



## 저작권 안내

이 자료는 시나공 카페 회원을 대상으로 하는 자료로서 개인적인 용도로만 사용할 수 있습니다. 허락 없이 복제하거나 다른 매체에 옮겨 실을 수 없으며, 상업적 용도로 사용할 수 없습니다.

※ 다음 문제를 읽고 알맞은 것을 골라 답안카드의 답란(가, 나, 다, 라)에 표기하시오.

## 제1과목 데이터베이스

1. 개체-관계(E-R) 모델에 대한 설명으로 옳지 않은 것은?

- 가. E-R 다이어그램으로 표현하며 P.Chen이 제안했다.  
나. 일대일(1:1) 관계 유형만을 표현할 수 있다.  
다. 개체 타입과 이들 간의 관계 타입을 이용해 현실 세계를 개념적으로 표현한 방법이다.  
라. E-R 다이어그램은 E-R 모델을 그래프 방식으로 표현한 것이다.

2. 로킹(Locking) 단위에 대한 설명으로 옳은 것은?

- 가. 로킹 단위가 크면 병행성 수준이 낮아진다.
- 나. 로킹 단위가 크면 병행 제어 기법이 복잡해진다.
- 다. 로킹 단위가 작으면 로크(Lock)의 수가 적어진다.
- 라. 로킹은 파일 단위로 이루어지며, 레코드 또는 필드는 로킹 단위가 될 수 없다.

3. 관계 데이터 모델에서 릴레이션(Relation)에 포함되어 있는 튜플(Tuple)의 수를 무엇이라고 하는가?

- |              |                      |
|--------------|----------------------|
| 가. Degree    | 나. Cardinality       |
| 다. Attribute | 라. Cartesian Product |

4. 트랜잭션의 특성으로 옳지 않은 것은?

- 가. 트랜잭션의 연산은 데이터베이스에 모두 반영되든지, 아니면 전혀 반영되지 않아야 한다.
- 나. 트랜잭션이 그 실행을 성공적으로 완료하면 언제나 일관성 있는 데이터베이스 상태로 변환한다.
- 다. 둘 이상의 트랜잭션이 동시에 병행 실행되는 경우 어느 하나의 트랜잭션 실행 중에 다른 트랜잭션의 연산이 끼어들 수 있다.
- 라. 트랜잭션에 의해서 생성된 결과는 계속 유지되어야 한다.

5. 다음 SQL문의 실행 결과를 가장 올바르게 설명한 것은?

DROP TABLE 인사 CASCADE;

- 가. 인사 테이블을 제거한다.
- 나. 인사 테이블을 참조하는 테이블과 인사 테이블을 제거한다.
- 다. 인사 테이블이 참조 중이면 제거하지 않는다.
- 라. 인사 테이블을 제거할 지의 여부를 사용자에게 다시 질의한다.

6. “회사원”이라는 테이블에서 “사원명”을 검색할 때, “연락번호”가 Null 값이 아닌 “사원명”을 모두 찾을 경우의 SQL 질의로 옳은 것은?

- 가. SELECT 사원명 FROM 회사원  
WHERE 연락처호 != NULL;
- 나. SELECT 사원명 FROM 회사원  
WHERE 연락처호 <> NULL;
- 다. SELECT 사원명 FROM 회사원  
WHERE 연락처호 IS NOT NULL;
- 라. SELECT 사원명 FROM 회사원  
WHERE 연락처호 DON'T NULL;

7. 뷰(View)에 대한 설명으로 옳지 않은 것은?

- 가. 뷰는 독자적인 인덱스를 가질 수 없다.  
나. 뷰의 정의를 변경할 수 없다.  
다. 뷰로 구성된 내용에 대한 삽입, 갱신, 삭제 연산에는 제약이 따른다.  
라. 뷰가 정의된 기본 테이블이 삭제되더라도 뷰는 자동적으로 삭제되지 않는다.

8. 데이터베이스 설계 단계 중 물리적 설계에서 옵션 선택시 고려사항으로 거리가 먼 것은?

- 가. 스키마의 평가 및 정제
- 나. 응답 시간
- 다. 저장 공간의 효율화
- 라. 트랜잭션 처리율

9. 데이터베이스 언어 중 DDL의 기능이 아닌 것은?

- 가. 논리적, 물리적 데이터 구조의 정의
- 나. 데이터 회복과 병행 수행 제어
- 다. 논리적 데이터 구조와 물리적 데이터 구조의 사상 정의
- 라. 데이터베이스 정의 및 수정

10. 데이터베이스 보안에 대한 설명으로 옳지 않은 것은?

- 가. 보안을 위한 데이터 단위는 테이블 전체로부터 특정 테이블의 특정한 행과 열 위치에 있는 특정한 데이터 값에 이르기까지 다양하다.
- 나. 각 사용자들은 일반적으로 서로 다른 객체에 대하여 다른 접근권리 또는 권한을 갖게 된다.
- 다. 불법적인 데이터의 접근으로부터 데이터베이스를 보호하는 것이다.
- 라. 보안을 위한 사용자들의 권한 부여는 관리자의 정책 결정보다는 DBMS가 자체 결정하여 제공한다.

11. It is specified between two relations and is used to maintain the consistency among tuples of the two relations. What is it?

- 가. Entity integrity constraint  
나. Referential integrity constraint  
다. Domain integrity constraint  
라. Data integrity constraint

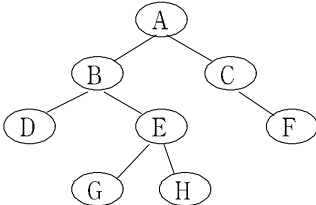
12. Which of the following does not belong to the DDL statement of SQL?

- 가. DELETE                      나. CREATE  
다. DROP                        라. ALTER

13. 시스템 카탈로그에 대한 설명으로 옳은 것은?

- 가. 메타 데이터를 갖고 있는 시스템 데이터베이스이다.  
나. 일반 사용자도 제한 없이 시스템 카탈로그의 내용을 직접 갱신할 수 있다.  
다. 시스템 카탈로그는 사용자의 테이블당 한 개씩 만들어진다.  
라. 시스템 카탈로그는 DBA가 생성한다.

14. 다음 트리에 대한 프리-오더(Pre-Order) 운행 결과는?



- 가. D G H E B F C A  
나. D B G E G A C F  
다. A B D E G H C F  
라. A B C D E F G H

15. 릴레이션에 대한 설명으로 옳지 않은 것은?

- 가. 모든 튜플은 서로 다른 값을 가지고 있다.  
나. 하나의 릴레이션에서 튜플은 순서를 가진다.  
다. 각 속성은 릴레이션 내에서 유일한 이름을 가진다.  
라. 모든 속성 값은 원자 값(Atomic Value)을 가진다.

16. 내장 SQL(Embedded SQL)에 대한 설명으로 옳지 않은 것은?

- 가. 응용 프로그램 내에 SQL 문장을 내포하여 프로그램이 실행될 때 함께 실행되도록 호스트 프로그램 언어에 삽입된 SQL을 의미한다.  
나. 호스트변수와 데이터베이스 필드의 이름이 동일해서는 안된다.  
다. 호스트변수의 데이터 타입은 이에 대응하는 데이터베이스 필드의 SQL 데이터 타입과 일치해야 한다.  
라. 내장 SQL 실행문은 호스트 언어에서 실행문이 나타날 수 있는 곳이면 프로그램의 어느 곳에서나 사용할 수 있다.

17. 데이터베이스의 특성으로 옳지 않은 것은?

- 가. 같은 내용의 데이터를 여러 사람이 동시에 공유할 수 있다.  
나. 데이터베이스는 데이터의 삽입, 삭제, 갱신으로 내용이 계속적으로 변한다.  
다. 수치적이고 비정형적인 질의에 대하여 실시간 처리로 응답할 수 있어야 한다.  
라. 데이터의 참조는 저장되어 있는 데이터 레코드들의 주소나 위치에 의해서 이루어진다.

18. 해싱을 이용한 파일 구조에 해당하는 것은?

- 가. 순차(Sequential) 파일  
나. 직접(Direct) 파일  
다. 색인 순차(Indexed Sequential) 파일  
라. 다중 키(Multi-Key) 파일

19. 제 2정규형에서 제 3정규형이 되기 위한 조건은?

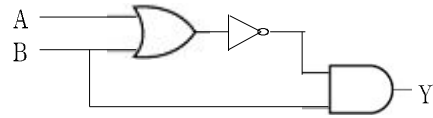
- 가. 이행적 함수 종속 제거  
나. 부분적 함수 종속 제거  
다. 다치 종속 제거  
라. 결정자이면서 후보 키가 아닌 것 제거

20. 순서가 A, B, C, D로 정해진 입력 자료를 스택에 입력하였다가 출력하는 경우, 출력 결과로서 가능하지 않은 것은?

- 가. D, A, B, C                      나. B, D, C, A  
다. C, B, D, A                      라. B, A, D, C

## 제2과목 전자계산기 구조

21. 그림과 같은 회로에서 출력 Y는?



- 가.  $Y = A \cdot B + B$                       나.  $Y = \overline{A \cdot B} + B$   
다.  $Y = \overline{A + B} + B$                       라.  $Y = (A + B) \cdot B$

22. 프로그램에 의해 제어되는 동작이 아닌 것은?

- 가. Input/Output                      나. Branch  
다. Status Sense                      라. RNI(Fetch)

23. 결선 게이트의 특징이 아닌 것은?

- 가. 게이트들의 출력단자를 직접 연결한다.  
나. 회로 비용을 절감할 수 있다.  
다. 많은 논리기능을 부여할 수 없다.  
라. Open Collector TTL로 게이트들의 출력 단자를 묶어서 사용한다.

24. 비트 슬라이스 마이크로프로세서(Bit Sliced Microprocessor)의 구성을 가장 잘 설명한 것은?

- 가. CPU를 하나의 IC로 만든 프로세서  
나. CPU, 기억장치, I/O Port가 한 IC에 구성된 프로세서  
다. Processor Unit, Microprogram Sequencer, Control Memory가 각각 다른 IC로 구성된 프로세서  
라. Processor Unit, Microprogram Sequencer, Control Memory가 한 IC로 구성된 프로세서

25. 중앙처리장치가 FETCH 상태에서 경우에 제어점을 제어하는 것은?

- 가. 플래그(Flag)                      나. 명령어(Instruction)  
다. 인터럽트 호출 신호                      라. 프로그램 카운터

26. 플립플롭 중 입력단자가 하나이며, "1"이 입력될 때마다 출력단자의 상태가 바뀌는 것은?

- 가. RS 플립플롭                      나. T 플립플롭  
다. D 플립플롭                      라. M/S 플립플롭

27. 64K인 주소 공간(Address Space)과 4K인 기억공간(Memory Space)을 가진 컴퓨터인 경우 한 페이지(Page)가 512워드 구성된다면 페이지와 블록 수는 각각 얼마인가?

- 가. 16페이지 12블록                      나. 128페이지 8블록  
다. 256페이지 16블록                      라. 64페이지 4K블록

28. 가상 메모리를 사용한 컴퓨터에서 Page Fault가 발생하면 어떤 현상이 일어나는가?

- 가. 요구된 Page가 주기억장치로 옮겨질 때까지 프로그램 수행이 중단된다.  
나. 요구된 Page가 가상메모리로 옮겨질 때까지 프로그램 수행이 중단된다.  
다. 현재 실행 중인 프로그램을 종료한 후 시스템이 정지된다.  
라. Page Fault라는 에러 메시지를 전송한 후에 시스템이 정지된다.

29. RISC(Reduced Instruction Set Computer)와 CISC(Complex Instruction Set Computer)의 특징이 아닌 것은?

- 가. RISC는 명령어의 길이가 고정적이다.  
 나. RISC는 하드웨어에 의해 직접 명령어가 수행된다.  
 다. CISC의 수행 속도가 더 빠르다.  
 라. 펜티엄을 포함한 인텔사의 x86 시리즈는 CISC 프로세서이다.

30. 인스트럭션을 수행하기 위한 메이저 상태에 대한 설명으로 옳은 것은?

- 가. 명령어를 가져오기 위해 기억장치에 접근하는 것을 Fetch 상태라 한다.  
 나. Execute 상태는 간접주소 지정방식의 경우 수행된다.  
 다. CPU의 현재 상태를 보관하기 위한 기억장치 접근을 Indirect 상태라 한다.  
 라. 명령어 종류를 판별하는 것을 Indirect 상태라 한다.

31. BCD 코드 1001에 대한 해밍 코드를 구하면?

- 가. 0011001                      나. 1000011  
 다. 0100101                      라. 0110010

32. 소프트웨어 인터럽트 사용 시 가장 큰 장점은?

- 가. 우선 순위 변경이 쉽다.  
 나. 속도가 빠르다.  
 다. 비용이 비싸다.  
 라. 페이지 체인 방식이다.

33. Op-Code가 4비트이면 연산자의 종류는 몇 개가 생성될 수 있는가?

- 가.  $2^4-1$                           나.  $2^4$   
 다.  $2^3$                               라.  $2^3-1$

34. 컴퓨터의 메모리 용량이 16K × 32Bit라 하면 MAR(Memory Address Register)와 MBR(Memory Buffer Register)은 각각 몇 비트인가?

- 가. MAR : 12, MBR : 16  
 나. MAR : 32, MBR : 14  
 다. MAR : 12, MBR : 32  
 라. MAR : 14, MBR : 32

35. 스택(Stack)이 사용되는 경우는?

- 가. 인터럽트가 발생할 때  
 나. 분기 명령이 실행될 때  
 다. 무조건 점프 명령이 실행될 때  
 라. 메모리 요구가 받아들여졌을 때

36. 내부 인터럽트의 원인이 아닌 것은?

- 가. 정전  
 나. 불법적인 명령의 실행  
 다. Overflow 또는 0(Zero)으로 나누는 경우  
 라. 보호 영역내의 메모리 주소를 Access하는 경우

37. 고속의 입·출력 장치에 사용되는 데이터 전송 방식은?

- 가. 데이터 채널                  나. I/O 채널  
 다. Selector 채널                라. Multiplexer 채널

38. 데이터 전송 방법 중 스트로브 제어 방법의 설명으로 옳지 않은 것은?

- 가. 전송을 시작한 송신장치가 버스에 놓인 데이터를 수신 장치가 받아 들였는지 여부를 알 수 있다.  
 나. 비동기 방식으로 각 전송 시간을 맞추기 위해 단 하나의 제어 라인을 갖는다.  
 다. 스트로브는 송신장치나 수신장치에 의하여 발생된다.

라. 수신장치는 스트로브 펄스를 발생시켜 송신부로 하여금 데이터를 제공하도록 알린다.

39. 서로 다른 17개의 정보가 있다. 이 중에서 하나를 선택하려면 최소 몇 개의 비트가 필요한가?

- 가. 3                                  나. 4  
 다. 5                                  라. 17

40. CPU가 인스트럭션을 수행하는 순서로 옳은 것은?

- ㉠ 인터럽트 조사              ㉡ 인스트럭션 디코딩  
 ㉢ 인스트럭션 Fetch        ㉣ Operand Fetch  
 ㉤ Execution

- 가. ㉢ → ㉠ → ㉡ → ㉣ → ㉤  
 나. ㉡ → ㉢ → ㉡ → ㉤ → ㉠  
 다. ㉡ → ㉢ → ㉣ → ㉤ → ㉠  
 라. ㉢ → ㉡ → ㉣ → ㉤ → ㉠

## 제3과목 운영체제

41. 페이지 오류율(Page Fault Ratio)과 스래싱(Thrashing)에 대한 설명으로 옳은 것은?

- 가. 페이지 오류율이 크면 스래싱이 많이 발생한 것이다.  
 나. 페이지 오류율과 스래싱은 전혀 관계가 없다.  
 다. 스래싱이 많이 발생하면 페이지 오류율이 감소한다.  
 라. 다중 프로그래밍의 정도가 높을수록 페이지 오류율과 스래싱이 감소한다.

42. UNIX에서 파일 조작을 위한 명령으로 거리가 먼 것은?

- 가. cp                                  나. mv  
 다. ls                                  라. rm

43. 파일 디스크립터의 내용으로 옳지 않은 것은?

- 가. 오류 발생 시 처리 방법  
 나. 보조기억장치 정보  
 다. 파일 구조  
 라. 접근 제어 정보

44. 운영체제의 작업 수행 방식에 관한 설명으로 옳지 않은 것은?

