



시험에 나오는 것만 공부한다!

시나공시리즈

기출문제 & 정답 및 해설 2007년 제4회 정보처리기사 A형



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. 관계형 데이터 모델의 참조 무결성 제약에 관한 설명 중 옳지 않은 것은?

- 가. 외래 키의 속성들은 참조하려는 테이블의 기본 키와 도메인이 동일해야 한다.
- 나. 외래 키의 속성명과 참조하려는 테이블의 기본 키의 속성명은 동일해야 한다.
- 다. 외래 키의 속성 개수와 참조하려는 테이블의 기본 키의 속성 개수는 같아야 한다.
- 라. 외래 키 값은 참조하려는 테이블의 기본 키 값으로 존재해야 한다.

2. 2단계 로킹(Two Phase Locking)에 대한 설명으로 옳지 않은 것은?

- 가. 직렬성을 보장한다.
- 나. 확장 단계와 축소 단계의 두 단계(Phase)가 있다.
- 다. 교착 상태를 예방할 수 있다.
- 라. 각 트랜잭션의 로크 요청과 해제 요청을 2단계로 실시한다.

3. 스키마(Schema)에 대한 설명으로 거리가 먼 것은?

- 가. 데이터베이스를 운용하는 소프트웨어이다.
- 나. 데이터 사전(Data Dictionary)에 저장된다.
- 다. 다른 이름으로 메타 데이터(Meta-Data)라고도 한다.
- 라. 데이터베이스의 구조(개체, 속성, 관계)에 대한 정의이다.

4. 릴레이션의 특성으로 적합하지 않은 것은?

- 가. 중복된 튜플이 존재하지 않는다.
- 나. 튜플 간의 순서는 없다.
- 다. 속성 간의 순서는 있다.
- 라. 모든 속성값은 원자값을 갖는다.

5. 시스템 카탈로그에 대한 설명으로 옳지 않은 것은?

- 가. 사용자가 시스템 카탈로그를 직접 갱신할 수 있다.
- 나. 일반 질의어를 이용해 그 내용을 검색할 수 있다.
- 다. DBMS가 스스로 생성하고, 유지하는 데이터베이스 내의 특별한 테이블의 집합체이다.
- 라. 데이터베이스 스키마에 대한 정보를 제공한다.

6. 트랜잭션이 가져야 되는 특성과 거리가 먼 것은?

- 가. 원자성(Atomicity)
- 나. 일관성(Consistency)
- 다. 독립성(Independency)
- 라. 영속성(Durability)

7. Which of following is not a component of Entity-Relationship diagram?

- 가. Rectangles, which represent entity sets
- 나. Ellipses, which represent database operations
- 다. Diamond, which represent relationships among entity sets
- 라. Lines, which link attributes to entity sets and entity sets to relationships

8. 운영체제의 작업 스케줄링 등에 응용될 수 있는 가장 적합한 자료 구조는?

- 가. 스택(Stack)
- 나. 큐(Queue)
- 다. 트리(Tree)
- 라. 연결 리스트(Linked List)

9. 연결 리스트(Linked List)에 대한 설명으로 거리가 먼 것은?

- 가. 노드의 삽입이나 삭제가 쉽다.
- 나. 노드들이 포인터로 연결되어 검색이 빠르다.
- 다. 연결을 해주는 포인터(Pointer)를 위한 추가 공간이 필요하다.
- 라. 연결 리스트 중에서 중간 노드 연결이 끊어지면 그 다음 노드를 찾기 힘들다.

10. SQL에서 DELETE 명령에 대한 설명으로 옳지 않은 것은?

- 가. 테이블의 행을 삭제할 때 사용한다.
- 나. 특정 테이블에 대하여 WHERE 조건절이 없는 DELETE 명령을 수행하면 DROP TABLE 명령을 수행했을 때와 같은 효과를 얻을 수 있다.
- 다. SQL을 사용 용도에 따라 분류할 경우 DML에 해당한다.
- 라. 기본 사용 형식은 "DELETE FROM 테이블 [WHERE 조건];"이다.

11. 분산 데이터베이스 설계 시 고려 사항으로 옳지 않은 것은?

- 가. 작업 부하(Work Load)의 노드별 분산 정책
- 나. 지역의 자치성 보장 정책
- 다. 데이터의 일관성 정책
- 라. 분산 노드 간 데이터의 중복성 보장과 가용성 감소 정책

12. 데이터베이스 설계 순서로 옳은 것은?

- 가. 요구 조건 분석 → 개념적 설계 → 논리적 설계 → 물리적 설계 → 구현
- 나. 요구 조건 분석 → 논리적 설계 → 개념적 설계 → 물리적 설계 → 구현
- 다. 요구 조건 분석 → 논리적 설계 → 물리적 설계 → 개념적 설계 → 구현
- 라. 요구 조건 분석 → 개념적 설계 → 물리적 설계 → 논리적 설계 → 구현

13. 데이터베이스의 물리적 설계 옵션 선택 시 고려 사항으로 거리가 먼 것은?

- 가. 스키마의 평가
나. 응답 시간
다. 저장 공간의 효율화
라. 트랜잭션 처리도(Throughput)

14. 다음의 빈칸에 적합한 단어는 무엇인가?

() is an ordered list in which all insertions and deletions are made at one end, called the top.

- 가. Queue 나. Dequeue
다. Stack 라. Linked list

15. 양방향에서 입·출력이 가능한 선형 자료구조로서 2개의 포인터를 이용하여 양쪽 끝 모두에서 삽입·삭제가 가능한 것은?

