	A
1	3
2	7
3	5
4	3
5	0
6	1

- ① =GCD(A1,A6)
- ② =MEDIAN(A1:A6)
- ③ =MODE(A1:A6)
- ④ =POWER(A1,A6)

정답 체크 :
 (1) GCD(3, 1)은 최대 공약수를 구하는 함수이다. 결과는 1이다.
 오답 체크 :
 (2) MEDIAN(3:1)은 중간값을 구하는 함수이다. 결과는 입력의 개수가 홀수개이면 중간 위치를 중간 값으로 구하고, 입력의 개수가 짝수개면 중앙에 있는 2개의 값을 평균한다. 결과는 3(=(3+3)/2)이다.
 (3) MODE(3:1)은 최빈값을 구하는 함수이다. 결과는 3이다(3의 빈도가 제일 높음).

(4) POWER(3, 1)은 거듭 제곱을 구하는 함수이다. 결과는 3(=31)이다.

18. RISC(Reduced Instruction Set Computer)에 대한 설명으로 옳은 것의 총 개수는?

- ㄱ. 칩 제작을 위한 R&D 비용이 감소한다.
- ㄴ. 개별 명령어 디코딩 시간이 CISC(Complex Instruction Set Computer)보다 많이 소요된다.
- ㄷ. 동일한 기능을 구현할 경우, CISC보다 적은 수의 레지스터가 필요하다.
- ㄹ. 복잡한 연산을 수행하려면 명령어를 반복수행하여야 하므로 CISC의 경우보다 프로그램이 복잡해진다.
- ㅁ. 각 명령어는 한 클럭에 실행하도록 고정되어 있어 파이프라인 성능을 향상시킬 수 있다.
- ㅂ. 마이크로코드 설계가 어렵다.
- ㅅ. 고정된 명령어이므로 명령어 디코딩 속도가 빠르다.

- ① 2개
- ② 3개
- ③ 4개
- ④ 5개

정답 체크 :

- (3) ㄱ. 제어장치가 간단하기 때문에 칩 제작을 위한 R&D 비용이 감소한다.
- ㄹ. 명령어가 간단하므로 복잡한 연산을 수행하려면 프로그램이 복잡해진다(많은 명령어를 사용해야 한다).
- ㅁ. 각 명령어를 한 클럭에 실행할 수 있다면 해저드 발생 빈도를 줄일 수 있어 파이프라인 성능을 향상시킬 수 있다.
- ㅅ. 고정되고 간단한 명령어이기 때문에 명령어 디코딩 속도가 빠르다.

오답 체크 :

- (1), (2), (4) ㄴ. 개별 명령어 디코딩 시간이 CISC보다 적게 소요된다. 왜냐하면 명령어가 간단하기 때문이다.
- ㄷ. 동일한 기능을 구현할 경우 CISC보다 많은 수의 레지스터가 필요하다. RISC의 경우 대부분의 연산이 레지스터 상에서 수행되기 때문이다.
- ㅂ. 제어장치가 간단하므로 마이크로코드 설계가 쉽다.

19. 참조 무결성에 대한 설명으로 옳지 않은 것은?

- ① 검색 연산의 수행 결과는 어떠한 참조 무결성 제약조건도 위배하지 않는다.
- ② 참조하는 릴레이션에서 튜플이 삭제되는 경우, 참조 무결성 제약조건이 위배될 수 있다.
- ③ 외래 키 값은 참조되는 릴레이션의 어떤 튜플의 기본 키 값과 같거나 널(NULL) 값일 수 있다.
- ④ 참조 무결성 제약조건은 DBMS에 의하여 유지된다.

정답 체크 :

- (2) "참조하는"이 아닌 "참조되는" 릴레이션에서 튜플이 삭제되는 경우, 참조 무결성 제약조건이 위배될 수 있다.

오답 체크 :

- (1) 삽입(insert), 삭제(delete), 수정(update)에서 참조 무결성 제약조건이 위배된다.
- (3) 기본 키 값과 같거나 널 값일 수는 있지만 기본 키에 없는 값을 가질 수는 없다.
- (4) DBMS는 4가지 무결성 제약조건(개체, 참조, 도메인, 사용자 정의)을 유지한다.

20. 프로세스(Process)와 스레드(Thread)에 대한 설명으로 옳지 않은 것은?

- ① 프로세스 내 스레드 간 통신은 커널 개입을 필요로 하지 않기 때문에 프로세스 간 통신보다 더 효율적으로 이루어진다.
- ② 멀티프로세서는 탑재 프로세서마다 스레드를 실행시킬 수 있기 때문에 프로세스의 처리율을 향상시킬 수 있다.
- ③ 한 프로세스 내의 모든 스레드들은 정적 영역(Static Area)을 공유한다.
- ④ 한 프로세스의 어떤 스레드가 스택 영역(Stack Area)에 있는 데이터 내용을 변경하면 해당 프로세스의 다른 스레드가 변경된 내용을 확인할 수 있다.

정답 체크 :

(4) 스레드는 스택을 독립적으로 가지므로 다른 스레드가 변경된 내용을 확인할 수 없다.

오답 체크 :

- (1) 프로세스 간 통신은 통신을 위해 별도의 메모리를 사용해야 하지만 스레드 간 통신은 미리 공유된 메모리를 사용하므로 더 효율적이다.
- (2) 멀티프로세서를 통해 여러 개의 스레드를 동시에 실행할 수 있어 처리율이 향상된다.
- (3) 스레드들은 주기억장치의 코드(텍스트), 데이터, 힙 영역을 공유한다. 이중 코드(텍스트), 데이터는 정적 영역이다.