- 가. 하나의 컴퓨터 시스템에서 여러 프로그램들이 같이 컴퓨터 시스템에 입력되어 주기억장치에 적재되고, 이들이 처리장치를 번갈아 사용하며 실행하도록 하는 것을 다중프로그래밍(Multiprogramming) 방식이라고 한다.  
 나. 한 대의 컴퓨터를 동시에 여러 명의 사용자가 대화식으로 사용하는 방식으로 처리 속도가 매우 빨라 사용자는 독립적인 시스템을 사용하는 것으로 인식하는 것을 일괄처리(Batch Processing) 방식이라고 한다.  
 다. 한 대의 컴퓨터에 중앙처리장치(CPU)가 2개 이상 설치되어 여러 명령을 동시에 처리하는 것을 다중프로세싱(Multi-processing) 방식이라고 한다.  
 라. 여러 대의 컴퓨터들에 의해 작업들을 나누어 처리하여 그 내용이나 결과를 통신망을 이용하여 상호 교환되도록 연결되어 있는 것을 분산처리(Distributed Processing) 방식이라고 한다.

45. 교착상태의 해결 방법 중 회피(Avoidance) 기법과 밀접한 관계가 있는 것은?

- 가. 점유 및 대기 방지      나. 비선점 방지  
다. 환경 대기 방지      라. 은행원 알고리즘 사용

46. 자식 프로세스의 하나가 종료될 때까지 부모 프로세스를 일시 중지시키는 유닉스 명령어는?

- 가. exit()      나. fork()  
다. exec()      라. wait()

47. 디스크 스케줄링 기법 중 SCAN을 사용하여 다음 작업 대기 큐의 작업을 모두 처리하고자 할 경우, 가장 최후에 처리되는 트랙은? (단, 현재 디스크 헤드는 50 트랙에서 40 트랙으로 이동해 왔다고 가정한다.)

작업 대기 큐 : 7, 55, 15, 38, 3

- 가. 3      나. 15      다. 38      라. 55

48. 스레드(Thread)에 관한 설명으로 옳지 않은 것은?

- 가. 스레드는 하나의 프로세스 내에서 병행성을 증대시키기 위한 메커니즘이다.  
나. 스레드는 프로세스의 일부 특성을 갖고 있기 때문에 경량(Light Weight) 프로세스라고도 한다.  
다. 스레드는 동일 프로세스 환경에서 서로 독립적인 다중 수행이 불가능하다.  
라. 스레드 기반 시스템에서 스레드는 독립적인 스케줄링의 최소 단위로서 프로세스의 역할을 담당한다.

49. 가상기억장치(Virtual Memory)에 대한 설명으로 거리가 먼 것은?

- 가. 보조기억장치의 일부 용량을 주기억장치처럼 가상으로 사용할 수 있도록 하는 개념이다.  
나. 별도의 주소 매핑 작업 없이 가상기억장치에 있는 프로그램을 주기억장치에 적재하여 실행할 수 있다.  
다. 가상기억장치의 구현은 일반적으로 페이징 기법과 세그먼테이션 기법을 이용한다.  
라. 주기억장치의 이용율과 다중 프로그래밍의 효율을 높일 수 있다.

50. 다음 설명에 해당하는 자원 보호 기법은?

접근 제어 행렬에 있는 각 행, 즉 영역을 중심으로 구성된 것으로서 각 사용자에게 대한 자격들로 구성되며, 자격은 객체와 그 객체에 허용된 연산 리스트이다.

- 가. Global Table      나. Capability List  
다. Access Control List      라. Lock/Key

51. 다음이 설명하는 디스크 스케줄링 기법은 무엇인가?

- 부하가 매우 큰 항공예약 시스템을 위해 개발되었으며 탐색시간(Seek Time)뿐만 아니라 회전 지연시간(Rotational Delay Time)의 최적화를 위해 개발된 기법이다.
- 헤드는 C-SCAN처럼 움직이며 예외적으로 모든 실린더는 그 실린더에 요청이 있던 없던 간에 전체 트랙이 한 바퀴 회전할 동안의 서비스를 받는다.

- 가. SSTF 기법  
나. N-단계 SCAN 기법  
다. FCFS 기법  
라. 에션바흐(Eschenbach) 기법

52. 모니터에 대한 설명으로 옳지 않은 것은?

- 가. 모니터의 경계에서 상호배제가 시행된다.  
나. 자료추상화와 정보은폐 기법을 기초로 한다.  
다. 공유 데이터와 이 데이터를 처리하는 프로시저로 구성된다.  
라. 모니터 외부에서도 모니터 내의 데이터를 직접 액세스할 수 있다.

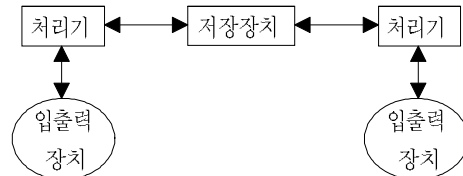
53. RR(Round Robin) 스케줄링에 대한 설명으로 옳지 않은 것은?

- 가. Time Slice를 크게 하면 입·출력 위주의 작업이나 긴급을 요하는 작업에 신속히 반응하지 못한다.  
나. Time Slice가 작을 경우 FCFS 스케줄링과 같아진다.  
다. Time Sharing System을 위해 고안된 방식이다.  
라. Time Slice가 작을수록 문맥교환 및 오버헤드가 자주 발생한다.

54. 디스크 공간 할당 기법 중 연속할당에 대한 설명으로 옳지 않은 것은?

- 가. 연속하는 논리적 블록들이 물리적으로 서로 인접하여 저장된다.  
나. 파일의 시작 주소와 크기만 기억하면 되므로 파일의 관리 및 구현이 용이하다.  
다. 파일의 크기가 자주 바뀌는 경우에는 구현이 어렵다.  
라. 단편화가 발생할 수 없으므로 주기적인 압축이 필요하다.

55. 다음 그림과 같은 구조를 갖는 시스템으로 가장 적합한 것은?



- 가. 약결합 다중 처리 시스템  
(Loosely-Coupled-Multiprocessing System)  
나. 강결합 다중 처리 시스템  
(Tightly-Coupled-Multiprocessing System)  
다. 단일버스 다중 처리 시스템  
(Single Bus Multiprocessing System)  
라. 공유버스 다중 처리 시스템  
(Shared Bus Multiprocessing System)

56. UNIX에 관한 설명으로 옳지 않은 것은?

- 가. 쉘(Shell)은 사용자와 시스템 간의 대화를 가능케 해주는 UNIX 시스템의 메커니즘이다.  
나. UNIX 시스템은 루트 노드를 시발로 하는 계층적 파일 시스템 구조를 사용한다.  
다. 커널(Kernel)은 프로세스 관리, 기억장치 관리, 입·출력 관리 등의 기능을 수행한다.  
라. UNIX 파일 시스템에서 각 파일에 대한 파일 소유자, 파일 크기, 파일 생성 시간에 대한 정보는 데이터 블록에 저장한다.

57. 빈 기억공간의 크기가 20K, 16K, 8K, 40K 일 때 기억장치 배치 전략으로 "Best Fit"을 사용하여 17K의 프로그램을 적재할 경우 내부 단편화의 크기는 얼마인가?

- 가. 3K      나. 23K      다. 64K      라. 67K

58. 분산 처리 시스템의 위상에 따른 분류에서 한 사이트의 고장이 다른 사이트에 영향을 주지 않지만, 중앙 사이트 고장 시 전체 시스템이 정지되는 형태는 무엇인가?

- 가. Tree 구조      나. Star 구조  
다. Ring 구조      라. Mesh 구조

59. 운영체제를 기능상 분류할 경우 “Control Program”과 “Process Program”으로 구분할 수 있다. 다음 중 “Control Program”에 해당하는 것으로만 짝지어진 것은?

- ① Supervisor Program  
② Language Translation Program  
③ Data Management Program  
④ Service Program  
⑤ Job Control Program  
⑥ Problem Program

가. ②, ④, ⑥                      나. ①, ③, ⑤  
다. ①, ⑤, ⑥                      라. ②, ③, ④

60. 다중 처리기의 운영체제 형태 중 주/종(Master/Slave) 처리기에 대한 설명으로 옳지 않은 것은?

가. 주프로세서만이 운영체제를 수행한다.  
나. 종프로세서는 입·출력 발생 시 주프로세서에게 서비스를 요청한다.  
다. 주프로세서가 고장나면 전체 시스템이 다운된다.  
라. 대칭적 구조를 갖는다.

#### 제4과목 소프트웨어 공학

61. 소프트웨어 재공학의 필요성이 대두된 가장 주된 이유는?

가. 요구사항 분석의 문제    나. 설계의 문제  
다. 구현의 문제              라. 유지보수의 문제

62. 소프트웨어 형상관리(Configuration Management)에 관한 설명으로 거리가 먼 것은?

가. 소프트웨어에서 일어나는 수정이나 변경을 알아내고 제어하는 것을 의미한다.  
나. 소프트웨어 개발의 전체 비용을 줄이고, 개발 과정의 여러 방해 요인이 최소화되도록 보증하는 것을 목적으로 한다.  
다. 형상관리를 위하여 구성된 팀을 책임 프로그래머 팀(Chief Programmer Team)이라고 한다.  
라. 형상관리에서 중요한 기술 중의 하나는 버전 제어 기술이다.

63. 소프트웨어 프로젝트 관리에 중요한 영향을 주는 3대 요소로 가장 타당한 것은?

가. 사람, 문제, 프로세스  
나. 문제, 프로젝트, 작업  
다. 사람, 문제, 도구  
라. 작업, 문제, 도구

64. 객체지향 개념에서 연관된 데이터와 함수를 함께 묶어 외부와 경계를 만들고 필요한 인터페이스만을 밖으로 드러내는 과정을 무엇이라고 하는가?

가. 메시지                      나. 캡슐화  
다. 상속                        라. 다형성

65. 소프트웨어 수명주기 모형 중 프로토타입 모형(Prototyping Model)의 가장 큰 장점은?

가. 위험요소가 쉽게 발견된다.  
나. 유지보수가 쉬워진다.  
다. 사용자 요구사항을 정확하게 파악할 수 있다.  
라. 소프트웨어 개발 일정을 정확하게 수립할 수 있다.

66. 람바우(Rumbaugh)의 OMT 기법에서 자료 흐름도와 가장 밀접한 관계가 있는 것은?

가. 객체 모델링                      나. 기능 모델링  
다. 동적 모델링                      라. 상속 모델링

67. 객체지향 기법에서 객체가 메시지를 받아 실행해야 할 객체의 구체적인 연산을 정의한 것은?

가. Entity                              나. Method  
다. Instance                              라. Class

68. 소프트웨어 품질 목표 중 다음 정의에 해당하는 것은?

정확하고 일관된 결과로 요구된 기능을 수행하는 시스템 능력

가. Efficiency                              나. Correctness  
다. Integrity                              라. Reliability

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

가. Process : 원  
나. Data Flow : 화살표  
다. Data Store : 삼각형  
라. Terminator : 사각형

70. CASE에 대한 설명으로 거리가 먼 것은?

가. 정형화된 메커니즘을 소프트웨어 개발에 적용하여 소프트웨어 생산성 향상을 구현한다.  
나. 시스템 개발 과정의 일부 또는 전체를 자동화시킨다.  
다. 개발 도구와 개발 방법론이 결합된 것이다.  
라. 도형, 목차, 총괄 도표, 상세 도표로 구성되어 전개된다.

71. 소프트웨어 프로젝트를 계획하려면 먼저 소프트웨어 범위(Scope)를 결정해야 한다. 다음 사항과 관계가 되는 범위 결정 요소는 무엇인가?

- 소프트웨어에 의해 간접적으로 제어되는 장치와 소프트웨어를 실행하는 하드웨어  
- 순서에 의거 절차적 운영상 소프트웨어 실행 절차  
- 기존의 소프트웨어나 새로운 소프트웨어를 연결하는 소프트웨어  
- 키보드나 기타 I/O 장치들을 통하여 소프트웨어를 사용하는 사람

가. 기능                              나. 성능  
다. 제약조건                              라. 인터페이스

72. 12개월이 기한인 S/W 프로젝트의 개발 일정이 지연되자, 2개월 남기고 사장은 프로젝트 관리자에게 3명의 인력을 추가 투입 하라고 지시했다. 이에 프로젝트 관리자는 반대했다. Brooks 법칙에 근거한 반대 이유로 가장 타당한 것은?

가. 인력관리가 어렵다.  
나. 비용 발생이 증가한다.  
다. 일정이 더 지연된다.  
라. 소프트웨어 질이 떨어진다.

73. LOC 기법에 의하여 예측된 총 라인 수가 50000일 경우 개발에 투입될 프로그래머의 수가 5명이고, 각 프로그래머의 평균 생산성이 월 당 500 라인일 때, 개발에 소요되는 기간은?

가. 10개월                              나. 20개월  
다. 25개월                              라. 50개월



## 74. 소프트웨어 재사용에 대한 설명으로 거리가 먼 것은?

- 가. 소프트웨어 품질을 향상시킨다.
- 나. 생산성이 증대된다.
- 다. 새로운 개발 방법 도입이 용이하다.
- 라. 개발 시간이 단축되고 비용이 감소한다.

## 75. 소프트웨어 재공학 활동 중 역공학에 해당하는 것은?

- 가. 소프트웨어 동작 이해 및 재공학 대상 선정
- 나. 소프트웨어 기능 변경 없이 소프트웨어 형태를 목적에 맞게 수정
- 다. 원시 코드로부터 설계 정보 추출 및 설계 절차 표현, 프로그램과 데이터 구조 정보 추출
- 라. 기존 소프트웨어 시스템을 새로운 기술 또는 하드웨어 환경에 이식

## 76. 바람직한 소프트웨어 설계 지침으로 볼 수 없는 것은?

- 가. 특정 기능을 수행하는 논리적 요소들로 분리되는 구조를 가지도록 한다.
- 나. 적당한 모듈의 크기를 유지한다.
- 다. 강한 결합도, 약한 응집도를 유지한다.
- 라. 모듈 간의 접속 관계를 분석하여 복잡도와 중복을 줄인다.

## 77. 소프트웨어 위기의 현상으로 보기 어려운 것은?

- 가. 소프트웨어 유지보수 비용의 증가
- 나. 소프트웨어 신뢰성, 정확성의 결여
- 다. 소프트웨어 개발 인력의 증가
- 라. 소프트웨어 개발 일정 준수의 어려움

## 78. 자료 사전(Data Dictionary)에서 자료의 반복을 나타내는 기호는?

- 가. ( )                      나. { }
- 다. [ ]                      라. \*\*

## 79. 화이트 박스 테스트 기법으로만 짝지어진 것은?

- ① Equivalence Partitioning Test
- ② Comparison Test
- ③ Basic Path Test
- ④ Condition Test
- ⑤ Data Flow Test
- ⑥ Cause-Effect Graphing Test
- ⑦ Loop Test

- 가. ①, ②, ⑦                      나. ②, ③, ④, ⑥, ⑦
- 다. ①, ②, ⑥                      라. ③, ④, ⑤, ⑦

## 80. 소프트웨어 공학에 대한 설명으로 가장 적합한 것은?

- 가. 소프트웨어의 제작부터 운영까지 생산성을 높이기 위해 기술적, 인간적인 요소에 대한 방법론을 제공한다.
- 나. 소프트웨어의 설계, 제작, 운영에 있어서 인간적인 요소를 배제한 프로그래밍 자체에 대한 공학적 연구를 의미한다.
- 다. 소프트웨어의 공학적이고 기술적인 영향을 사회 경제적인 시각에서만 설명한다.
- 라. 소프트웨어의 위기를 해결하기 위해서 현재 이미 해결된 문제들에 대해서 역사적 관점을 설명한다.

## 제5과목 데이터통신

## 81. 데이터 전송 제어 절차를 순서대로 바르게 나타낸 것은?

- ① 통신 회선 접속                      ② 정보 전송
- ③ 데이터 링크 해제                      ④ 데이터 링크 확립
- ⑤ 통신 회선 분리

- 가. ① → ④ → ② → ③ → ⑤
- 나. ⑤ → ④ → ③ → ① → ②
- 다. ② → ① → ③ → ④ → ⑤
- 라. ④ → ② → ① → ③ → ⑤

## 82. IP(Internet Protocol)의 설명 중 옳지 않은 것은?

- 가. 비연결형 전송 서비스를 제공한다.
- 나. 비신뢰성 전송 서비스를 제공한다.
- 다. 데이터그램이라는 데이터 전송 형식을 갖는다.
- 라. 스트림(Stream) 전송 기능을 제공한다.