- 가. 데크(Deque) 나. 스택(Stack)
다. 큐(Queue) 라. 트리(Tree)

16. 데이터베이스의 특성으로 옳지 않은 것은?

- 가. 실시간 접근성 나. 지속적인 변화
다. 동시 공유 라. 주소에 의한 참조

17. 다음의 Infix로 표현된 수식을 Postfix 표기로 옳게 변환한 것은?

$$A = (B - C) * D + E$$

- 가. $A = A * - B C + D E$
나. $A = A + * - B C D E$
다. $A B C - D * E + =$
라. $A B C * D - E + =$

18. 데이터 모델의 구성 요소로 거리가 먼 것은?

- 가. Mapping 나. Structure
다. Operation 라. Constraint

19. SQL의 명령은 사용 용도에 따라 DDL, DML, DCL로 구분할 수 있다. 다음 명령 중 그 성격이 나머지 셋과 다른 것은?

- 가. SELECT 나. UPDATE
다. INSERT 라. GRANT

20. 데이터베이스의 3층 스키마 중 모든 응용 시스템과 사용자들이 필요로 하는 데이터를 통합한 조직 전체의 데이터베이스 구조를 논리적으로 정의하는 스키마는?

- 가. 개념 스키마 나. 외부 스키마
다. 내부 스키마 라. 응용 스키마

제2과목 전자계산기 구조

21. 2진수 $(1011)_2$ 를 Gray Code로 변환하면?

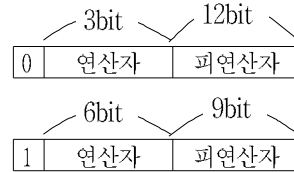
- 가. 1001 나. 1100
다. 1111 라. 1110

22. 다음 연산의 결과는(단, 수의 표현은 2's Complement임)?

$$101011 - 100110$$

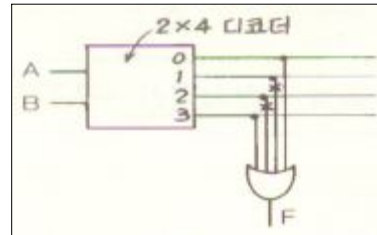
- 가. 000110 나. 000101
다. 100110 라. 100101

23. 16-Bit 컴퓨터 시스템에서 다음과 같은 2가지의 명령어 형식을 사용할 때 최대 연산자의 수는?



- 가. 64 나. 72
다. 86 라. 144

24. 다음 그림에서 F를 A, B의 불식으로 나타내면?(단, 그림에서 X는 선의 절단을 표시함)



- 가. $F = A + B$ 나. $F = AB + AB$
다. $F = AB$ 라. $F = AB + AB$

25. 마이크로 명령 형식으로 적합하지 않은 것은?

- 가. 수평 마이크로 명령 나. 제어 마이크로 명령
다. 수직 마이크로 명령 라. 나노 명령

26. 매크로(Macro)의 인수(因數) 사용에 해당되지 않는 것은?

- 가. 인수의 형(Type) 나. 인수의 위치
다. 인수를 지정 라. 인수의 수를 변동

27. 프로그램 실행 중에 트랩(Trap)이 발생하는 조건이 아닌 것은?

- 가. Overflow 또는 Underflow 시
나. 0(Zero)에 의한 나눗셈
다. 불법적인 명령
라. 패리티 가격오류

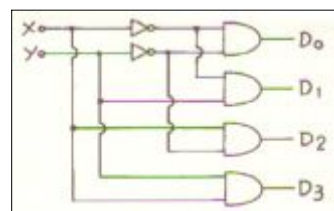
28. 중앙연산처리장치의 하드웨어적인 요소가 아닌 것은?

- 가. IR(Instruction Register)
나. MAR(Memory Address Register)
다. MODEM(MODulator DEModulator)
라. PC(Program Counter)

29. 연관기억장치(Associative Memory)에 대한 설명과 가장 관계가 없는 것은?

- 가. 저장 공간의 확대가 목적이다.
나. 신속한 검색이 가능하다.
다. 주소를 필요로 하지 않는다.
라. 하드웨어의 비용이 크다.

30. 다음 회로는 무엇인가?



- 가. Decoder 나. Multiplexer
다. Encoder 라. Shifter

31. 16진수 A4D를 8진수로 바꾸면?

- 가. 5115 나. 5116
다. 5117 라. 5118

32. 누산기(Accumulator)에 대한 설명 중 옳은 것은?

- 가. 연산장치에 있는 레지스터의 하나로서 연산 결과를 기억하는 장치이다.
나. 기억장치 주변에 있는 회로인데 가감승제 계산 및 논리 연산을 행하는 장치이다.
다. 일정한 입력 숫자들을 더하여 그 누계를 항상 보관하는 장치이다.
라. 정밀 계산을 위해 특별히 만들어 두어 유효 숫자의 개수를 늘리기 위한 것이다.

33. 프로그램 수행 도중 서로 다른 번지의 주소를 동시에 지정하는 방식은?

- 가. 파이프라인 방식 나. 인터리빙 방식
다. 인코딩 방식 라. 메모리 캐시 방식

34. 컴퓨터가 인터럽트 루틴 수행 후에 처리하는 것은?

- 가. 전원을 다시 동작한다.
나. 모니터 화면에 인터럽트 종류를 디스플레이 한다.
다. 메모리의 내용을 지워서 다른 프로그램이 적재될 수 있도록 한다.
라. 인터럽트 처리 시 보존시켰던 PC 및 제어상태 데이터를 PC와 제어상태 레지스터에 복구한다.

35. 가상기억장치에 대한 설명 중 틀린 것은?

- 가. 주소공간이란 가상공간의 집합을 말한다.
나. 실제 컴퓨터의 기억장치내 주소를 물리주소라고 한다.
다. 가상주소를 물리주소로 변환하는 방법의 하나로 CAM을 사용한다.
라. 빈번히 참조되는 프로그램이나 데이터를 별도의 메모리에 저장하여 처리한다.

36. Stack 구조의 컴퓨터에서 사용하는 연산 명령어의 주소지정방식은?

- 가. 0-Address 나. 1-Address
다. 2-Address 라. 3-Address

37. 명령어 수행시간이 10ns이고, 명령어 패치 시간이 5ns, 명령어 준비시간이 3ns이라면 인스트럭션의 성능은 얼마인가?

- 가. 0.1 나. 0.3 다. 0.5 라. 1.25

38. 하드웨어 신호에 의하여 특정 번지의 서브루틴을 수행하는 것은?

- 가. Handshaking Mode
나. Vectored Interrupt
다. DMA
라. Subroutine Call

39. 우선순위 인터럽트 운영 방식이 아닌 것은?

- 가. LCFS(Last Come First Service)
나. FCFS(First Come First Service)
다. Masking Scheme
라. Fixed Service

40. 소프트웨어에 의한 우선순위 체제의 특성을 설명한 것으로 옳지 않은 것은?

- 가. 경제적이다.
나. 융통성이 있다.
다. 반응속도가 느리다.
라. 정보량이 매우 적은 시스템에 적합하다.

제3과목 운영체제

41. UNIX의 파일 시스템 구조와 거리가 먼 것은?

- 가. 부트 블록 나. 사용자 블록
다. I-node 블록 라. 슈퍼 블록

42. 두 개의 프로세스 간 선행 순서를 $P_i < P_j$ 로 표현할 경우 P_j 가 먼저 실행된다고 가정한다면, $P_2 < P_1$, $P_4 < P_2$, $P_4 < P_3$ 의 선행관계가 있는 경우에 병행으로 실행될 수 있는 프로세스로 짝지어진 것은?

- 가. P_1, P_3 나. P_1, P_4
다. P_2, P_4 라. P_3, P_4

43. 주기억장치 배치 전략 기법으로 최초 적합(First Fit) 방법을 사용하고 할 때, 다음과 같은 기억장소 리스트에서 10K 크기의 작업은 어느 기억공간에 할당되는가?(단, 탐색은 위에서 아래로 한다.)

운영체제	
사 용 중	
5K	← A
사 용 중	
15K	← B
사 용 중	
25K	← C

- 가. A 나. B
다. C 라. 할당할 수 없다.

44. 3개의 페이지 프레임(Frame)을 가진 기억장치에서 페이지 요청을 다음과 같은 페이지 번호 순으로 요청했을 때 교체 알고리즘으로 FIFO 방법을 사용한다면 몇 번의 페이지 부재(Fault)가 발생하는가?(단, 현재 기억장치는 모두 비어 있다고 가정한다.)

요청된 페이지 번호의 순서 : 2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2

- 가. 7번 나. 8번 다. 9번 라. 10번

45. 일반적으로 사용되는 자원 보호 기법의 종류에 해당하지 않는 것은?

- 가. 접근 제어 행렬(Access Control Matrix)
나. 접근 제어 리스트(Access Control List)
다. 권한 행렬(Capability Matrix)
라. 권한 리스트(Capability List)

46. 운영체제를 기능상 분류했을 때, 제어 프로그램 중 다음 설명에 해당하는 것은?

주기억장치와 보조기억장치 사이의 자료 전송, 파일의 조작 및 처리, 입·출력 자료와 프로그램 간의 논리적 연결 등 시스템에서 취급하는 파일과 데이터를 표준적인 방법으로 처리할 수 있도록 관리한다.

- 가. 문제 프로그램(Problem Program)
나. 감시 프로그램(Supervisor Program)
다. 작업 제어 프로그램(Job Control Program)
라. 데이터 관리 프로그램(Data Management Program)

47. UNIX에서 I-node는 한 파일이나 디렉터리에 관한 모든 정보를 포함하고 있는데, 이에 해당하지 않는 것은?

- 가. 파일이 가장 처음 변경된 시간 및 파일의 타입
나. 파일 소유자의 사용자 번호
다. 파일이 만들어진 시간
라. 데이터가 담긴 블록의 주소

- 가. 상호 배제 조건의 부정
- 나. 점유 및 대기 조건의 부정
- 다. 비선점 조건의 부정
- 라. 환형 대기 조건의 부정

가. 20 나. 50
다. 80 라. 100

- 가. Ring Connected
- 나. Multi Access Bus Connected
- 다. Partially Connected
- 라. Fully Connected

가. 다중 프로그래밍 시스템
나. 시분할 시스템
다. 일괄처리 시스템
라. 분산처리 시스템

- 가. 페이지 맵 테이블(Page Map Table)이 필요하다.
- 나. 세그먼트이션은 프로그램을 여러 개의 블록으로 나누어 수행한다.
- 다. 각 세그먼트는 고유한 이름과 크기를 갖는다.
- 라. 기억장치 보호키가 필요하다.