## 83. 다음이 설명하고 있는 전송 기술은?

효율적인 전송을 위해 넓은 대역폭(고속 전송속도)을 가진 하나의 전송 링크를 통하여 여러 신호(데이터)를 동시에 실어 보내는 기술

- 가. 다중화                      나. 부호화
- 다. 양자화                      라. 압축화

## 84. 데이터 통신에서 오류 검출을 위해 사용되는 기법이 아닌 것은?

- 가. Parity Check
- 나. Block Sum Check
- 다. Cyclic Redundancy Check
- 라. Run Length Check

## 85. 데이터 전송 속도의 척도를 나타내는 것이 아닌 것은?

- 가. 변조 속도                      나. 데이터 신호 속도
- 다. 반송파 주파수 속도                      라. 베어러(Bearer) 속도

## 86. 다음이 설명하고 있는 오류 제어 방식은?

- 데이터 프레임의 정확한 수신 여부를 매번 확인하면서 다른 프레임을 전송해 나가는 오류 제어 방식이다.
- 송신기에서 하나의 데이터 프레임을 전송한 다음 반드시 확인 신호인 ACK를 기다려야 한다.
- 구현이 간단한 장점이 있으나, 데이터 프레임을 전송한 후, 응답 메시지를 수신하는데 걸리는 시간이 길어질수록 링크 사용 면에서 비효율적이다.

- 가. Stop-and-Wait ARQ
- 나. Go-back-N ARQ
- 다. Selective-Repeat ARQ
- 라. Forward-Stop ARQ

## 87. 데이터 통신 방식에 대한 설명으로 가장 옳은 것은?

- 가. 전이중 통신 방식은 통신 회선의 효율이 가장 높으며 전화 등에 사용된다.
- 나. 반이중 통신 방식의 예로는 TV, Radio, 무전기 등이 있다.
- 다. 단방향 통신 방식이나 반이중 통신 방식의 경우 반드시 4선식 회선이 필요하다.
- 라. 전이중 통신 방식은 양쪽 방향으로 신호의 전송이 가능하기는 하나 어떤 순간에는 반드시 한쪽 방향으로만 전송이 이루어지는 경우이다.

88. 음성 전화망과 같이 메시지가 전송되기 전에 발신지에서 목적지까지의 물리적 통신 회선 연결이 선행되어야 하는 교환 방식은?

- 가. 메시지 교환 방식
- 나. 데이터그램 방식
- 다. 회선 교환 방식
- 라. ARQ 방식

89. 다음 LAN의 구성 형태(Topology)와 매체 접근 제어(MAC ; Media Access Control) 방식의 연결이 잘못 짝지어진 것은?

- 가. Star형 - 회선 교환 방식
- 나. Ring형 - 토큰 링(Token Ring)
- 다. Bus형 - CSMA/CD 방식
- 라. Mesh형 - 레지스터 삽입 방식

90. HDLC는 링크 구성 방식에 따라 세 가지 동작 모드를 가진다. 이에 해당하지 않는 것은?

- 가. NBM 나. ABM
- 다. ARM 라. NRM

91. LAN의 매체 접근 제어 중 토큰 패싱 방식에 대한 설명으로 가장 옳은 것은?

- 가. 노드 사이의 접근 충돌을 막기 위해서 네트워크 접근을 교대로 허용한다.
- 나. 데이터 전송 시 반드시 토큰을 취득하여야 하고, 전송을 마친 후에는 토큰을 반납한다.
- 다. 노드 수가 많거나 데이터 양이 많은 경우에는 충돌이 일어나기 때문에 데이터의 손실이 매우 크다.
- 라. 우선 순위가 없기 때문에 모든 노드들이 균등한 전송 기회를 갖는다.

92. HDLC 프로토콜에 관한 설명이 아닌 것은?

- 가. 점대점 링크 및 멀티포인트 링크를 위하여 개발되었다.
- 나. 반이중 통신과 전이중 통신을 모두 지원한다.
- 다. 에러 제어를 위해서는 Stop-and-Wait 방식을 지원한다.
- 라. 슬라이딩 윈도우 방식에 의해 흐름 제어를 제공한다.

93. 동기식 시분할 다중화(Synchronous TDM)에 대한 설명으로 옳지 않은 것은?

- 가. 전송시간을 일정한 간격의 시간 슬롯(Time Slot)으로 나누고, 이를 주기적으로 각 채널에 할당한다.
- 나. 하나의 프레임은 일정한 수의 시간 슬롯(Time Slot)으로 구성된다.
- 다. 송신단에서는 각 채널의 입력 데이터를 각각의 채널 버퍼에 저장하고, 이를 순차적으로 읽어낸다.
- 라. 통계적 시분할 다중화(Statistical TDM) 방식 보다 전송 용량의 낭비가 적다.

94. 데이터 통신에서 사용되는 비동기 전송 방식에 대한 설명으로 옳지 않은 것은?

- 가. 수신기는 문자 단위의 재동기를 위해서 시작 비트(Start Bit)와 정지 비트(Stop Bit)를 사용한다.
- 나. 비동기식 전송은 단순하여 저렴하게 구현될 수 있으나 문자당 2~3 비트의 오버헤드(Overhead)가 요구된다.
- 다. 정지 비트는 휴지 상태와 같으므로 송신기는 다음 문자를 보낼 준비가 될 때까지 정지 비트를 계속 전송한다.
- 라. 신호 내에 클럭 정보를 포함하여 전송시키기 위해 맨체스터(Manchester) 부호화 방법을 사용한다.

95. 디지털 데이터를 아날로그 신호로 변환하는 방식이 아닌 것은?

- 가. ASK 나. FSK
- 다. PSK 라. PCM

96. 패킷(Packet) 교환과 관계가 없는 것은?

- 가. 패킷 단위로 데이터 전송
- 나. 고정적인 전송 대역폭
- 다. 가상회선 방식
- 라. 데이터그램 방식

97. HDLC(High-Level-Data Link Control)의 명령과 응답에 대한 프레임 종류가 아닌 것은?

- 가. Supervisory Frame
- 나. Handle Frame
- 다. Information Frame
- 라. Unnumbered Frame

98. OSI 참조 모델에서 통신회선을 통하여 비트 전송을 수행하기 위하여 전기적, 기계적인 제어 기능을 수행하는 계층은?

- 가. Physical Layer
- 나. Session Layer
- 다. Network Layer
- 라. Application Layer

99. TCP/IP 프로토콜 중 인터넷 계층에 대응하는 OSI 참조 모델의 계층은?

- 가. Physical Layer
- 나. Presentation Layer
- 다. Network Layer
- 라. Session Layer

100. 주파수 분할 다중화(FDM)에서 부채널 간의 상호 간섭을 방지하기 위한 것은?

- 가. 가드 밴드(Guard Band)
- 나. 채널(Channel)
- 다. 버퍼(Buffer)
- 라. 슬롯(Slot)

## 정답 및 해설

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

- 1 개체-관계 모델은 특정 DBMS에 맞게 작성하는 것이 아니므로 일대일(1:1) 관계뿐만 아니라 1:N, N:M의 관계를 자유롭게 표현할 수 있습니다. 개체-관계 모델의 개념과 특징을 숙지하세요.

**개체-관계 모델**

개념적 데이터 모델의 가장 대표적인 것으로, 1976년 피터 첸(Peter Chen)에 의해 제안된 이래 개체와 개체간의 관계를 기본 요소로 이용하여 현실 세계의 무질서한 데이터를 개념적인 논리 데이터로 표현하기 위한 방법으로 많이 사용되고 있다.

**개체-관계(Entity-Relationship) 모델의 특징**

- E-R 모델은 개체 타입(Entity Type)과 이들간의 관계 타입(Relationship Type)을 이용해 현실 세계를 개념적으로 표현한다.
  - E-R 모델에서는 데이터를 개체(Entity), 관계(Relationship), 속성(Attribute)으로 묘사한다.
  - E-R 모델은 특정 DBMS를 고려한 것은 아니다.
  - 최초에는 Entity, Relationship, Attribute와 같은 개념들로 구성되었으나 나중에는 일반화 계층 같은 복잡한 개념들이 첨가되어 확장된 모델로 발전했다.
- 2 로킹(Locking)은 하나의 트랜잭션이 데이터를 액세스 하는 동안 다른 트랜잭션이 그 데이터 항목을 액세스할 수 없도록 하는 방법입니다. 로킹 단위가 크면 로크 수가 적어 관리하기 쉽지만 병행성 수준이 낮아지고, 로킹 단위가 작으면 로크 수가 많아 관리하기는 복잡하지만 병행성 수준이 높아집니다. 나머지 보기로 제시된 로킹 단위의 특징을 정리하세요.

- 3 관계 데이터 모델에서 릴레이션(Relation)에 포함되어 있는 튜플(Tuple)의 수를 카디널리티(Cardinality) 또는 기수, 대응수라고 합니다. 관계형 데이터베이스에서 릴레이션을 구성하는 용어들은 매우 중요합니다. 확실히 정리하고 넘어가세요.

**릴레이션의 구성 요소**

- **튜플(Tuple)**
  - 튜플은 릴레이션을 구성하는 각각의 행을 말한다.
  - 튜플은 속성의 모임으로 구성된다.
  - 파일 구조에서 레코드와 같은 의미이다.
  - 튜플의 수를 카디널리티(Cardinality) 또는 기수, 대응수라고 한다.
- **속성(Attribute)**
  - 속성은 데이터베이스를 구성하는 가장 작은 논리적 단위이다.
  - 파일 구조상의 데이터 항목 또는 데이터 필드에 해당된다.
  - 속성은 개체의 특성을 기술한다.
  - 속성의 수를 디그리(Degree) 또는 차수라고 한다.
- **도메인(Domain)**

- 도메인은 하나의 애틀리뷰트가 취할 수 있는 같은 타입의 원자(Atomic)값 들의 집합이다.
- 도메인은 실제 애틀리뷰트 값이 나타날 때 그 값의 합법 여부를 시스템이 검사하는 데에도 이용된다.
- **릴레이션 인스턴스** : 데이터 개체를 구성하고 있는 속성들에 데이터 타입이 정의되어 구체적인 데이터 값을 갖고 있는 것을 말한다.

- 4 둘 이상의 트랜잭션이 동시에 병행 실행되는 경우 어느 하나의 트랜잭션 실행 중에 다른 트랜잭션의 연산이 끼어들 수 없습니다. 중요해요! 트랜잭션의 4가지 특성을 꼭 숙지하세요.

**트랜잭션의 특징**

- **Atomicity(원자성)**
  - 트랜잭션의 연산은 데이터베이스에 모두 반영되든지 아니면 전혀 반영되지 않아야 한다.
  - 트랜잭션 내의 모든 명령은 반드시 완벽히 수행되어야 하며, 모두가 완벽히 수행되지 않고 어느 하나라도 오류가 발생하면 트랜잭션 전부가 취소되어야 한다.
- **Consistency(일관성)**
  - 트랜잭션이 그 실행을 성공적으로 완료하면 언제나 일관성 있는 데이터베이스 상태로 변환한다.
  - 시스템이 가지고 있는 고정 요소는 트랜잭션 수행 전과 트랜잭션 수행 완료 후의 상태가 같아야 한다.
- **Isolation(독립성, 격리성)**
  - 둘 이상의 트랜잭션이 동시에 병행 실행되는 경우 어느 하나의 트랜잭션 실행 중에 다른 트랜잭션의 연산이 끼어들 수 없다.
  - 수행 중인 트랜잭션은 완전히 완료될 때까지 다른 트랜잭션에서 수행 결과를 참조할 수 없다.
- **Durability(영속성, 지속성)** : 성공적으로 완료된 트랜잭션의 결과는 영구적으로 반영되어야 한다.

- 5 SQL의 DROP 문은 스키마, 테이블, 뷰 등의 제거에 사용되는 명령이고, CASCADE 옵션은 삭제할 요소를 참조하는 다른 모든 개체를 함께 삭제하는 옵션입니다. 즉 인사 테이블을 참조하는 테이블과 인사 테이블을 모두 제거합니다. DROP 이 개체를 제거하는 명령이라는 것과 DROP에 사용되는 옵션에 대해서 알아두세요.

**DROP 옵션**

- **CASCADE 옵션** : 삭제할 요소를 참조하는 다른 모든 개체를 함께 삭제한다. 즉 Main Table의 데이터 삭제 시 각 외래키에 대해 부합되는 모든 데이터를 삭제하는 참조 무결성의 법칙을 설정한다.
  - **RESTRICTED 옵션** : 삭제할 요소를 다른 개체가 참조 중일 때는 삭제를 취소한다.
- 6 SQL 문장은 절별로 분리하여 이해하면 쉽습니다.



- **SELECT 사원명** : 사원명을 표시합니다.
- **FROM 회사원** : '회사원' 테이블의 자료를 검색합니다.
- **WHERE 연락처번호 IS NOT NULL** : '연락번호'가 NULL이 아닌 자료를 검색합니다.

NULL 값을 질의할 때는 IS NULL, NULL 값이 아닐 경우는 IS NOT NULL을 사용한다는 것을 기억해 두세요.

- 7 하나의 뷰를 삭제하면 그 뷰를 기초로 정의된 다른 뷰도 자동으로 삭제됩니다. 자주 출제되는 내용입니다. 뷰의 개념 및 특징을 확실히 정리하세요.

#### 뷰(View)

- 뷰는 기본 테이블로부터 유도된 테이블이기 때문에 기본 테이블과 같은 형태의 구조를 사용하며, 조작도 기본 테이블과 거의 같다.
- 뷰는 가상 테이블이기 때문에 물리적으로 구현되어 있지 않다.
- 데이터의 논리적 독립성이 어느 정도 제공된다.
- 필요한 데이터만 뷰로 정의해서 처리할 수 있기 때문에 관리가 용이하고 명령문이 간단해진다.
- 뷰를 통해서만 데이터에 접근하게 하면 뷰에 나타나지 않는 데이터를 안전하게 보호하는 효율적인 기법으로 사용할 수 있다.
- 기본 테이블의 기본키를 포함한 속성(열) 집합으로 뷰를 구성해야만 삽입, 삭제, 갱신 연산이 가능하다.
- 정의된 뷰는 다른 뷰의 정의에 기초가 될 수 있다.
- 하나의 뷰를 삭제하면 그 뷰를 기초로 정의된 다른 뷰도 자동으로 삭제된다.

- 8 스키마의 평가 및 정제는 논리적 설계 단계에서 수행합니다. 물리적 설계에서는 물리적 설계 단계의 특징과 함께 물리적 설계 시 고려할 사항도 종종 출제됩니다. 물리적 설계 시 고려할 사항을 정리하세요.

#### 물리적 설계 시 고려사항

- 저장 레코드 양식 설계
- 접근 경로 설계
- 레코드 집중의 분석 및 설계
- 인덱스의 구조
- 레코드 크기
- 파일에 존재하는 레코드 개수
- 파일에 대한 트랜잭션의 갱신과 참조 성향
- 시스템 운용 시 파일 크기의 변화 가능성

- 9 데이터 회복과 병행 수행 제어에 사용되는 데이터베이스 언어는 DCL입니다. DDL의 용도와 DDL의 3가지 유형을 꼭 기억해 두세요.

#### 데이터베이스 언어 DDL(데이터 정의어)

- DDL(Data Define Language)은 SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의하거나 변경 또는 삭제할 때 사용하는 언어이다.
- 데이터베이스 관리자나 데이터베이스 설계자가 사용한다.
- 데이터 정의어(DDL)의 3가지 유형
  - **CREATE** : SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의함
  - **ALTER** : TABLE에 대한 정의를 변경하는 데 사용함
  - **DROP** : SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 삭제함

- 10 데이터베이스 보안을 위한 사용자들의 권한 부여는 DBMS라는 소프트웨어 시스템이 자체 결정하는 것이 아니고, 사람인 관리자의 정책 결정에 의해서 정해집니다. 똑같은 형태로 몇

번 출제된 문제입니다. 보기로 제시된 내용을 통해 데이터베이스 보안의 의미를 정리해 두세요.

- 11 두 릴레이션 사이에 명세된 것으로 두 릴레이션의 튜플들 사이에서 일관성을 유지하는 데 사용되는 제약 조건은 참조 무결성 제약 조건입니다. 자주 출제되는 내용입니다. 특히 참조 무결성과 개체 무결성은 매회 빠지지 않고 출제되고 있음을 유념하세요.

#### 무결성의 종류

- **참조 무결성** : 외래키 값은 NULL이거나 참조 릴레이션의 기본키 값과 동일해야 한다는 규정. 즉 릴레이션은 참조할 수 없는 외래키 값을 가질 수 없다는 규정
- **개체 무결성** : 기본 테이블의 기본키를 구성하는 어떤 속성도 NULL 값을 가질 수 없다는 규정
- **널 무결성** : 테이블의 특정 속성(Attribute) 값이 NULL이 될 수 없도록 하는 규정
- **고유 무결성** : 테이블의 특정 속성에 대해서 각 튜플이 갖는 값들이 서로 달라야 한다는 규정
- **도메인 무결성** : 특정 속성의 값이, 그 속성이 정의된 도메인에 속한 값이어야 한다는 규정
- **키 무결성** : 하나의 테이블에는 적어도 하나의 키가 존재해야 한다는 규정
- **관계 무결성** : 릴레이션에 어느 한 튜플의 삽입 가능 여부 또는 한 릴레이션과 다른 릴레이션의 튜플들 사이의 관계에 대한 적절성 여부를 지정한 규정

- 12 보기의 SQL 중 DDL에 속하지 않는 것은 DELETE입니다. DELETE는 DML에 속합니다. 이 문제를 틀렸다면 9번 문제의 해설을 통해 DDL의 개념과 3가지 유형을 다시한번 정리하세요.