- 가. 1단계 디렉터리 구조
- 나. 2단계 디렉터리 구조
- 다. 트리 디렉터리 구조
- 라. 일반 그래프형 디렉터리 구조

- 가. 프로젝트 관리가 용이하다.
- 나. 노력과 비용이 절감된다.
- 다. 요구사항을 충실히 반영한다.
- 라. 관리와 개발이 명백히 구분된다.

63. 소프트웨어 품질 목표에 대한 설명으로 옳지 않은 것은?

- 가. 신뢰성(Reliability) : 정확하고 일관된 결과를 얻기 위해 요구된 기능을 수행하는 정도
- 나. 이식성(Protability) : 다양한 하드웨어 환경에서도 운용 가능하도록 쉽게 수정될 수 있는 정도
- 다. 상호 운용성(Interoperability) : 다른 소프트웨어와 정보를 교환할 수 있는 정도
- 라. 사용 용이성(Usability) : 전체나 일부 소프트웨어가 다른 응용 목적으로 사용될 수 있는 정도

64. 프로젝트 계획 수립 시 소프트웨어 범위(Scope) 결정의 주요 요소로 거리가 먼 것은?

- 가. 소프트웨어 개발 환경
- 나. 소프트웨어 성능
- 다. 소프트웨어 제약 조건
- 라. 소프트웨어 신뢰도

65. 소프트웨어 품질 보증을 위한 정형 기술 검토의 지침사항으로 옳지 않은 것은?

- 가. 논쟁과 반박의 제한성
- 나. 의제의 무제한성
- 다. 제품 검토의 집중성
- 라. 참가인원의 제한성

66. 자료 흐름도의 구성 요소가 아닌 것은?

- | | |
|--------------------|---------------|
| 가. Process | 나. Data Store |
| 다. Data Dictionary | 라. Terminator |

67. 객체지향 시스템에서 전통적 시스템의 함수(Function) 또는 프로시저(Procedure)에 해당하는 연산기능을 무엇이라고 하는가?

- 가. 메소드(Method) 나. 메시지(Message)
다. 모듈(Module) 라. 패키지(Package)

68. 소프트웨어 개발의 생산성에 영향을 미치는 요소로 거리가 먼 것은?

- 가. 프로그래머의 능력
- 나. 팀 의사 전달
- 다. 제품의 복잡도
- 라. 소프트웨어 사용자의 능력

69. 자료 사전에서 반복을 의미하는 기호는?

- 가. () 나. { }
- 다. [] 라. +

70. 럼바우의 객체지향 분석 모델링(Modeling)에 해당하지 않는 것은?

- 가. Relational Modeling
나. Object Modeling
다. Functinal Modeling
라. Dynamic Modeling

71. 소프트웨어 재사용에 대한 설명으로 옳지 않은 것은?

- 가. 개발 시간과 비용을 감소시킨다.
- 나. 프로젝트 실패의 위험을 줄여 준다.
- 다. 재사용 부품의 크기가 작을수록 재사용률이 낮다.
- 라. 소프트웨어 개발자의 생산성을 증가시킨다.

72. 소프트웨어 생명주기 모형 중 Boehm이 제시한 고전적 생명주기 모형으로서 선형 순차적 모델이라고도 하며, 타당성 검토, 계획, 요구사항 분석, 설계, 구현, 테스트, 유지보수의 단계를 통해 소프트웨어를 개발하는 모형은?

- 가. 폭포수 모형 나. 프로토타입 모형
다. 나선형 모형 라. RAD 모형

73. 공학적으로 좋은 소프트웨어에 대한 설명으로 적합하지 않은 것은?

- 가. 사용법, 구조의 설명, 성능, 기능이 이해하기 쉬어야 한다.
- 나. 유지보수가 용이해야 한다.
- 다. 실행 속도가 빠르고, 소요 기억 용량을 많이 차지할수록 좋다.
- 라. 사용자 수준에 따른 적당한 사용자 인터페이스를 제공해야 한다.

74. 객체지향 기법의 캡슐화(Encapsulation)에 대한 설명으로 거리가 먼 것은?

- 가. 변경 발생 시 오류의 파급효과가 적다.
- 나. 인터페이스가 단순화된다.
- 다. 소프트웨어 재사용성이 높아진다.
- 라. 상위 클래스의 모든 속성과 연산을 하위 클래스가 물려받는 것을 의미한다.

75. CASE에 대한 설명으로 거리가 먼 것은?

- 가. 자동화된 기법을 통해 소프트웨어 품질이 향상된다.
- 나. 소프트웨어 부품의 재사용성이 향상된다.
- 다. 프로토타입 모델에 위협 분석 기능을 추가한 생명주기 모형이다.
- 라. 소프트웨어 도구와 방법론의 결합이다.

76. 화이트 박스 검사로 찾기 힘든 오류는?

- 가. 논리 흐름도 나. 루프 구조
다. 자료 구조 라. 순환 복잡도

77. 소프트웨어 유지보수 유형 중 현재 수행 중인 기능의 수정, 새로운 기능의 추가, 전반적인 기능 개선 등의 요구를 사용자로부터 받았을 때 수행되는 유형으로서, 유지보수 유형 중 제일 많은 비용이 소요되는 것은?

- 가. Preventive Maintenance
나. Adaptive Maintenance
다. Corrective Maintenance
라. Perfective Maintenance

78. 소프트웨어 프로젝트 관리의 효과적 수행을 위한 3P에 해당하지 않는 것은?

- 가. Program 나. People
다. Problem 라. Process

79. 블랙 박스 검사에 대한 설명으로 옳지 않은 것은?