- 13 시스템 카탈로그는 메타 데이터를 갖고 있는 시스템 데이터베이스입니다. 나머지 보기가 틀린 이유를 확인하고 카탈로그의 의미와 특징을 꼭 숙지해 두세요.

나. 일반 사용자는 시스템 카탈로그의 내용을 갱신할 수 없습니다.

다. 시스템 카탈로그는 테이블이 아니라 데이터베이스를 대상으로 생성됩니다.

라. 카탈로그는 DBMS가 스스로 생성하고, 유지합니다.

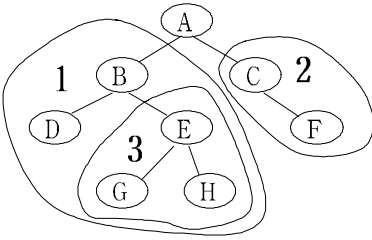
#### 시스템 카탈로그시스템 카탈로그(System Catalog)의 의미

- 시스템 카탈로그는 시스템 그 자체에 관련이 있는 다양한 객체에 관한 정보를 포함하는 시스템 데이터베이스이다.
- 시스템 카탈로그는 데이터베이스에 포함되는 모든 데이터 객체에 대한 정의나 명세에 관한 정보를 유지 관리하는 시스템 테이블이다.
- 데이터 정의어의 결과로 구성되는 기본 테이블, 뷰, 인덱스, 패키지, 접근 권한 등의 데이터베이스 구조 및 통계 정보를 저장한다.
- 카탈로그들이 생성되면 자료 사전(Data Dictionary)에 저장되기 때문에 좁은 의미로는 카탈로그를 자료 사전이라고도 한다.
- 카탈로그에 저장된 정보를 메타 데이터(Meta-Data)라고 한다.

#### 시스템 카탈로그의 특징

- 카탈로그 자체도 시스템 테이블로 구성되어 있어 일반 사용자도 SQL을 이용하여 내용을 검색해 볼 수 있다.
- INSERT, DELETE, UPDATE문으로 카탈로그를 갱신하는 것은 허용되지 않는다.
- 카탈로그는 DBMS가 스스로 생성하고, 유지한다.

- 14 먼저 서브트리를 하나의 노드로 생각할 수 있도록 서브트리 단위로 묶습니다. Preorder와 Inorder와 Postorder로 운행하는 것도 알아야 합니다.



Preorder는 Root → Left → Right 이므로 A12가 됩니다.

- 1은 BD3이므로 ABD32가 됩니다.
- 2는 CF이므로 ABD3CF가 됩니다.
- 3은 EGH이므로 ABDEGHCF가 됩니다.

## 2진 트리의 운행법

### • Inorder

Inorder는 Left → Root → Right 이므로 1A2가 된다.

- 1은 DB3이므로 DB3A2가 된다.
- 2는 CF이므로 DB3ACF가 된다.
- 3은 GEH이므로 DBGEHACF가 된다.

### • Postorder

Postorder는 Left → Right → Root 이므로 12A가 된다.

- 1은 D3B이므로 D3B2A가 된다.
- 2는 FC이므로 D3BFCA가 된다.
- 3은 GHE이므로 DGHEBFCA가 된다.

- 15 한 릴레이션에 포함된 튜플 사이에는 순서가 없습니다. 중요한 내용입니다. 릴레이션의 특징은 무조건 암기하지 말고 다음에 주어진 예를 <학생> 릴레이션에 적용시켜 이해하면 쉽습니다.

## 릴레이션의 특징

### <학생>

학과	이름	학년	신장	학과
89001	홍길동	2	170	CD
89002	이순신	1	169	CD
87012	임격정	2	180	ID
86032	장보고	4	174	ED

- 한 릴레이션에 포함된 튜플들은 모두 상이하다.  
예) <학생> 릴레이션을 구성하는 홍길동 레코드는 홍길동에 대한 학적사항을 나타내는 것으로 <학생> 릴레이션 내에서는 유일하다.
- 한 릴레이션에 포함된 튜플 사이에는 순서가 없다.  
예) <학생> 릴레이션에서 홍길동 레코드와 임격정 레코드의 위치가 바뀌어도 상관없다.
- 튜플들의 삽입, 삭제 등의 작업으로 인해 릴레이션은 시간에 따라 변한다.  
예) <학생> 릴레이션에 새로운 학생의 레코드를 삽입하거나, 기존 학생에 대한 레코드를 삭제함으로써 테이블은 내용 면에서나 크기 면에서 변하게 된다.
- 릴레이션 스키마를 구성하는 속성들 간의 순서는 중요하지 않다.  
예) 학번, 이름 등의 속성을 나열하는 순서가 이름, 학번순으로 바뀌어도 데이터 처리에는 전혀 문제가 되지 않는다.
- 속성의 유일한 식별을 위해 속성의 명칭은 유일해야 하지만, 속성을 구성하는 값은 동일한 값이 있을 수 있다.  
예) 각 학생의 학년을 기술하는 속성인 '학년'은 다른 속성 명들과 구분되어 유일해야 하지만 '학년' 속성에는 2,

- 1, 2, 4 등이 입력된 것처럼 동일한 값이 있을 수 있다.
- 릴레이션을 구성하는 튜플을 유일하게 식별하기 위해 속성들의 부분집합을 키(Key)로 설정한다.  
예) <학생> 릴레이션에서는 '학번'이나 '성명'이 튜플들을 구분하는 유일한 값인 키가 될 수 있다.
- 속성은 더 이상 쪼갤 수 없는 원자값만을 저장한다.  
예) '학년'에 저장된 1, 2, 4 등은 더 이상 세분화할 수 없다.

- 16 호스트 변수는 데이터베이스 필드명과 구분하기 위해 변수명 앞에 콜론(:)을 붙이므로 호스트 변수와 데이터베이스 필드의 이름이 동일해도 됩니다. Embedded-SQL의 정의와 특징을 숙지하세요.

## 내장 SQL의 정의

내장 SQL은 응용 프로그램내에 데이터베이스에서 사용하는 데이터를 정의하거나 질의하는 SQL 문장을 내포하여 프로그램이 실행될 때 함께 실행되도록 호스트 프로그램 언어로 만든, 프로그램에 삽입된 SQL이다.

## 내장 SQL의 특징

- 내장 SQL 실행문은 호스트 언어에서 실행문이 나타날 수 있는 곳이면 프로그램의 어느 곳에서나 사용할 수 있다.
- 일반 SQL문은 수행 결과로 여러 개의 튜플을 반환하는 반면, 내장 SQL은 단 하나의 튜플만을 반환한다.
- 내장 SQL문에 의해 반환되는 튜플은 일반 변수를 사용하여 저장할 수 있다.
- Host Program의 컴파일 시 내장 SQL문은 선행처리에 의해 분리되어 컴파일 된다.
- 호스트 변수와 데이터베이스 필드의 이름은 같아도 된다.
- 내장 SQL문에 사용된 호스트 변수의 데이터 타입은 이에 대응하는 데이터베이스 필드의 SQL 데이터 타입과 일치하여야 한다.
- 삽입 SQL문이 실행되면 SQL 실행의 상태가 SQL 상태 변수에 전달된다.

- 17 데이터베이스는 데이터 참조 시 데이터 위치나 주소가 아니라 값에 의해서 참조하는데, 이를 내용에 의한 참조라고 합니다. 최근 들어 자주 출제되고 있습니다. 데이터베이스의 특징을 정리하세요.

## 데이터베이스의 특성

- **실시간 접근성** : 수시적이고 비정형적인 질의(조회)에 대하여 실시간 처리(Real Time Processing) 응답이 가능해야 한다.
- **계속적인 변화** : 새로운 데이터의 삽입(Insertion), 삭제(Deletion), 갱신(Update)으로 항상 최신의 데이터를 유지해야 한다.
- **동시 공유** : 여러 사용자가 동시에 자기가 원하는 데이터를 이용할 수 있어야 한다.
- **내용에 의한 참조** : 데이터베이스에 있는 데이터를 참조할 때 데이터 레코드의 주소나 위치에 의해서가 아니라, 사용자가 요구하는 데이터 내용으로 데이터를 찾는다.

- 18 해싱을 이용한 파일 구조는 위치에 관계없이 한 번에 직접 접근이 가능한 직접(DAM) 파일입니다. 해싱의 개념과 해시 테이블에서 사용하는 용어를 숙지하세요.

## 해싱의 개요

- 해싱은 Hash Table이라는 기억공간을 할당하고, 해시 함수(Hash Function)를 이용하여 레코드 키에 대한 Hash Table 내의 Home Address를 계산한 후 주어진 레코드를 해당 기억장소에 저장하거나 검색 작업을 수행하는 방식이다.
- 해싱은 DAM(직접 접근) 파일을 구성할 때 사용되며, 접근

속도는 빠르나 기억공간이 많이 요구된다.

- 검색 속도가 가장 빠르다.
- 삽입, 삭제 작업의 빈도가 많을 때 유리한 방식이다.

#### 해시 테이블(Hash Table, 해시표)

- **버킷(Bucket)** : 하나의 주소를 갖는 파일의 한 구역을 의미하며, 버킷의 크기는 같은 주소에 포함될 수 있는 레코드 수를 의미함
- **슬롯(Slot)** : 한 개의 레코드를 저장할 수 있는 공간으로, n개의 슬롯이 모여 하나의 버킷을 형성함
- **Collision(충돌 현상)** : 서로 다른 두 개 이상의 레코드가 같은 주소를 갖는 현상
- **Synonym** : 같은 Home Address를 갖는 레코드들의 집합
- **Overflow** : 계산된 Home Address의 Bucket 내에 저장할 기억공간이 없는 상태로 Bucket을 구성하는 Slot이 여러 개일 때 Collision은 발생해도 Overflow는 발생하지 않을 수 있음

- 19 제 2정규형에서 제 3정규형이 되기 위한 작업은 이행적 함수 종속을 제거하는 것입니다. 중요해요! 정규화의 단계를 꼭 암기하세요.

#### 정규화 과정



#### 정규화 단계 암기 요령

정규화라는 출소자가 말했다.

두부이겨다줘는도부이결다조

도메인이 원자값

부분적 함수 종속 제거

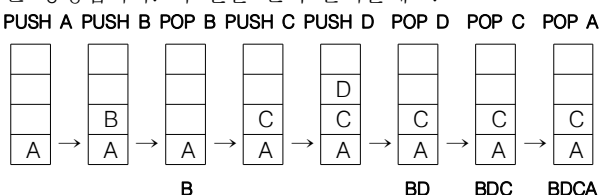
이행적 함수 종속 제거

결정자이면서 후보키가 아닌 것 제거

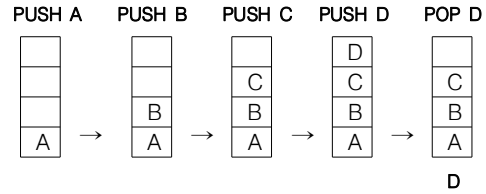
다치 종속 제거

조인 종속성 이용

- 20 A, B, C, D 순으로 입력된 상태에서는 D, A, B, C 순으로 출력할 수 없습니다. A, B, C, D가 입력된 후 D를 출력할 수 있으며, 이후 C, B를 출력해야만 A를 출력할 수 있으므로 D, A, B, C 순으로 출력할 수는 없습니다. PUSH는 스택에 자료를 입력하는 명령이고, POP는 스택에서 자료를 출력하는 명령입니다. 나 번을 먼저 살펴볼게요.

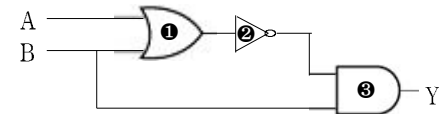


가 번은 D 출력후에 A를 출력해야 하는데, C, B를 출력하지 않고는 A를 출력할 수 없으므로 불가능합니다.



다, 라 번도 위와 같은 방법으로 스택에 자료를 넣었다 꺼내 보면서 출력이 될 수 있는지 확인해 보세요.

- 21 논리회로에 사용된 각 논리 게이트를 분리하여 논리식으로 표현한 후 1개의 논리식으로 합치면 됩니다.



문제의 논리회로를 논리식으로 표현하면 ①은  $A+B$  이고 ②에 의해 부정이 되므로  $(A+B)$ 입니다. ③은 ②  $\cdot$  B이므로 논리식은  $(A+B) \cdot B$ 가 됩니다. 종종 출제되는 문제입니다. 꼭 이해하고 넘어가세요.

- 22 RNI는 프로그램에 의해 제어되는 동작이 아닙니다. 처음 출제된 문제입니다. 문제와 답만 기억하고 넘어가세요.

- 23 결선 게이트란 각 게이트들의 출력 단자들을 직접 선으로 연결하여 논리 기능을 발휘하도록 하는 게이트로서 많은 논리 기능을 부여할 수 있습니다. 처음 출제된 문제입니다. 보기에 주어진 결선 게이트의 특징만 간단히 읽어보고 넘어가세요.

- 24 비트 슬라이스 마이크로프로세서는 Processor Unit, Micro-program Sequencer, Control Memory가 각각 다른 IC로 구성된 프로세서를 말합니다. 역시 처음 출제된 문제입니다. 이 문제에서는 비트 슬라이스 마이크로프로세서의 개념만 기억하고 넘어가세요.

- 25 제어점은 제어신호에 의해 동작되는 장치에서 그 신호를 받아서 데이터 입·출력을 시작하는 지점인 In-Gate와 Out-Gate를 말합니다. 중앙처리장치가 FETCH 상태인 경우에 제어점을 제어하는 것은 명령어입니다. 메이저 스테이트의 각 단계에서 제어점을 제어하는 것이 무엇인지 알아두세요.

- **Fetch** : 명령어
- **Indirect** : 유효주소
- **Execute** : 명령어의 연산자
- **Interrupt** : Interrupt 체제에 따라 달라짐

- 26 플립플롭 중 입력 단자가 하나이며, "1" 이 입력될 때마다 출력 단자의 상태가 바뀌는 것은 T 플립플롭입니다. RS, D, JK 플립플롭을 서로 구별할 수 있도록 특징을 알아두세요.

- **RS 플립플롭** : 플립플롭의 기본으로, S와 R선의 입력을 조절하여 임의의 Bit 값을 그대로 유지시키거나, 무조건 0또는 1의 값을 기억시키기 위해서 사용
- **D 플립플롭** : RS 플립플롭의 R선에 인버터(Inverter)를 추가하여 S선과 하나로 묶어서 입력선을 하나만 구성한 플립플롭
- **JK 플립플롭** : RS 플립플롭에서  $S=R=1$ 일 때 동작되지 않는 결점을 보완한 플립플롭으로 모든 플립플롭의 기능을



포함함

27 페이지는 주소공간에서, 그리고 블록은 기억공간에서 사용되는 용어입니다. 즉  $64K/512=128$  페이지,  $4K/512 = 8$ 블록( $K=1024$ 입니다.)입니다. 이런 계산 문제는 의외로 쉽습니다. 놓치지 마세요.

28 가상 메모리를 사용한 컴퓨터에서 Page Fault가 발생하면 요구된 Page가 주기억장치로 옮겨질 때까지 프로그램 수행이 중단됩니다. 이때 교체할 페이지를 결정해서 보조기억장치의 이전 위치에 기억시키고 새로운 페이지를 교체한 페이지의 위치에 놓는 것을 스테이징이라고 합니다. 이 문제에서는 가상 메모리를 사용한 컴퓨터에서 Page Fault가 발생하면 요구된 Page가 주기억장치로 옮겨질 때까지 프로그램 수행이 중단된다는 것만 기억하고 넘어가세요.

29 CISC는 RISC에 비해 속도가 느립니다. 이번에는 처음 출제된 문제가 많네요. CISC와 RISC의 특징을 비교해서 파악해 두세요.

CISC와 RISC의 비교

구분	명령어	주소 지정	레지스터	전력 소모	처리 속도	용도
RISC	적음	간단	많음	적음	빠름	서버, 워크스테이션
CISC	많음	복잡	적음	많음	느림	개인용 컴퓨터(PC)

30 페이지 상태 중 명령어를 가져오기 위해 기억장치에 접근하는 것을 Fetch 상태라 합니다. 다른 보기가 틀린 이유를 확인하고, 무슨 단계를 말하는지 구분할 수 있을 정도로 페이지 스테이트 각 단계의 특징을 정리하세요.

나. 간접주소 지정방식의 경우에만 수행되는 것은 간접(Indirect) 사이클입니다.

다. 페이지 상태 중 현재 상태를 보관하기 위해 기억장치를 접근하는 사이클은 없습니다. 다만 인터럽트 사이클에서 현재 상태를 저장시키는 과정은 있습니다.

라. 명령어의 종류를 판별하는 것을 패치(Fetch) 사이클이라 합니다.

페이지 스테이트

· 인출 단계(Fetch Cycle)

- Fetch Cycle은 명령어를 주기억장치에서 중앙처리장치의 명령 레지스터로 가져와 해독하는 단계이다.
- 읽어와 해석된 명령어가 1 Cycle 명령이면 이를 수행한 후 다시 Fetch Cycle로 변천한다.

· 간접 단계(Indirect Cycle)

- Fetch 단계에서 해석된 명령의 주소부가 간접주소인 경우 수행된다.
- 이 사이클에서는 Fetch 단계에서 해석한 주소를 읽어온 후 그 주소가 간접주소이면 유효주소를 계산하기 위해 다시 Indirect 단계를 수행한다.

· 실행 단계(Execute Cycle)

- Fetch 단계에서 인출하여 해석한 명령을 실행하는 단계이다.
- Execute 단계에서는 플래그 레지스터의 상태 변화를 검사하여 Interrupt 단계로 변천할 것인지를 판단한다.
- Execute 단계에서는 Interrupt 요청신호를 나타내는 플래그 레지스터의 변화가 없으면 Fetch 단계로 변천한다.

· 인터럽트 단계(Interrupt Cycle)

- 인터럽트 발생시 복귀주소(PC)를 저장시키고, 제어 순서를 인터럽트 처리 프로그램의 첫 번째 명령으로 옮기는 단계이다.
- 인터럽트 단계를 마친 후에는 항상 Fetch 단계로 변천한다.

다.

31 제 5과목 데이터 통신에 나올 문제가 전자계산기 과목에 나왔네요. 한번 풀어 볼까요? 패리티 비트가 짝수인지 홀수인지 명시되어 있지 않으므로 두 가지 모두 구해봐야 합니다. 먼저 짝수 패리티로 해밍 코드를 풀어보겠습니다.

① 총 정보 비트 수가 4개이므로 패리티 비트가 들어갈 자리인 1, 2, 4자리를 비운 나머지 자리에 정보 비트를 기입합니다.

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

② 1번 비트를 결정하기 위해서 3, 5, 7번 비트를 이용하여 1의 비트 수가 짝수 개가 되도록 합니다.

0	H	1	H	0	0	1
---	---	---	---	---	---	---

3, 5, 7번 비트의 1의 개수는 짝수이므로 1번 비트를 0로 하여 전체를 짝수로 맞춥니다.

③ 2번 비트를 결정하기 위해서 3, 6, 7번 비트를 이용하여 1의 비트 수가 짝수 개가 되도록 한다.

0	0	1	H	0	0	1
---	---	---	---	---	---	---

3, 6, 7번 비트의 1의 개수는 짝수이므로 2번 비트를 0로 하여 전체를 짝수로 맞춥니다.

④ 4번 비트를 결정하기 위해서 5, 6, 7번 비트를 이용하여 1의 비트 수가 짝수 개가 되도록 합니다.

0	0	1	1	0	0	1
---	---	---	---	---	---	---

5, 6, 7 번의 1의 개수는 홀수 이므로 4번 비트를 1로 하여 전체를 짝수로 맞춥니다.

홀수 패리티로 구하는 방법은 ②, ③, ④번에서 1의 개수가 1이 되도록 1,2,4번 비트의 값을 정하면 됩니다. 그러면 결과는 '1110001'입니다.

오류 검출을 위한 패리티 비트 결정하는 방법을 알아야 문제를 풀 수 있는데, 한번 읽어보고 이해되지 않으면 1,2,4번 비트에 대한 결정 방법을 외워두세요.

오류 검출을 위한 패리티 비트 결정하는 방법

n번 패리티 비트를 결정하기 위해서는 n비트만큼을 포함하고, n비트씩 건너뛴 비트들을 대상으로 패리티 비트를 결정합니다. 즉 1번 패리티 비트를 결정하기 위해서는 1비트만큼을 포함하고 1비트씩 건너뛴 1, 3, 5, 7, ... 비트가 대상이 됩니다. 2번 비트를 결정하기 위해서는 2비트만큼을 포함하고 2비트씩 건너뛴 2, 3, 6, 7, 10, 11, ... 비트를 이용하고, 4번 비트를 결정하기 위해서는 4비트만큼을 포함하고 4비트씩 건너뛴 4, 5, 6, 7, 12, 13, 14, 15, ... 비트를 이용합니다.

32 소프트웨어 인터럽트 사용 시 가장 큰 장점은 우선 순위 변경이 쉽다는 것입니다. 하드웨어 방식이 소프트웨어 방식에 비해 빠르고, 비싸고, 복잡하고, 융통성이 없다는 것은 컴퓨터와 관련된 모든 분야에서 통용되는 차이점입니다. 두 가지 방식의 차이점을 꼭 파악해 두세요.

구분	하드웨어	소프트웨어
반응 속도	고속	저속
회로 복잡도	복잡	간단
경제성	비경제적	경제적
융통성	없음	있음

33 OP 코드(연산자부)의 크기(비트 수)는 표현할 수 있는 명령의 종류를 나타내는 것으로, OP 코드가 N Bit일 때 최대  $2^N$ 개의 명령어를 사용할 수 있습니다. 이 문제에서는 OP 코드가 4비트 이므로  $2^4(32)$ 개의 명령어를 생성할 수 있습니다. 명

령어의 구성 형식을 정리하세요.

### 명령어의 구성

컴퓨터에서 실행되는 명령어는 크게 연산자에 해당하는 연산자부(Operation 부)와 명령에 필요한 자료의 정보가 표시되는 자료(Operand) 부로 구성된다.

### 연산자부(OP-Code 부, Operation Code부)

- 연산자부는 수행해야 할 동작에 맞는 연산자를 표시한다.
- 연산자부의 크기(비트수)는 표현할 수 있는 명령의 종류를 나타내는 것으로, n Bit면 최대  $2^n$ 개의 명령어를 사용할 수 있다.

[예] 연산자부가 5Bit라면 최대  $2^5 = 32$ 개의 명령어(연산자)를 사용할 수 있다.

- 연산자부에는 주소부의 유효 주소가 결정되는 방법을 지정하기 위한 모드 비트를 추가하기도 한다. (0 : 직접, 1 : 간접)

### Operand 부(주소부)

- 주소부는 실제 데이터에 대한 정보를 표시하는 부분이다.
- 기억장소의 주소, 레지스터 번호, 사용할 데이터 등을 표시한다.
- 주소부의 크기는 메모리의 용량과 관계가 있다.

[예] 주소부가 16Bit라면  $2^{16} = 65,536 = 64K$ 의 메모리를 주소로 지정하여 사용할 수 있다.

- 34  $16K \times 32\text{Bit}$ 의 의미는 워드의 크기가 32비트인 기억장소가 16K개 있다는 의미입니다. 즉 주소를 지정하는 MAR은 16K를 지정할 수 있는 Bit가 필요하고, 내용을 담는 MBR은 32Bit입니다. 16은  $2^4$ 이고 K는  $2^{10}$ 이므로  $2^4 \times 2^{10}$ 은  $2^{14}$ , 즉 MAR은 14비트가 필요합니다. 워드의 길이는 워드를 담을 수 있는 MBR(DR), IR의 크기와 같고 주소선의 수는 주소를 지정할 때 사용하는 PC, MAR(AR)의 크기와 같다는 것도 알고 있어야 합니다.

- 35 스택이 사용되는 경우는 인터럽트가 발생할 때입니다. 스택의 응용 분야에 관한 문제는 전자계산기와 운영체제에서도 자주 출제되는 내용입니다. 꼭 암기하세요.

### Stack을 이용하는 예

- 부프로그램 호출시 복귀주소를 저장할 때
- 함수 호출의 순서 제어
- 인터럽트가 발생하여 복귀주소를 저장할 때
- 후위 표기법(Postfix Notation)으로 표현된 산술식을 연산할 때
- 0 주소지정방식 명령어의 자료 저장소
- 재귀(Recursive) 프로그램의 순서제어
- 컴파일러를 이용한 언어번역시

- 36 정전은 외부 인터럽트입니다. 내부 인터럽트의 원인과 종류를 꼭 알아두세요.

### 내부 인터럽트

- 잘못된 명령이나 데이터를 사용할 때 발생하며, 트랩(Trap)이라고도 부른다.
- 명령어 잘못에 의한 인터럽트 : 프로그램에서 명령어를 잘못 사용한 경우
- 프로그램 인터럽트(Program Interrupt) : 0으로 나누거나, Overflow 또는 Underflow가 발생한 경우

- 37 고속의 입·출력 장치에 사용되는 채널은 Selector 채널입니다. 중요합니다. 채널의 종류와 특징을 기억해 두세요.

### 채널의 종류

Selector Channel (선택 채널)	<ul style="list-style-type: none"> <li>• 고속 입·출력 장치(자기 디스크, 자기 테이프, 자기 드럼)와 입·출력하기 위해 사용한다.</li> <li>• 특정한 한 개의 장치를 독점하여 입·출력한다.</li> </ul>
Multiplexer Channel (다중 채널)	<ul style="list-style-type: none"> <li>• 저속 입·출력 장치(카드리더, 프린터)를 제어하는 채널이다.</li> <li>• 동시에 여러 개의 입·출력 장치를 제어한다.</li> </ul>
Block Multiplexer Channel	<ul style="list-style-type: none"> <li>• 고속 입·출력 장치를 제어하는 채널이다.</li> <li>• 동시에 여러 개의 입·출력 장치를 제어한다.</li> </ul>

- 38 스트로브 제어 방식에서는 전송을 시작한 송신장치가 버스에 놓인 데이터를 수신 장치가 받아 들였는지 여부를 알 수 없습니다. 자주 출제되는 내용은 아닙니다. 보기에 주어진 스트로브 제어 방식의 특징만 간단히 살펴보고 넘어가세요.

- 39 17개의 정보 중 하나를 선택한다는 것, 즉 0, 1, 2, ..., 16이면 17개의 정보이므로 16을 이진수로 표현할 때 몇 비트가 필요한가?와 같은 의미입니다. 16을 이진수로 표현하면 10000이므로 최소 5개의 비트가 필요합니다.

- 40 인스트럭션을 수행하는 순서는 무조건 외우지 말고, 명령어를 가져와서 해독한 후 실행한다는 것에 맞춰 이해하면 쉽게 기억됩니다. 또한 인터럽트 조사는 실행(Execute)한 다음에 수행한다는 것을 기억하세요. 순서를 꼭 이해하고 넘어가세요.

- ① **Fetch** : 주기억장치에서 명령어를 가져(Fetch)와 해독(Decoding) 함
- ② **Indirect** : Fetch 단계에서 해독한 명령이 간접주소일 자료가 있는 주소를 계산하기 위해서 수행함
- ③ **Execute** : 실행함. 실행을 마친 후 인터럽트 요청을 검사함
- ④ **Interrupt** : Execute 단계에서 인터럽트 요청이 있었으면 수행함

- 41 페이지 오류율은 프로그램 실행 시 참조할 페이지가 주기억 장치에 없는 비율을 의미하는 것으로, 페이지 오류율이 크면 프로세스의 처리 시간보다 페이지 교체 시간이 더 많아지는 스래싱 현상이 더 많이 발생하게 됩니다. 페이지 오류율과 스래싱의 관계를 정확히 숙지하고, 나머지 보기가 잘못된 이유를 파악해 두세요.

- 나, 다. 페이지 오류율이 많이 발생되면 스래싱 현상이 증가됩니다.
- 라. 다중 프로그래밍의 정도가 아주 높아지면 페이지 오류율과 스래싱이 증가됩니다.

- 42 ls는 현재 디렉터리 내의 파일 목록을 확인하는 명령어으로써, 파일 조작을 위한 명령어가 아닙니다. UNIX 명령어의 기능을 묻는 문제는 자주 출제됩니다. 나머지 명령어의 기능도 정확히 알아두세요.

- cp : 파일 복사
- mv : 파일 이동 또는 이름 변경
- rm : 파일 삭제

- 43 파일 디스크립터는 파일을 관리하기 위한 시스템(운영체제)이 필요로 하는, 파일에 대한 정보를 갖고 있는 제어 블록으로 파일 디스크립터의 내용이 아닌 것은 오류 발생 시 처리 방법입니다. 파일 디스크립터에 포함되는 정보가 아닌 것을 묻는 문제가 종종 출제됩니다. 파일 디스크립터에 포함되는



정보를 꼭 정리해 두세요.

· **파일 디스크립터에 포함되어 있는 정보** : 파일 이름, 보조 기억장치에서의 파일 위치, 파일 구조, 보조기억장치의 유형, 액세스 제어 정보, 파일 유형, 생성 날짜와 시간, 제거 날짜와 시간, 최종 수정 날짜 및 시간, 액세스한 횟수 등

- 44 나 번은 일괄 처리 방식이 아니라 시분할 시스템에 대한 설명입니다. 일괄 처리 시스템(Batch Processing System)은 일정량 또는 일정 기간 동안 데이터를 모아서 한꺼번에 처리하는 방식을 의미합니다. 동일한 형태로 종종 출제되는 문제입니다. 각 보기에 제시된 운영 방식의 특징을 구분하여 정리해 두세요.
- 45 교착 상태 해결 방안 중 회피 기법은, 교착 상태 발생 가능성을 인정하고 교착 상태가 발생하려고 할 때 교착 상태 가능성을 피해가는 기법으로, 대표적인 회피 기법에는 ‘은행원 알고리즘’이 있습니다. 은행원 알고리즘이 회피 기법인 것만 알면 풀 수 있는 문제가 출제됩니다. ‘은행원 알고리즘’이 회피 기법이라는 것을 꼭 기억하세요.
- 46 자식 프로세스의 하나가 종료될 때까지 부모 프로세스를 임시 중지시키는 유닉스 명령어는 wait( )입니다. UNIX 명령어의 기능을 묻는 문제는 자주 출제된다고 했죠? UNIX 명령어가 나올 때마다 각 명령어의 기능은 꼭 파악해 두세요.
- **exit( )** : 프로세스 수행을 종료함
  - **fork( )** : 새로운 프로세스를 생성함(하위 프로세스 호출, 프로세스 복제 명령)
  - **exec( )** : 새로운 프로세스를 수행함
- 47 SCAN은 현재 헤드의 위치에서 진행 방향이 결정되면 탐색 거리가 짧은 순서대로 그 방향의 모든 요청을 서비스하고, 끝까지 이동한 후 역방향의 요청 사항을 서비스하는 기법으로, 이동 순서는 40 → 38 → 15 → 7 → 3 → 55순입니다. 그러므로 가장 최후에 처리되는 트랙은 55 트랙입니다. 이동 순서는 스케줄링의 동작 원리를 기억하면 됩니다. 문제를 통해 SCAN 스케줄링의 동작 원리를 꼭 기억두세요.
- 48 스레드(Thread)는 프로세스 내에서의 작업 단위로, 동일 프로세스 환경에서 서로 독립적인 다중 수행이 가능합니다. 자주 출제되는 문제는 아닙니다. 보기로 제시된 스레드의 특징 정도만 알고 넘어가세요.
- 49 가상기억장치에 저장된 프로그램을 실행하려면 가상기억장치의 주소를 주기억장치의 주소로 바꾸는 주소 매핑 작업이 필요합니다. 가상기억장치의 전반적인 특징을 묻는 문제가 가끔 출제되니 정리해 두세요.
- 가상기억장치**
- 가상기억장치는 보조기억장치(하드디스크)의 일부를 주기억장치처럼 사용하는 것으로, 용량이 작은 주기억장치를 마치 큰 용량을 가진 것처럼 사용하는 기법이다.
  - 프로그램을 여러 개의 작은 블록 단위로 나누어서 가상기억장치에 보관해놓고, 프로그램 실행시 요구되는 블록만 주기억장치에 불연속적으로 할당하여 처리한다.
  - 주기억장치의 용량보다 큰 프로그램을 실행하기 위해 사용한다.
  - 주기억장치의 이용율과 다중 프로그래밍의 효율을 높일 수 있다.
  - 가상기억장치에 저장된 프로그램을 실행하려면 가상기억장치의 주소를 주기억장치의 주소로 바꾸는 주소 변환 작업이 필요하다.
  - 블록 단위로 나누어 사용하므로 연속 할당 방식에서 발생

할 수 있는 단편화를 해결할 수 있다.