- 가. 각 기능이 완전히 작동되는 것을 입증하는 검사이다.
- 나. 인터페이스 결합, 성능 결합, 초기화와 종료 이상 결합 등을 찾아낸다.
- 다. 동치 분할 검사는 논리적인 조건과 대응하는 행동을 간략히 표현할 수 있도록 하는 검사 사례 설계 기법이다.
- 라. 경계값 분석은 입력의 경계값에서 발생하는 오류를 제거하기 위한 검사 기법이다.

80. 다음 중 공학적으로 잘 작성된 소프트웨어가 갖는 특성으로 옳은 것은?

- 가. 원하는 요구사항 중에 중요한 사항만 반영한다.
나. 유지보수 비용이 많이 들어간다.
다. 신뢰성이 떨어지더라도 효율성이 높다.
라. 사용자가 손쉽게 사용할 수 있다.

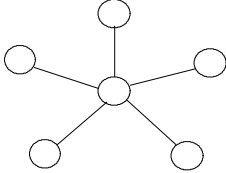
81. 다음 중 전용 전송로를 사용하는 방식은?

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

82. 문자의 시작과 끝에 START 비트와 STOP 비트가 부가되어 전송의 시작과 끝을 알려 전송하는 방식은?

- 가. 비동기식 전송 나. 동기식 전송
다. 전송 동기 라. PCM 전송

83. 다음 LAN의 네트워크 토폴로지(Topology)는 어떤 형인가?



- 가. 링형 나. 성형
다. 버스형 라. 트리형

84. OSI 7계층 구조로 최하위 계층부터 최상위 계층의 순서가 옳은 것은?

- 가. Physical Layer → Network Layer → Transport Layer → Data Link Layer → Session Layer → Presentation Layer → Application Layer
나. Physical Layer → Network Layer → Data Link Layer → Transport Layer → Session Layer → Presentation Layer → Application Layer
다. Physical Layer → Data Link Layer → Network Layer → Transport Layer → Session Layer → Presentation Layer → Application Layer
라. Physical Layer → Data Link Layer → Network Layer → Transport Layer → Presentation Layer → Session Layer → Application Layer

85. HDLC(High-Level Data Link Control)에 관련된 설명이 아닌 것은?

- 가. 비트 지향형 전송을 한다.
나. CRC 방식을 이용하여 오류 제어를 한다.
다. 정지 및 대기 방식을 사용한다.
라. 정보 프레임과 감독 프레임 등이 있다.

86. 인터넷 상에서 도메인 주소를 IP 주소로 변환하여 주는 서버를 무엇이라고 하는가?

- 가. 웹 서버 나. DNS 서버
다. 파일 서버 라. 팝 서버

87. 다음에서 데이터 링크의 전송 제어 절차의 순서가 올바른 것은?

- ① 데이터 전송 ② 회선 접속 ③ 데이터 링크 확립
④ 회선 절단 ⑤ 링크 종료

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

88. TCP 헤더에 포함되는 정보가 아닌 것은?

- 가. 긴급 포인터 나. 호스트 주소
다. 순서 번호 라. 체크섬

89. 다음 중 자유 경쟁으로 채널 사용권을 확보하는 방법으로, 노드 간의 충돌을 허용하는 네트워크 접근 방법은?

- 가. Slotted Ring 나. Token Passing
다. CSMA/CD 라. Polling

90. HDLC에서 피기백킹(Piggybacking) 기법을 통해 데이터에 대한 확인 응답을 보낼 때 사용되는 프레임은?

- 가. I-프레임 나. S-프레임
다. U-프레임 라. A-프레임

91. 다음 중 IP의 라우팅 프로토콜이 아닌 것은?

- 가. IGP 나. RIP
다. EGP 라. HDLC

92. 전송 시간을 일정한 간격의 시간 슬롯(Time Slot)으로 나누고, 이를 주기적으로 각 채널에 할당하는 다중화 방식은?

- 가. 주파수 분할 다중화
나. 동기식 시분할 다중화
다. 코드 분할 다중화
라. 공간 분할 다중화

93. 전송 오류 제어 중 오류가 발생한 프레임뿐만 아니라 오류 검출 이후의 모든 프레임을 재전송하는 ARQ 방식은?

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

94. PCM(Pulse Code Modulation) 과정에 포함되지 않는 것은?

- 가. 다중화 나. 샘플링
다. 양자화 라. 부호화

95. 주파수 분할 다중화에서 인접한 채널 간의 간섭을 방지하기 위한 대역은?

- 가. Buffer 나. Slot
다. Channel 라. Guard Band

96. 호스트의 물리 주소를 통해 논리 주소인 IP 주소를 얻어오기 위해 사용되는 프로토콜은?

- 가. ICMP 나. IGMP
다. ARP 라. RARP

97. 다음 중 멀티 포인터(Multi-Point) 방식에서 보조국(Secondary Station)이 주국(Primary Station)에게 보낼 데이터를 갖고 있는지 주국에서 확인하는 방식은?

- 가. 폴링(Polling) 나. 셀렉션(Selection)
다. 요청(Request) 라. 응답(Response)

98. 다음 중 패킷 교환망의 설명으로 틀린 것은?

- 가. 가상 회선 방식과 데이터그램 방식이 있다.
나. 전송에 실패한 패킷의 경우 재전송이 가능하다.
다. 패킷 단위로 헤더를 추가하므로 패킷별 오버헤드가 발생한다.
라. 실시간 전송이나, 다량의 데이터 전송에 적합하다.