- 가상기억장치의 일반적인 구현 방법에는 블록의 종류에 따라 페이징 기법과 세그먼테이션 기법으로 나눌 수 있다.
- 50 문제의 지문에 제시된 자원 보호 기법은 권한(자격) 리스트(Capability List)입니다. 자원 보호 기법의 종류를 구분하는 문제가 가끔 출제됩니다. 나머지 자원 보호 기법의 특징도 정리해 두세요.
- **접근 제어 행렬(Access Control Matrix)** : 자원 보호의 일반적인 모델로, 객체에 대한 접근 권한을 행렬로써 표시한 기법
  - **전역 테이블(Global Table)** : 가장 단순한 구현 방법으로, 3개의 순서쌍인 영역, 객체, 접근 권한의 집합을 목록 형태로 구성한 기법
  - **락-키(Lock-Key)** : 접근 제어 리스트와 권한 리스트를 절충한 기법
- 51 항공예약 시스템을 위해 개발되었으며, 탐색 시간과 회전 지연시간 최적화를 위해 개발된 디스크 스케줄링은 ‘에션바흐(Eschenbach) 기법’입니다. 디스크 스케줄링 기법에서는 각 기법을 찾는 문제와 계산 문제가 자주 출제되나 ‘에션바흐’에 대한 문제는 처음 출제되었습니다. 문제를 통해 에션바흐 기법의 특징을 정리하고, 나머지 디스크 스케줄링 기법도 서로 구분할 수 있도록 특징을 숙지해 두세요.
- **FCFS(First Come First Service)** : 가장 간단한 스케줄링으로, 디스크 대기 큐에 가장 먼저 들어온 트랙에 대한 요청을 먼저 서비스하는 기법
  - **SSTF(Shortest Seek Time First)** : 탐색 거리(Seek Distance)가 가장 짧은 트랙에 대한 요청을 먼저 서비스하는 기법
  - **SCAN** : SSTF가 갖는 탐색 시간의 편차를 해소하기 위한 기법으로, 현재 헤드의 위치에서 진행 방향이 결정되면 탐색 거리가 짧은 순서에 따라 그 방향의 모든 요청을 서비스하고, 끝까지 이동한 후 역방향의 요청 사항을 서비스함
  - **N-step SCAN** : SCAN 기법을 기초로 하며 어떤 방향의 진행이 시작될 당시에 대기 중이던 요청들만 서비스하고, 진행 도중 도착한 요청들은 한데 모아서 다음의 반대 방향 진행 때 서비스하는 기법
  - **에션바흐(Eschenbach) 기법** : 부하가 매우 큰 항공 예약 시스템을 위해 개발되었으며, 탐색 시간과 회전 지연 시간을 최적화하기 위한 최초의 기법
  - **SLTF(Shortest Latency Time First)** : 섹터 큐잉(Sector Queuing)이라고 하며, 회전 시간의 최적화를 위해 구현된 기법
- 52 모니터 내의 데이터는 모니터 외부에서 액세스 할 수 없습니다. 모니터의 특징을 묻는 문제가 종종 출제되니 꼭 정리하고 넘어가세요.
- 모니터(Monitor)**
- 동기화를 구현하기 위한 특수 프로그램 기법으로 특정 공유 자원을 프로세스에게 할당하는 데 필요한 데이터와 이 데이터를 처리하는 프로시저로 구성된다.
  - 자료 추상화와 정보 은폐 개념을 기초로 하며 공유 자원을 할당하기 위한 병행정 구조로 이루어져 있다.
  - 모니터 내의 공유 자원을 사용하려면 프로세스는 반드시 모니터의 진입부를 호출해야 한다.
  - 외부의 프로시저는 직접 액세스할 수 없다.
  - 모니터의 경계에서 상호 배제가 시행된다.
  - 모니터에는 한 순간에 하나의 프로세스만 진입하여 자원을 사용할 수 있다.

- 53 RR(Round-Robin) 스케줄링에서 FCFS 스케줄링과 같아질 경우는 Time Slice가 작을 경우가 아니라 Time Slice가 커질 경우입니다. Time Slice가 작을 경우와 커질 경우의 특징을 구분하는 문제가 종종 출제됩니다. 문제를 통해 RR 스케줄링의 특징을 정확하게 알아두세요.

- 54 연속 할당은 파일을 디스크의 연속된 기억 공간에 할당하는 방법으로, 파일의 생성과 삭제가 반복되면서 단편화가 발생합니다. 그러므로 주기적인 압축이 필요합니다. 자주 출제되는 문제는 아닙니다. 연속 할당의 특징을 한번 읽어보고 넘어가세요.

#### 연속 할당

- 파일을 디스크의 연속된 기억 공간에 할당하는 방법으로, 생성되는 파일 크기만큼의 공간이 있어야 한다.
- 논리적으로 연속된 레코드들이 물리적으로 인접한 공간에 저장되기 때문에 접근 시간이 빠르다.
- 디렉터리는 파일의 시작 주소와 길이에 대한 정보만 가지고 있으므로 디렉터리가 단순하고, 관리 및 구현이 용이하다.
- 파일 크기에 알맞은 연속 공간이 없을 경우 파일이 생성되지 않는다.
- 파일의 생성과 삭제가 반복되면서 단편화가 발생한다.
- 단편화를 줄이기 위해 재배치에 의한 주기적인 압축(Compaction)이 필요하다.
- 파일의 크기가 시간에 따라 변경될 경우 구현하기가 어렵다.

- 55 그림과 같이 여러 개의 처리기(프로세서)가 하나의 저장장치를 공유하여 사용하는 시스템은 강결합 다중 처리 시스템(Tightly-coupled Multiprocessing System)입니다. 강결합과 약결합 시스템의 특징을 구분하는 문제가 종종 출제되었으나 그림으로 출제된 경우는 처음입니다. 각 시스템의 특징을 정확히 숙지하여 다양한 형태의 문제에 대비하세요.

#### 약결합 시스템(Loosely Coupled System)

- 각 프로세서마다 독립된 메모리를 가진 시스템으로, 분산 처리 시스템이라고도 한다.
- 둘 이상의 독립된 컴퓨터 시스템을 통신망(통신 링크)을 통하여 연결한 시스템이다.
- 각 시스템마다 독자적인 운영체제를 가지고 있다.
- 각 시스템은 독립적으로 작동할 수도 있고, 필요한 경우에는 상호 통신을 할 수도 있다.
- 프로세서 간의 통신은 메시지 전달이나 원격 프로시저 호출을 통해서 이루어진다.
- 각 시스템마다 독자적인 운영이 가능하므로 프로세서 간의 결합력이 약하다.

#### 강결합 시스템(Tightly Coupled System)

- 동일 운영체제 하에서 여러 개의 프로세서가 하나의 메모리를 공유하여 사용하는 시스템으로, 다중(병렬) 처리 시스템이라고도 한다.
- 하나의 운영체제가 모든 프로세서와 시스템 하드웨어를 제어한다.
- 프로세서 간의 통신은 공유 메모리를 통해서 이루어진다.
- 하나의 메모리를 사용하므로 프로세서 간의 결합력이 강하다.
- 공유 메모리를 차지하려는 프로세서 간의 경쟁을 최소화해야 한다.

- 56 UNIX 파일 시스템에서 각 파일에 대한 정보(소유자, 크기, 생성 시간 등)는 데이터 블록이 아니라 I-node에 저장됩니다.

다. UNIX에서는 셸과 커널의 역할과 파일 시스템의 구조를 묻는 문제가 자주 출제됩니다. 보기를 통해 셸(Shell)과 커널(Kernel)의 역할을 기억하고, UNIX 파일 시스템의 구조를 정리해 두세요.

#### UNIX 파일 시스템의 구조

부트 블록	부팅시 필요한 코드를 저장하고 있는 블록
슈퍼 블록	전체 파일 시스템에 대한 정보를 저장하고 있는 블록
I-node 블록(Index-node)	<ul style="list-style-type: none"> <li>• 각 파일이나 디렉터리에 대한 모든 정보를 저장하고 있는 블록</li> <li>• 정보 : 파일 소유자의 사용자 번호(UID) 및 그룹 번호(GID), 파일 크기, 파일 type(일반·디렉터리·특수 파일 등), 생성 시기, 최종 변경 시기, 최근 사용 시기, 파일의 번호 권한, 파일 링크 수, 데이터가 저장된 블록의 시작 주소</li> </ul>
데이터 블록	디렉터리별로 디렉터리 엔트리와 실제 파일에 대한 데이터가 저장된 블록

- 57 최적 적합(Best Fit)은 프로그램이나 데이터가 들어갈 수 있는 크기의 빈 영역 중에서 단편화를 가장 작게 남기는 분할 영역에 배치시키는 방법으로, 17K의 프로그램은 20K 영역에 적재됩니다. 내부 단편화는 분할된 영역이 할당될 프로그램의 크기보다 크기 때문에 프로그램이 할당된 후 사용되지 않고 남아 있는 빈 공간을 의미하므로, 여기서 내부 단편화는 3K(20K-17K)가 됩니다. 3가지 배치 전략과 내부 단편화, 외부 단편화와 관련된 문제가 자주 출제됩니다. 각각의 의미를 파악해 두세요.

#### 배치 전략

- **최초 적합(First Fit)** : 프로그램이나 데이터가 들어갈 수 있는 크기의 빈 영역 중에서 첫 번째 분할 영역에 배치시키는 방법
- **최적 적합(Best Fit)** : 프로그램이나 데이터가 들어갈 수 있는 크기의 빈 영역 중에서 단편화를 가장 작게 남기는 분할 영역에 배치시키는 방법
- **최악 적합(Worst Fit)** : 프로그램이나 데이터가 들어갈 수 있는 크기의 빈 영역 중에서 단편화를 가장 많이 남기는 분할 영역에 배치시키는 방법

#### 단편화

- **내부 단편화** : 분할된 영역이 할당될 프로그램의 크기보다 크기 때문에 프로그램이 할당된 후 사용되지 않고 남아 있는 빈 공간
- **외부 단편화** : 분할된 영역이 할당될 프로그램의 크기보다 작기 때문에 프로그램이 할당될 수 없어 사용되지 않고 빈 공간으로 남아 있는 분할된 전체 영역

- 58 중앙 사이트 고장 시 전체 시스템이 정지되는 형태는 성(Star)형 연결 구조입니다. 네트워크 위상의 특징을 구분하는 문제는 자주 출제됩니다. 각 위상을 구분할 수 있도록 특징을 정리해 두세요.

#### 분산 운영체제의 구조

- **완전 연결(Fully Connection)형** : 각 사이트들이 시스템 내의 다른 모든 사이트들과 직접 연결된 구조로, 사이트의 수가 n개이면 링크(연결) 수는  $n(n-1)/2$ 개가 됨
- **트리(Tree) 또는 계층(Hierarchy)형** : 각 사이트들이 트리 형태로 연결된 구조로, 기본 비용은 부분 연결형보다 적게 들고, 통신 비용은 트리의 깊이에 비례함
- **스타(Star)형=성형** : 모든 사이트가 하나의 중앙 사이트에

직접 연결되어 있고, 그 외 다른 사이트와는 연결되어 있지 않은 구조

- **다중 접근 버스 연결(Multi Access Bus Connection)형** : 시스템 내의 모든 사이트들이 공유 버스에 연결된 구조로, 기본 비용은 사이트 수에 비례하고, 통신 비용은 일반적으로 저렴함
- **부분 연결(Partially Connection)형** : 시스템 내의 일부 사이트들 간에만 직접 연결하는 것으로, 직접 연결되지 않은 사이트는 연결된 다른 사이트를 통해 통신하는 구조

- 59 운영체제는 기능상 제어 프로그램(Control Program)과 처리 프로그램(Process Program)으로 분류할 수 있고, 제어 프로그램은 다시 감시 프로그램(Supervisor Program)과 작업 제어 프로그램(Job Control Program), 데이터 관리 프로그램(Data Management Program)으로 구분할 수 있습니다. 언어 번역 프로그램(Language Translation Program), 서비스 프로그램(Service Program), 문제 프로그램(Problem Program)은 처리 프로그램입니다. 제어 프로그램의 종류뿐만 아니라 종류별 기능도 구분하여 정리해 두세요.

#### 제어 프로그램

- **감시 프로그램(Supervisor Program)** : 제어 프로그램 중 가장 중요한 역할을 담당하는 것으로, 각종 프로그램의 실행과 시스템 전체의 작동 상태를 감시·감독하는 프로그램
- **작업 제어 프로그램(Job Control Program)** : 어떤 업무를 처리하고 다른 업무로의 이행을 자동으로 수행하기 위한 준비 및 그 처리에 대한 완료를 담당하는 프로그램으로, 작업의 연속 처리를 위한 스케줄 및 시스템 자원 할당 등을 담당함
- **데이터 관리 프로그램(Data Management Program)** : 주기억장치와 보조기억장치 사이의 데이터 전송과 보조기억장치의 자료 갱신 및 유지 보수 기능을 수행하는 프로그램

- 60 주/중 처리기 시스템은 주프로세서만 입·출력을 수행하므로 대칭 구조가 아니라 비대칭 구조입니다. 주/중(Master/Slave) 처리기 시스템에서의 주/중 처리기의 역할을 중심으로 특징을 정리해 두세요.

#### Master/Slave(주/중) 처리기

- 하나의 프로세서를 Master(주프로세서)로 지정하고, 나머지는 Slave(종프로세서)로 지정하는 구조이다.
- 주프로세서가 고장나면 전체 시스템이 다운된다.
- 주프로세서만 입·출력을 수행하므로 비대칭 구조를 갖는다.
- **주프로세서** : 입·출력과 연산을 담당하며, 운영체제를 수행함
- **종프로세서** : 연산만 담당하며, 입·출력 발생 시 주프로세서에게 서비스를 요청함

- 61 소프트웨어 제공학은 유지보수 생산성 향상을 통해 소프트웨어 위기를 해결하는 방법입니다. 즉 소프트웨어 제공학의 필요성이 대두된 가장 주된 이유가 소프트웨어의 유지보수 문제 때문이라는 것입니다. 문제와 답만 기억하고 넘어가세요.

- 62 책임 프로그래머 팀은 프로젝트 구성의 한 형태로, 한 관리자가 의사 결정을 하고 팀 구성원들은 그 결정에 따르는 구성 방식입니다. 책임 프로그래머 팀은 단지 프로젝트 구성의 한 형태이지 형상관리와는 아무런 상관이 없습니다. 형상관리는 의미를 묻는 정도의 내용만 출제되니 나머지 보기를 통해 형상관리의 의미 및 특징을 간단히 정리해 두세요.

- 63 소프트웨어 프로젝트 관리에 중요한 영향을 주는 3대 요소는 사람(People), 문제(Problem), 프로세스(Process)입니다.

3P의 종류를 묻는 문제가 종종 출제되므로 명확히 알아두세요.

- 64 데이터와 함수를 함께 묶어 외부와 경계를 만들고 필요하면 인터페이스만을 밖으로 드러내는 과정을 ‘캡슐화’라고 합니다. 캡슐화의 의미를 묻는 문제가 종종 출제됩니다. 문제를 통해 캡슐화의 의미를 기억하고, 나머지 용어는 서로 구분할 수 있을 정도로 의미를 파악해 두세요.

• **메시지(Message)** : 객체들 간에 상호작용을 하는데 사용되는 수단으로, 객체의 메소드(동작, 연산)를 일으키는 외부의 요구 사항

• **상속성** : 이미 정의된 상위 클래스(부모 클래스)의 모든 속성과 연산을 하위 클래스가 물려받는 것

• **다형성(Polymorphism)** : 메시지에 의해 객체(클래스)가 연산을 수행하게 될 때 하나의 메시지에 대해 각 객체(클래스)가 가지고 있는 고유한 방법으로 응답할 수 있는 능력

- 65 프로토타이핑 모형(Prototyping Model, 원형 모형)은 실제 개발될 소프트웨어에 대한 견본(시제품)을 만들어 최종 결과물을 예측하는 모형으로, 이 모형의 가장 큰 장점은 요구사항을 충실히 반영할 수 있고, 요구사항의 변경이 용이하다는 것입니다. 가끔 출제되는 문제입니다. 프로토타이핑 모형의 가장 큰 장점을 기억해 두세요.

- 66 립바우의 OMT 기법에서 자료 흐름도와 가장 관련이 있는 것은 기능 모델링입니다. 립바우 객체 분석 기법의 3가지 모델링과 각 모델링의 특징을 묻는 문제가 출제되니 각 모델링의 특징을 구분하여 정리해 두세요.

#### 객체 모델링 기법(OMT, Object-Modeling Technique)

• **객체 모델링(Object Modeling)** : 정보 모델링이라고도 하며, 시스템에서 요구되는 객체를 찾아내어 속성과 연산 식별 및 객체들 간의 관계를 규정하여 객체 다이어그램으로 표시하는 것

• **동적 모델링(Dynamic Modeling)** : 상태 다이어그램(상태도)을 이용하여 시간의 흐름에 따른 객체들 사이의 제어 흐름, 상호 작용, 동작 순서 등의 동적인 행위를 표현하는 모델링

• **기능 모델링(Functional Modeling)** : 자료 흐름도(DFD)를 이용하여 다수의 프로세스들 간의 자료 흐름을 중심으로 처리 과정을 표현한 모델링

- 67 객체지향의 기본 개념 중 객체가 메시지를 받아 실행해야 할 객체의 구체적인 연산을 정의한 것은 메소드(Method)입니다. 보기에 제시된 객체지향의 기본 개념은 시험에 자주 출제되는 내용입니다. 각 개념을 구분할 수 있도록 알아두세요.

• **개체(Entity)** : 소프트웨어에 의해 인식되는 여러 종류의 자료

• **인스턴스(Instance)** : 클래스에 속한 각각의 객체

• **클래스(Class)** : 공통된 속성과 연산(행위)을 갖는 객체의 집합으로 객체의 일반적인 타입(Type)을 의미함

- 68 소프트웨어 품질 목표 중 정확하고 일관된 결과로 요구된 기능을 오류없이 수행하는 시스템의 능력을 나타내는 것은 신뢰성(Reliability)입니다. 소프트웨어 품질 목표의 종류를 묻거나 품질 목표의 각 개념을 묻는 문제가 출제됩니다. 보기로 제시된 품질 목표의 개념을 구분하여 알아두세요.

• **효율성(Efficiency)** : 요구되는 기능을 수행하기 위해 필요한 자원의 소요 정도

• **정확성(Correctness)** : 사용자의 요구 기능을 충족시키는 정도

• **무결성(Integrity)** : 허용되지 않는 사용이나 자료의 변경



을 제어하는 정도

- 69 자료 흐름도(DFD)에서 자료 저장소(Data Store)는 평행선으로 나타냅니다. 자료 흐름도의 4가지 구성 요소를 구분하는 문제는 자주 출제됩니다. 자료 흐름도의 4가지 구성 요소와 표기 기호는 반드시 알고 있어야 합니다.

**자료 흐름도의 구성 요소**

- **프로세스(Process)** : 자료를 변환시키는 시스템의 한 부분(처리 과정)을 나타내고 처리, 기능, 변환, 버블이라고도 하며 원이나 둥근 사각형으로 표시하고 그 안에 프로세스 이름을 기입함
- **자료 흐름(Flow)** : 자료의 이동(흐름)을 나타내며, 화살표 위에 자료의 이름을 기입함
- **자료 저장소(Data Store)** : 시스템에서의 자료 저장소(파일, 데이터베이스)를 나타내며, 도형 안에 자료 저장소 이름을 기입함
- **단말(Terminator)** : 시스템과 교신하는 외부 개체로, 입력 데이터가 만들어지고 출력 데이터를 받는다.(정보의 생산자와 소비자), 이중선 도형 안에 이름을 기입함

- 70 도형 목차, 총괄 도표, 상세 도표로 구성되어 전개되는 것은 HIPO입니다. CASE(Computer Aided Software Engineering)는 소프트웨어 개발 과정에서 사용되는 요구 분석, 설계, 구현, 검사 및 디버깅 과정 전체 또는 일부를 컴퓨터와 전용 소프트웨어 도구를 사용하여 자동화하는 것입니다. CASE의 개념을 묻는 문제가 종종 출제되니 다양한 CASE의 개념을 정리해 두세요.

**CASE**

- 소프트웨어 개발 과정에서 사용되는 요구 분석, 설계, 구현, 검사 및 디버깅 과정 전체 또는 일부를 컴퓨터와 전용 소프트웨어 도구를 사용하여 자동화하는 것이다.
- 소프트웨어 생명 주기의 전체 단계를 연결해 주고 자동화해 주는 통합된 도구를 제공해 주는 기술이다.
- 소프트웨어 개발 도구와 방법론이 결합된 것으로, 정형화된 구조 및 방법(메커니즘)을 소프트웨어 개발에 적용하여 생산성 향상을 구현하는 공학 기법이다.
- 소프트웨어 개발의 모든 단계에 걸쳐 일관된 방법론을 제공하는 자동화 도구(CASE Tool)들을 지원하고, 개발자들은 이 도구를 사용하여 소프트웨어 개발의 표준화를 지향하며, 자동화의 이점을 얻을 수 있게 해준다.

- 71 소프트웨어 범위 결정 요소 중 지문의 내용은 인터페이스에 대한 설명입니다. 소프트웨어 범위 결정 요소의 종류를 묻는 문제는 종종 출제되었으나 종류별 개념을 묻는 문제는 처음 출제되었습니다. 문제를 통해 인터페이스의 의미를 기억하고, 소프트웨어의 개발 영역을 결정하는 결정 요소에는 데이터, 소프트웨어에 대한 기능, 성능, 제약 조건, 인터페이스 및 신뢰도 등이 있다는 것도 함께 알아두세요.

- 72 브룩스(Brooks)의 법칙은 새로운 개발 인력이 진행중인 프로젝트에 투입될 경우 작업 적응 기간과 부작용으로 인해 일정이 더 지연된다는 법칙입니다. 그러므로 프로젝트 관리자가 문제의 프로젝트를 반대하는 이유는 일정이 더 지연되기 때문입니다. Brooks의 법칙의 개념만 알면 맞출 수 있는 문제가 출제됩니다. 문제를 통해 Brooks의 개념만 정확히 파악하십시오.

- 73 프로그래머의 수가 5명이고, 월 평균 생산이 500 라인이라면 5명이 생산하는 월 평균 양은 2500 라인입니다. 총 라인 수가 50,000일 경우 개발에 소요되는 기간은  $20(50,000 / 2500)$ 개월입니다. 가끔 출제되는 문제지만 상식적인 수준

에서 풀 수 있는 문제입니다.

- 74 소프트웨어 재사용 도입의 문제점 중 하나는 새로운 개발 방법론을 도입하기 어렵다는 것입니다. 소프트웨어 재사용의 이점과 문제점을 가려내는 문제가 출제됩니다. 각 내용을 파악해 두세요.

**소프트웨어 재사용의 이점**

- 개발 시간과 비용을 단축시킨다.
- 소프트웨어 품질을 향상시킨다.
- 소프트웨어 개발의 생산성을 향상시킨다.
- 프로젝트 실패의 위험을 감소시킨다.
- 시스템 구축 방법에 대한 지식을 공유하게 된다.
- 시스템 명세, 설계, 코드 등 문서를 공유하게 된다.

**소프트웨어 재사용 도입의 문제점**

- 어떤 것을 재사용할 것인지 선정해야 한다.
- 시스템에 공통적으로 사용되는 요소들을 발견해야 한다.
- 프로그램의 표준화가 부족하다.
- 새로운 개발 방법론을 도입하기 어렵다.
- 재사용을 위한 관리 및 지원이 부족하다.
- 기존 소프트웨어에 재사용 소프트웨어를 추가하기 어렵다.

- 75 역공학은 원시 코드로부터 설계 정보 추출 및 설계 절차 표현, 프로그램과 데이터 구조 정보를 추출하는 것입니다. 즉 결과에서 분석 및 설계 정보를 역으로 추출하는 것이죠! 자주 출제되는 내용은 아닙니다. 문제를 통해 역공학의 의미만 알고 넘어가세요.

- 76 바람직한 소프트웨어 설계는 모듈 안의 요소들이 서로 관련되어 있는 정도를 나타내는 응집도는 강하게 하고, 모듈 간에 상호 의존하는 정도를 나타내는 결합도는 약하게 해야 합니다. 가끔 출제되는 문제입니다. 바람직한 소프트웨어 설계 지침 사항을 정리해 두세요.

**바람직한 설계**

- 설계는 소프트웨어 구조를 나타내야 한다.
- 설계는 독립적인 기능적 특성을 가진 요소(모듈)로 구성되어야 한다.
- 설계는 모듈 구조, 즉 특정 기능 또는 부기능을 수행하는 논리적 요소들로 분리되는 구조를 가져야 한다.
- 소프트웨어 요소(모듈) 간의 효과적인 제어를 위해 설계에서 계층적 자료 조직이 제시되어야 한다.
- 설계는 자료와 프로시저에 대한 분명하고 분리된 표현을 포함해야 한다.
- 설계는 모듈 간과 외부 개체 간의 연결 복잡성을 줄이는 인터페이스를 가져야 한다.
- 설계는 요구사항 분석에서 얻어진 정보를 이용하여 반복적인 방법으로 이루어져야 한다.

- 77 소프트웨어 위기(Crisis)는 여러 가지 원인에 의해 소프트웨어 개발 속도가 하드웨어 개발 속도를 따라가지 못해 소프트웨어에 대한 사용자들의 요구사항을 처리할 수 없는 문제가 발생함을 의미하는데, 여러 가지 원인 중 하나는 개발 인력의 증가가 아니라 개발 인력의 부족입니다. 소프트웨어 위기의 의미를 생각하면서 소프트웨어 위기 현상을 한번 읽어보고 넘어가세요.

**소프트웨어 위기의 현상**

- 개발 인력의 부족과 그로 인한 인건비 상승
- 성능 및 신뢰성의 부족
- 개발 기간의 지연 및 개발 비용의 증가
- 유지보수가 어렵고, 이에 따른 비용 증가
- 소프트웨어의 생산성 저하

## · 소프트웨어의 품질 저하

- 78 자료 사전은 자료 흐름도에 있는 자료를 더 자세히 정의하고 기록한 것으로, 자료 사전에서 반복을 의미하는 기호는 { }입니다. 자료 사전의 표기 기호를 묻는 문제는 종종 출제됩니다. 자료 사전에서 사용하는 표기 기호는 기호의 기능을 구분할 수 있을 정도로만 알아두세요.

## 자료 사전에서 사용되는 표기 기호

=	자료의 정의 : ~로 구성되어 있다(is composed of)
+	자료의 연결 : 그리고(and)
( )	자료의 생략 : 생략 가능한 자료(Optional)
[   ]	자료의 선택 : 또는(or)
{ }	자료의 반복

- 79 화이트 박스 테스트 기법에는 기초 경로 검사(Basic Path Test), 제어 구조 검사(조건 검사(Condition Test), 루프 검사(Loop Test), 데이터 흐름 검사(Data Flow Test) 등이 있습니다. 동치 분할 검사(Equivalence Partitioning Test), 비교 검사(Comparison Test), 원인-효과 그래프 검사(Cause-effect Graphing Test)는 블랙 박스 테스트 기법입니다. 문제를 통해 화이트 박스 테스트 기법과 블랙 박스 테스트 기법의 종류를 반드시 구분하여 정리해 두세요.
- 80 소프트웨어 공학(SE, Software Engineering)은 소프트웨어의 위기를 극복하기 위한 방안으로, 소프트웨어의 제작부터 운영까지 생산성을 높이기 위해 기술적, 인간적인 요소에 대한 방법론을 제공하는 것입니다. 자주 출제되는 문제는 아닙니다. 문제를 통해 소프트웨어 공학의 의미만 기억하고 넘어가세요.
- 81 전송 제어 절차를 순서대로 나열하면 '회선 접속 → 데이터 링크 확립 → 정보 전송 → 데이터 링크 해제 → 회선 절단'순입니다. 단순히 전송 제어 절차의 순서를 묻는 문제가 거의 매회 출제됩니다. 전송 제어 절차 순서를 정확히 숙지하세요.
- 82 스트림(Stream) 전송은 TCP의 기능입니다. TCP와 IP의 기능을 구분하는 문제가 종종 출제됩니다. 이 문제에서는 나머지 보기로 제시된 IP의 대표적인 기능들만 확실히 정리해 두세요.
- 83 문제의 지문에 제시된 내용은 다중화에 대한 설명입니다. 자주 출제되지 않는 내용입니다. 이 문제에서는 다중화의 개념만 기억하세요.
- 84 보기 중 오류 검출을 위해 사용되는 기법이 아닌 것은 Run Length Check입니다. Run Length 부호화 방식은 데이터 압축 기법입니다. 자주 출제되지 않는 문제입니다. 보기로 제시된 오류 검출 기법의 종류만 간단히 정리하고 넘어가세요.
- 85 데이터 전송 속도 중 반송파 주파수 속도라는 것은 없습니다. 자주 출제되는 내용은 아닙니다. 보기로 제시된 속도들의 의미만 간단히 정리하세요.
- 변조 속도 : 1초 동안 몇 번의 신호 변화가 있었는가를 나타내는 것으로, 단위는 Baud를 사용함
  - 신호 속도 : 1초 동안 전송 가능한 비트의 수를 나타내는 것으로, 단위는 bps(bit/sec)를 사용함
  - 베리어 속도 : 데이터 신호에 동기 문자, 상태 신호 등을 합한 속도로, 단위는 bps(bit/sec)를 사용함
- 86 문제의 지문으로 제시된 내용은 ARQ 기법 중 Stop-and-

Wait ARQ 기법의 특징입니다. 자동 반복 요청(ARQ)은 의미와 종류, 각 종류의 수행 방식을 묻는 문제 등 다양하게 출제됩니다. 확실히 정리해 두세요.

## 자동 반복 요청(ARQ, Automatic Repeat reQuest)

오류 발생 시 수신측은 오류 발생을 송신측에 통보하고, 송신측은 오류 발생 블록을 재전송하는 모든 절차를 의미하는 것으로, 다음과 같은 방법이 있다.

## 정지-대기(Stop-and-Wait) ARQ

- 송신측에서 한 개의 블록을 전송한 후 수신측으로부터 응답을 기다리는 방식이다.
- 수신측의 응답이 긍정 응답(ACK)이면 다음 블록을 전송하고, 부정 응답(NAK)이면 앞서 송신했던 블록을 재전송한다.
- 블록을 전송할 때마다 수신측의 응답을 기다려야 하므로 전송 효율이 가장 낮다.
- 오류가 발생한 경우 앞서 송신했던 블록만 재전송하면 되므로 구현 방법이 가장 단순하다.

## 연속(Continuous) ARQ

- 정지-대기(Stop-and-Wait) ARQ가 갖는 오버헤드를 줄이기 위해 연속적으로 데이터 블록을 보내는 방식으로, 수신측에서는 부정 응답(NAK)만을 송신한다.

## Go-Back-N ARQ

- 여러 블록을 연속적으로 전송하고, 수신측에서 부정 응답(NAK)을 보내오면 송신측이 오류가 발생한 블록 이후의 모든 블록을 재전송한다.
- 전송 오류가 발생하지 않으면 쉬지 않고 연속적으로 송신이 가능하다.
- 오류가 발생한 부분부터 모두 재전송하므로 중복 전송의 단점이 있다.

## 선택적 재전송(Selective Repeat) ARQ

- 여러 블록을 연속적으로 전송하고, 수신측에서 부정 응답(NAK)을 보내오면 송신측이 오류가 발생한 블록만을 재전송한다.
- 수신측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리 회로와 큰 용량의 버퍼가 필요하다.

## 적응적(Adaptive) ARQ

- 전송 효율을 최대로 하기 위해서 데이터 블록의 길이를 채널의 상태에 따라 그때그때 동적으로 변경하는 방식이다.
- 전송 효율이 제일 좋다.
- 제어 회로가 매우 복잡하고 비용이 많이 들어 현재는 거의 사용되지 않고 있다.

- 87 전이중 통신 방식은 동시에 양방향으로 전송이 가능한 방식으로 통신 회선의 효율이 가장 좋으며 전화 등에 사용됩니다. 통신 방식의 종류별 특징을 서로 구분할 수 있어야 풀 수 있는 문제가 종종 출제됩니다. 나머지 보기가 틀린 이유를 파악하면서 정리하세요.

나. 반이중 통신 방식은 무전기나 모뎀을 이용한 데이터 통신에서 사용됩니다.

다. 단방향 통신 방식은 1선식, 반이중 통신 방식은 2선식 선로를 사용합니다.

라. 반이중 통신 방식에 대한 설명입니다.

- 88 문제에 제시된 내용은 회선 교환 방식의 가장 대표적인 특징입니다. 회선 교환 방식의 특징을 묻는 문제가 종종 출제되고 있습니다. '일단 접속이 이루어지면 접속을 해제할 때까지 전용선처럼 사용할 수 있다'는 것을 중심으로 특징을 정리하세요.

## 회선 교환 방식

- 데이터 전송 전에 먼저 물리적 통신 회선을 통한 연결이



필요하다.

- 일단 접속이 되고 나면 그 통신 회선은 전용 회선에 의한 통신처럼 데이터가 전달된다(고정 대역 전송).
- 접속에는 긴 시간이 소요되나, 일단 접속되면 전송 지연이 거의 없어 실시간 전송이 가능하다.
- 회선이 접속되더라도 수신측이 준비되어 있지 않으면 데이터 전송이 불가능하다.
- 데이터 전송에 필요한 전체 시간이 축적 교환 방식에 비해 길다.
- 접속된 두 지점이 회선을 독점하기 때문에 접속된 이외의 다른 단말기는 전달 지연을 가지게 된다.
- 데이터가 전송되지 않는 동안에도 접속이 유지되기 때문에 데이터 전송이 연속적이지 않은 경우 통신 회선이 낭비된다.
- 일정한 데이터 전송률을 제공하므로 동일한 전송 속도가 유지된다.
- 전송된 데이터의 오류 제어나 흐름 제어는 사용자에게 의해서 수행된다.