99. 어느 회선의 속도가 400[Baud]이고, 각 신호가 4비트의 정보를 나타낸다면 데이터 전송률은 몇 Bps인가?

- 가. 400 나. 800
다. 1,600 라. 3,200

100. 다음 중 아날로그 데이터를 디지털 신호로 변환하는 것은?

- 가. PCM(Pulse Code Modulation)
나. AM(Amplitude Modulation)
다. PSK(Phase Shift Keying)
라. FDM(Frequency Division Multiplexing)

정답 및 해설

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

1 외래 키의 속성명과 참조하려는 테이블의 기본 키의 속성명이 동일할 필요는 없습니다. 문제에 제시된 다른 보기를 통해 참조 무결성의 특징을 정리하고, 개체 무결성과 도메인 무결성의 의미도 숙지하세요.

■ **개체 무결성** : 릴레이션에서 기본 키를 구성하는 속성은 널(NULL) 값이나 중복값을 가질 수 없음

■ **도메인 무결성** : 특정 속성의 값이 그 속성이 정의된 도메인에 속한 값이어야 한다는 규정

2 2단계 로킹(Two Phase Locking) 규약은 직렬성을 보장하는 대표적인 로킹 규약으로서, 직렬성을 보장하는 장점은 있지만 교착 상태를 예방할 수 없다는 단점이 있습니다. 처음 출제된 문제입니다. 보기를 통해 2단계 로킹 규약에 대한 특징을 간단하게 정리하고 넘어가세요.

3 데이터베이스를 운용하는 소프트웨어는 DBMS입니다. 스키마의 정의와 특징은 꼭 기억해야 합니다.

스키마(Schema)

■ 스키마는 데이터베이스의 구조와 제약조건에 관한 전반적인 명세(Specification)를 기술(Description)한다.

■ 스키마는 데이터베이스를 구성하는 데이터 개체(Entity), 속성(Attribute), 관계(Relationship) 및 데이터 조작 시 데이터 값들이 갖는 제약조건 등에 관해 전반적으로 정의한다.

■ 스키마는 사용자의 관점에 따라 외부 스키마, 개념 스키마, 내부 스키마로 나뉜다.

■ 스키마(Schema)는 데이터 사전에 저장되며, 다른 이름으로 메타 데이터(Meta-Data)라고도 한다.

4 한 릴레이션을 구성하는 속성 사이에는 일정한 순서가 없습니다. 자주 출제되는 내용입니다. 릴레이션의 특징을 무조건 암기하지 말고 다음에 주어진 예를 <학생> 릴레이션에 적용시켜 보면서 이해하세요.

<학생> 릴레이션

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

■ 한 릴레이션에 포함된 튜플들은 모두 상이하다.

예) <학생> 릴레이션을 구성하는 홍길동 레코드는 홍길동에 대한 학적 사항을 나타내는 것으로 <학생> 릴레이션 내에서는 유일하다.

■ 한 릴레이션에 포함된 튜플 사이에는 순서가 없다.

예) <학생> 릴레이션에서 홍길동 레코드와 임꺽정 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 순서는 서로 바뀌어도 상관없다.

■ 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 속성명들과 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) '학번'에 저장된 1, 2, 4 등은 더 이상 세분화할 수 없다.

5 사용자가 일반 질의어를 이용해 시스템 카탈로그의 내용을 검색할 수는 있지만 직접 갱신할 수는 없습니다. 자주 출제되는 내용입니다. 시스템 카탈로그의 특징을 정리하세요.

시스템 카탈로그의 특징

■ 카탈로그 자체도 시스템 테이블로 구성되어 있어 일반 이용자도 SQL을 이용하여 내용을 검색해 볼 수 있다.

■ INSERT, DELETE, UPDATE문으로 카탈로그를 갱신하는 것은 허용되지 않는다.

■ DBMS가 스스로 생성하고 유지한다.

■ 카탈로그는 사용자가 SQL문을 실행시켜 기본 테이블, 뷰, 인덱스 등에 변화를 주면 시스템이 자동으로 갱신된다.

6 트랜잭션(Transaction)이 가져야 할 특성으로는 원자성(Atomicity), 일관성(Consistency), 독립성(Isolation, 격리성), 영속성(Durability, 지속성)이 있습니다. 이 문제처럼 해석을 달리하여 출제되는 경우도 있으니 트랜잭션의 특성을 꼭 영문과 함께 기억해 두세요.

트랜잭션의 특성

■ Atomicity(원자성)

- 트랜잭션의 연산은 데이터베이스에 모두 반영되었는지 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 트랜잭

선에서 수행 결과를 참조할 수 없다.

■ Durability(영속성, 지속성)

성공적으로 완료된 트랜잭션의 결과는 영구적으로 반영되어야 한다.

- 7 타원(Ellipses)은 속성을 표시합니다. 거의 매회 출제되는 내용입니다. E-R 다이어그램에서 사용되는 기호는 반드시 암기해야 합니다.

- 사각형 : 개체(Entity) 타입
- 다이아몬드 : 관계(Relationship) 타입
- 타원 : 속성(Attribute)
- 밑줄 타원 : 기본키 속성
- 선, 링크 : 개체 타입과 속성을 연결

- 8 운영체제의 작업 스케줄링 등에 응용될 수 있는 가장 적합한 자료 구조는 먼저 들어온 자료를 먼저 처리하는 큐입니다. 한쪽으로는 입력만, 다른 한쪽으로는 출력만 가능한 큐의 개념과 용도를 알아두세요.

큐(Queue)의 개념과 용도

- 선형 리스트의 한쪽에서는 삽입 작업이 이루어지고 다른 쪽에서는 삭제 작업이 이루어지도록 구성된 자료 구조이다.
- 가장 먼저 삽입된 자료가 가장 먼저 삭제되는 선입선출(FIFO, First In First Out) 방식으로 처리한다.
- 프런트(F, Front) 포인터
 - 가장 먼저 삽입된 자료의 기억공간을 가리키는 포인터이다.
 - 삭제 작업을 할 때 사용한다.
- 리어(R, Rear)포인터
 - 가장 마지막에 삽입된 자료가 위치한 기억장소를 가리키는 포인터이다.
 - 삽입 작업을 할 때 사용한다.
- Queue를 이용하는 예
 - 창구업무처럼 서비스 순서를 기다리는 등의 대기 행렬의 처리에 사용한다.
 - 운영체제의 작업 스케줄링에 사용한다.

- 9 연결 리스트(Linked List)는 노드들이 포인터로 연결되어 포인터를 찾아가는 시간이 필요하므로 선형 리스트에 비해 검색 속도가 느립니다. 연결 리스트의 특징을 정리하세요.

연결 리스트(Linked List)

- 연결 리스트는 자료들을 반드시 연속적으로 배열시키지는 않고 임의의 기억공간에 기억시키되, 자료 항목의 순서에 따라 노드의 포인터 부분을 이용하여 서로 연결시킨 자료 구조이다.
- 노드의 삽입, 삭제 작업이 용이하다.
- 기억공간이 연속적으로 놓여 있지 않아도 저장이 가능하다.
- 연결을 위한 링크(포인터) 부분이 필요하기 때문에 순차 리스트에 비해 기억공간 이용 효율이 좋지 않다.
- 접근 속도가 느리다.
- 연결 리스트 중에서 중간 노드의 연결이 끊어지면 그 다음 노드를 찾기 힘들다.
- 희소 행렬을 링크드 리스트로 표현하면 기억장소가 절약된다.
- ※ **희소 행렬(Sparse Matrix)** : 행렬의 요소 중 많은 항들이 0으로 되어 있는 형태로, 기억장소를 절약하기 위해 링크드 리스트를 이용하여 저장함

- 10 특정 테이블에 대하여 WHERE 조건절이 없는 DELETE 명령을 수행하면 특정 테이블내의 데이터만 모두 삭제됩니다.

테이블 구조까지 제거하는 DROP TABLE 명령과는 기능이 다릅니다. DELETE 문의 특징을 간단히 정리하고 넘어가세요.

삭제문(DELETE FROM ~)

- 기본 테이블에 있는 튜플들 중에서 특정 튜플을 삭제할 때 사용한다.

```
DELETE
FROM 테이블명
WHERE 조건;
```

- 모든 레코드를 삭제할 때는 WHERE절을 생략한다.
- 모든 레코드를 삭제하더라도 테이블 구조는 남아 있기 때문에 디스크에서 테이블을 완전히 제거하는 DROP과는 다르다.

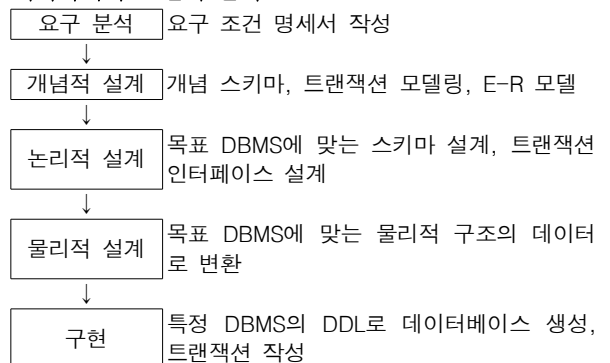
- 11 중복성 보장이 아니라, 동일 데이터가 여러 곳에 중복되어 있더라도 사용자는 마치 하나의 데이터만 존재하는 것처럼 사용하고, 시스템은 자동으로 여러 자료에 대한 작업을 수행할 수 있는 중복 투명성(Replication Transparency)이 제공되어야 하며, 가용성은 높여야 합니다. 분산 데이터베이스에서는 분산 데이터베이스의 4대 목표에 대한 내용이 자주 출제됩니다. 번역을 달리하여 출제되는 경우도 있으니 영어 원문과 함께 각 내용을 정확하게 숙지하세요.

분산 데이터베이스의 4대 목표

위치 투명성 (Location Transparency)	액세스하려는 데이터베이스의 실제 위치를 알 필요없이 단지 데이터베이스의 논리적인 명칭만으로 액세스할 수 있음
중복(복제)투명성 (Replication Transparency)	동일 데이터가 여러 곳에 중복되어 있더라도 사용자는 마치 하나의 데이터만 존재하는 것처럼 사용하고, 시스템은 자동으로 여러 자료에 대한 작업을 수행함
병행 투명성 (Concurrency Transparency)	분산 데이터베이스와 관련된 다수의 트랜잭션의 결과는 영향을 받지 않음
장애 투명성 (Failure Transparency)	트랜잭션, DBMS, 네트워크, 컴퓨터 장애에도 불구하고 트랜잭션을 정확하게 처리함

- 12 데이터베이스 설계는 순서뿐만 아니라 각각의 단계에서 수행하는 작업도 개략적으로 알고 있어야 합니다.