89 LAN의 구성 형태 중 Mesh(망)형은 없습니다. Mesh(망)형은 주로 공중망을 구성할 때 사용합니다. LAN을 망의 구성 형태에 따라 분류하면 성(Star)형, 버스(Bus)형, 링(Rig)형, 계층(Tree)형으로 분류할 수 있습니다. 망의 구성 형태에 따른 분류를 기억하고 보기로 제시된 구성 형태와 매체 접근 제어 방식의 연결 형태도 간단히 알아두세요.

90 HDLC의 데이터 전송 모드에는 표준(정규) 응답 모드(NRM), 비동기 응답 모드(ARM), 비동기 균형(평형) 모드(ABM)가 있습니다. 데이터 전송 모드의 종류를 묻는 문제가 종종 출제됩니다. 3가지 종류는 꼭 기억하고 각각의 특징은 간단히 읽어보세요.

#### 데이터 전송 모드

- **표준(정규) 응답 모드(NRM, Normal Response Mode)**
  - 반이중 통신을 하는 포인트 투 포인트(Point-to-Point) 또는 멀티 포인트(Multi-Point) 불균형 링크 구성에 사용
  - 중국은 주국의 허가(Poll)가 있을 때에만 송신
- **비동기 응답 모드(ARM, Asynchronous Response Mode)**
  - 전이중 통신을 하는 포인트 투 포인트(Point-to-Point) 불균형 링크 구성에 사용
  - 중국은 주국의 허가(Poll) 없이도 송신이 가능하지만, 링크 설정이나 오류 복구 등의 제어 기능은 주국만 함
- **비동기 균형(평형) 모드(ABM, Asynchronous Balanced Mode)**
  - 포인트 투 포인트(Point-to-Point) 균형 링크에서 사용
  - 혼합국끼리 허가 없이 언제나 전송할 수 있도록 설정

91 토큰 패싱이란 노드 사이의 충돌을 방지하기 위해 토큰(전송 권한)을 사용하여 한 노드만 전송을 수행토록 하는 것으로 전송 후에는 다른 노드가 전송할 수 있도록 토큰을 반납하는 방식입니다. 토큰 패싱 방식은 적용하는 망의 형태에 따라 토큰 버스와 토큰 링으로 구분할 수 있습니다. 처음 출제된 문제입니다. 이 문제에서는 토큰 패싱의 개념만 명확히 정리해 두세요.

92 에러 제어를 위해 Stop-and-Wait 방식을 지원하는 것은 BSC 프로토콜입니다. HDLC 프로토콜은 G-Back-N과 선택적 재전송 ARQ 방식을 사용합니다. HDLC는 주로 프레임의 구조와 관련된 문제가 출제됩니다. 보기로 제시된 특징과 다음에 설명한 HDLC 프레임의 구조에 대해서 정리해 두세요.

#### HDLC 프레임의 구조

- **플래그(Flag)** : 프레임의 시작과 끝을 나타내는 고유한 비트 패턴(01111110)
- **주소부(Address Field)** : 송·수신국을 식별하기 위해 사용. 불특정 다수에게 전송하는 방송용(Broadcast)은 '11111111', 시스템에 의해 임의로 수신국이 지정되는 시험용(No Station)은 '00000000'을 사용
- **제어부(Control Field)** : 프레임의 종류를 식별하기 위해 사용. 제어부의 첫 번째, 두 번째 비트를 사용하여 구별함
- **정보부(Information Field)** : 실제 정보 메시지가 들어 있는 부분으로, 송·수신측 간의 협의에 따라 길이와 구성이 정해짐
  - **정보 프레임(Information Frame)** : 제어부가 '0'으로 시작하는 프레임으로, 사용자 데이터를 전달하는 역할을 함
  - **감독 프레임(Supervisor Frame)** : 제어부가 '10'으로 시작하는 프레임으로, 오류 제어와 흐름 제어를 위해 사용됨
  - **비번호 프레임(Unnumbered Frame)** : 제어부가 '11'로 시작하는 프레임으로, 링크의 동작 모드 설정과 관련 있음
- **FCS(Frame Check Sequence Field, 프레임 검사 순서 필드)** : 프레임 내용에 대한 오류 검출을 위해 사용되는 부분으로, 일반적으로 CRC 코드가 사용됨

93 동기식 시분할 다중화 방식은 모든 단말기에 균등한(고정된) 시간폭을 제공합니다. 이는 전송할 데이터가 없는 경우에도 시간폭이 제공되므로 통계적(비동기식) 시분할 다중화 방식에 비해 전송 용량의 낭비가 많이 발생합니다. 동기식 시분할 다중화기와 비동기식 시분할 다중화기의 특징을 서로 구분하는 문제가 출제되는데 주로 비동기식 시분할 다중화기의 특징을 중심으로 출제되니 확실히 정리해 두세요.

#### 비동기식 시분할 다중화기(ATDM, Asynchronous TDM)

- 마이크로프로세서를 이용하여 접속된 단말기 중 전송할 데이터가 있는 단말기에만 시간폭(Time Slot)을 제공한다.
- 비동기식 시분할 다중화기는 낭비되는 시간폭을 줄일 수 있고, 남은 시간폭을 다른 용도로 사용할 수 있으므로, 전송 효율이 높다.
- 동일한 조건일 경우 동기식 시분할 다중화기보다 많은 수의 단말기들이 전송 매체에 접속할 수 있으며, 더 높은 전송 효율을 가진다.
- 다중화기의 내부 속도와 단말기의 속도 차이를 보완하기 위한 버퍼가 필요하다.
- 데이터 전송량이 많아질 경우 전송 지연이 생길 수 있다.
- 동기식 시분할 다중화기에 비해 접속하는 데 소요되는 시간이 길다.
- 주소 제어, 흐름 제어, 오류 제어 등의 기능을 하므로 복잡한 제어 회로와 임시 기억 장치가 필요하며 가격이 비싸다.
- 지능 다중화기, 확률적 다중화기, 통계적 시분할 다중화기라고도 한다.

94 맨체스터 부호화 방식은 동기 전송 중 비트 동기 방식에서 사용합니다. 비동기 전송 방식의 세부적인 내용까지 알아야 풀 수 있는 어려운 문제입니다. 비동기 전송 방식의 특징은 나머지 보기로 제시된 내용 정도만 알아두고 자주 출제되는 동기식 전송 방식의 특징도 함께 정리해 두세요.

#### 동기식 전송

- 미리 정해진 수만큼의 문자열을 한 블록(프레임)으로 만들어 일시에 전송하는 방식이다.
- 송·수신 양쪽의 동기를 유지하기 위해서 타이밍 신호(클

력)를 계속적으로 공급하거나 동기 문자를 전송한다.

- 동기화된 방식으로 비트들이 송·수신되기 때문에 비동기식 전송과는 다르게 시작 비트와 종료 비트가 필요 없다.
- 블록과 블록 사이에 휴지 시간(Idle Time)이 없다.
- 프레임 단위로 전송하므로 전송 속도가 빠르다.
- 시작/종료 비트로 인한 오버헤드가 없고, 휴지 시간이 없으므로, 전송 효율이 좋다.
- 주로 원거리 전송에 사용한다.
- 단말기는 반드시 버퍼 기억 장치를 내장하여야 한다.
- 동기식 전송 방식에는 비트 동기 방식과 블록 동기 방식이 있으며, 블록 동기 방식은 다음과 같이 문자 위주 동기 방식과 비트 위주 동기 방식으로 나뉜다.
  - **문자(Character)위주 동기 방식** : SYN 등의 동기 문자(전송 제어 문자)에 의해 동기를 맞추는 방식으로, BSC 프로토콜에서 사용된다.
  - **비트(Bit)위주 동기 방식** : 데이터 블록의 처음과 끝에 8비트의 플래그 비트(01111110)를 표시하여 동기를 맞추는 방식으로, HDLC와 SDLC 프로토콜에서 사용된다.

- 95 PCM(펄스 코드 변조)은 아날로그 데이터를 디지털 신호로 변환하는 방식입니다. 산업기사 시험에 주로 출제되는 문제이나 신호 변환 방식의 기본적인 내용이니 보기로 제시된 신호 변환 방식의 개념을 간단히 읽어보세요.
- **ASK(진폭 편이 변조)** : 2진수 0과 1을 서로 다른 진폭의 신호로 변조하는 방식
  - **FSK(주파수 편이 변조)** : 2진수 0과 1을 서로 다른 주파수로 변조하는 방식
  - **PSK(위상 편이 변조)** : 2진수 0과 1을 서로 다른 위상을 갖는 신호로 변조하는 방식

- 96 고정적인 전송 대역폭을 제공하는 것은 회선 교환 방식입니다. 패킷 교환 방식의 특징을 묻는 문제가 자주 출제됩니다. 확실히 정리해 두세요.

#### 패킷 교환 방식(Packet Switching)

- 메시지를 일정한 길이의 패킷으로 잘라서 전송하는 방식이다.
- 패킷은 장애 발생 시의 재전송을 위해 패킷 교환기에 일시 저장되었다가 곧 전송되며 전송이 끝난 후 폐기된다.
- 패킷 교환망은 OSI 참조 모델의 네트워크 계층에 해당한다.
- 패킷형 터미널을 위한 DTE와 DCE 사이의 접속 규정은 X.25이다.
- 패킷망 상호 간의 접속을 위한 프로토콜은 X.75이다.
- 하나의 통신 회선을 여러 사용자가 공유할 수 있으므로 회선 이용률이 높다.
- 수신측에서 분할된 패킷을 재조립해야 한다.
- 응답 시간이 빠르므로 대화형 응용이 가능하다.
- 통신량의 제어를 통한 망의 안전성을 높일 수 있다.
- 전송 시 교환기, 통신 회선 등에 장애가 발생하여도 다른 정상적인 경로를 선택하여 우회할 수 있다.
- 음성 전송보다 데이터 전송에 더 적합하다.

- 97 HDLC 프레임의 종류가 아닌 것은 Handle Frame입니다. HDLC는 주로 프레임의 구조와 관련된 문제가 출제된다고 했죠? 이 문제를 틀렸다면 92번 해설을 통해 HDLC 프레임을 다시 한번 정리하세요.

- 98 문제에 제시된 내용은 Physical Layer(물리 계층)에 대한 설명입니다. OSI 7계층과 관련된 문제는 어떤 계층을 설명한 것인지를 구분하는 문제가 출제됩니다. OSI 7계층별 주요 기능을 암기하세요.

#### 물리 계층(Physical Layer)

- 물리 계층은 전송에 필요한 두 장치 간의 실제 접속과 절단 등 기계적, 전기적, 기능적, 절차적 특성에 대한 규칙을 정의한다.
- 물리적 전송 매체와 전송 신호 방식을 정의하며, RS-232C, X.21 등의 표준이 있다.

#### 데이터 링크 계층(Data Link Layer)

- 데이터 링크 계층은 두 개의 인접한 개방 시스템들 간에 신뢰성 있고 효율적인 정보 전송을 할 수 있도록 한다.
- 송신측과 수신측의 속도 차이를 해결하기 위한 흐름 제어 기능을 한다.
- 프레임의 시작과 끝을 구분하기 위한 프레임의 동기화 기능을 한다.
- 오류의 검출과 회복을 위한 오류 제어 기능을 한다.
- 프레임의 순서적 전송을 위한 순서 제어 기능을 한다.
- HDLC, LAPB, LLC, LAPD 등의 표준이 있다.

#### 네트워크 계층(Network Layer, 망 계층)

- 네트워크 계층은 개방 시스템들 간의 네트워크 연결을 관리하는 기능과 데이터의 교환 및 중계 기능을 한다.
- 네트워크 연결을 설정, 유지, 해제하는 기능을 한다.
- 경로 설정(Routing), 데이터 교환 및 중계, 트래픽 제어, 패킷 정보 전송을 수행한다.
- 관련 표준으로는 X.25, IP 등이 있다.

#### 전송 계층(Transport Layer)

- 전송 계층은 논리적 안정과 균일한 데이터 전송 서비스를 제공함으로써 종단 시스템(End-to-End) 간에 투명한 데이터 전송을 가능하게 한다.
- OSI 7 계층 중 하위 3 계층과 상위 3 계층의 인터페이스(Interface)를 담당한다.
- 종단 시스템(End-to-End) 간의 전송 연결 설정, 데이터 전송, 연결 해제 기능을 한다.
- 주소 설정, 다중화, 오류 제어, 흐름 제어를 수행한다.
- TCP, UDP 등의 표준이 있다.

#### 세션 계층(Session Layer)

- 세션 계층은 송·수신측 간의 관련성을 유지하고 대화 제어를 담당하는 계층이다.
- 대화(회화) 구성 및 동기 제어, 데이터 교환 관리 기능을 한다.
- 송·수신측의 대화(회화) 동기를 위해 전송하는 정보의 일정한 부분에 체크점을 두어 정보의 수신 상태를 체크하며, 이때의 체크점을 동기점이라고 한다.
- 동기점은 오류가 있는 데이터의 회복을 위해 사용하는 것으로, 종류에는 소동기점과 대동기점이 있다.

#### 표현 계층(Presentation Layer)

- 표현 계층은 응용 계층으로부터 받은 데이터를 세션 계층에 보내기 전에 통신에 적당한 형태로 변환하고, 세션 계층에서 받은 데이터는 응용 계층에 맞게 변환하는 기능을 한다.
- 서로 다른 데이터 표현 형태를 갖는 시스템 간의 상호 접속을 위해 필요한 계층이다.
- 코드 변환, 데이터 암호화, 데이터 압축, 구문 검색, 정보 형식(포맷) 변환, 문맥 관리 기능을 한다.

#### 응용 계층(Application Layer)

- 응용 계층은 사용자(응용 프로그램)가 OSI 환경에 접근할 수 있도록 서비스를 제공한다.
- 응용 프로세스 간의 정보 교환, 전자 사서함, 파일 전송 등의 서비스를 제공한다.

- 99 TCP/IP 프로토콜 중 인터넷 계층에 대응하는 OSI 계층은 네트워크(Network) 계층입니다. TCP/IP의 계층이 OSI의

어느 계층에 대응되는지를 묻는 문제가 종종 출제됩니다.  
TCP/IP의 구조에 대해 정리해 두세요.

#### TCP/IP 구조

OSI	TCP/IP	기능
응용 계층 표현 계층 세션 계층	응용 계층	<ul style="list-style-type: none"> <li>• 응용 프로그램 간의 데이터 송수신 제공</li> <li>• TELNET, FTP, SMTP, SNMP 등</li> </ul>
전송 계층	전송 계층	<ul style="list-style-type: none"> <li>• 호스트들 간의 신뢰성 있는 통신 제공</li> <li>• TCP, UDP</li> </ul>
네트워크 계층	인터넷 계층	<ul style="list-style-type: none"> <li>• 데이터 전송을 위한 주소 지정, 경로 설정을 제공</li> <li>• IP, ICMP, IGMP, ARP, RARP</li> </ul>
데이터 링크 계층 물리 계층	링크 계층	<ul style="list-style-type: none"> <li>• 실제 데이터(프레임)를 송수신하는 역할</li> <li>• Ethernet, IEEE 802, HDLC, X.25, RS-232C</li> </ul>

100 문제에 제시된 내용은 가드 밴드에 대한 설명입니다. 주파수 분할 다중화기는 특징이나 가드밴드의 개념을 묻는 문제가 출제됩니다. 가드 밴드의 개념을 기억하고 특징을 정리하세요.

#### 주파수 분할 다중화기(FDM, Frequency Division Multiplexer)

- 통신 회선의 주파수를 여러 개로 분할하여 여러 대의 단말기가 동시에 사용할 수 있도록 한 것이다.
- 전송 신호에 필요한 대역폭보다 통신 회선의 유효 대역폭이 큰 경우에 사용한다.
- 다중화기 자체에 변·복조 기능이 내장되어 있어 모뎀을 설치할 필요가 없다.
- 시분할 다중화기에 비해 구조가 간단하고 가격이 저렴하다.
- 저속(1,200bps 이하)의 비동기식 전송, 멀티 포인트(Multi-Point) 방식에 적합하다.
- 아날로그 신호 전송에 적합하다.
- 대역폭을 나누어 사용하는 각 채널들 간의 상호 간섭을 방지하기 위한 보호 대역(Guard Band)이 필요하다.
- 보호 대역(Guard Band) 사용으로 인한 대역폭의 낭비가 초래된다.