데이터베이스 설계 순서



- 13 스키마의 평가는 논리적 설계 단계에서 수행합니다. 물리적 설계에서는 물리적 데이터베이스의 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

- 14 스택은 순서 리스트로서 Top이라 불리는 한쪽 끝에서 삽입과 삭제가 일어납니다. 자주 출제되는 내용입니다. 스택의 개념과 특징을 꼭 숙지하고 넘어가세요.

스택(Stack)

- 스택은 리스트의 한쪽 끝으로만 자료의 삽입, 삭제 작업이 이루어지는 자료 구조이다.
- 스택은 가장 나중에 삽입된 자료가 가장 먼저 삭제되는 후입선출(LIFO, Last In First Out) 방식으로 자료를 처리한다.
- TOP
 - Stack으로 할당된 기억공간에 가장 마지막으로 삽입된 자료가 기억된 위치를 가리키는 요소
 - 스택 포인터(SP, Stack Pointer)라고도 한다.
- Bottom : 스택의 가장 밑바닥이다.

스택(Stack)의 사용 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 순서 제어
- 컴파일러를 이용한 언어 번역 시

- 15 문제에 제시된 내용은 데크에 대한 설명입니다. 종종 출제되는 내용입니다. 데크의 개념과 특징을 정리하고 넘어가세요.

데크(Deque)

- 삽입과 삭제가 리스트의 양쪽 끝에서 모두 발생할 수 있는 자료 구조이다.
- Double Ended Queue의 약자이다.
- Stack과 Queue의 장점만 따서 구성한 것이다.
- 입력이 한쪽에서만 발생하고 출력은 양쪽에서 일어날 수 있는 입력 제한과, 입력은 양쪽에서 일어나고 출력은 한쪽에서만 이루어지는 출력 제한이 있다.
- 입력 제한 데크 : Scroll
- 출력 제한 데크 : Shelf

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

- 17 후위 표기 방식(Postfix)은 중위 표기 방식(Infix)으로 된 수식에서 연산자를 해당 피연산자 2개의 뒤(오른쪽)로 이동한 것입니다.

① 먼저 연산 우선순위에 따라 괄호로 묶습니다.

$$(A = (((B - C) * D) + E))$$

② 연산자를 해당 괄호의 뒤로 옮깁니다.

$$(A = (((B - C) * D) + E)) \rightarrow (A(((B - C) - D) * E) +) =$$

③ 필요 없는 괄호를 제거합니다.

$$A B C - D * E + =$$

- 18 데이터 모델의 3요소는 구조(Structure), 연산(Operation), 제약조건(Constraint)입니다. 세 가지 요소의 의미 보다는 세 가지 요소를 암기하면 맞힐 수 있는 문제가 주로 출제됩니다. 종류 세 가지는 꼭 암기하고 각각 의미는 가볍게 읽어보세요.

데이터 모델에 표시할 사항

- 구조(Structure) : 논리적으로 표현된 개체들 간의 관계를 표시함
- 연산(Operation) : 데이터베이스에 저장된 실제 데이터를 처리하는 방법을 표시함
- 제약조건(Constraint) : 데이터베이스에 저장될 수 있는 실제 데이터의 논리적인 제약조건을 표시함

- 19 GRANT는 DCL이고 나머지는 DML입니다. 각 언어의 특징을 파악하고 각각에는 어떤 명령들이 있는지 구분할 수 있도록 공부하세요.

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.
- 명령어 : CREATE, ALTER, DROP

■ DML(데이터 조작어)

- DML(Data Manipulation Language)은 데이터베이스 사용자가 응용 프로그램이나 질의어를 통하여 저장된 데이터를 실질적으로 처리하는 데 사용되는 언어이다.
- 데이터베이스 사용자와 데이터베이스 관리 시스템간의 인터페이스를 제공한다.
- 명령어 : SELECT, INSERT, DELETE, UPDATE

■ DCL(데이터 제어어)

- DCL(Data Control Language)은 데이터의 보안, 무결성, 회복, 병행 수행 제어 등을 정의하는 데 사용되는 언어이다.
- 데이터베이스 관리자가 데이터 관리를 목적으로 사용한다.
- 명령어 : COMMIT, ROLLBACK, GRANT, REVOKE

- 20 문제에 제시된 내용은 개념 스키마에 대한 설명입니다. 3계층 스키마에 대한 문제는 거의 매회 출제됩니다. 3단계 스키마의 종류와 각각의 의미를 확실히 정리하세요.

외부 스키마(External Schema) = 서브 스키마 = 사용자 뷰(View)

- 외부 스키마(External Schema)는 사용자나 응용 프로그램이 각 개인의 입장에서 필요로 하는 데이터베이스의 논리적 구조를 정의한 것이다.

- 외부 스키마(External Schema)는 전체 데이터베이스의 한 논리적인 부분으로 볼 수 있으므로 서브 스키마(Subschema)라고도 한다.
- 하나의 데이터베이스 시스템에는 여러 개의 외부 스키마가 존재할 수 있으며, 하나의 외부 스키마를 여러 개의 응용 프로그램이나 사용자가 공유할 수도 있다.
- 같은 데이터베이스에 대해서도 서로 다른 관점을 정의할 수 있도록 허용한다.
- 일반 사용자는 질의어(SQL)를 이용하여 DB를 쉽게 사용할 수 있다.
- 응용 프로그래머는 COBOL, C 등의 언어를 사용하여 DB에 접근한다.

개념 스키마(Conceptual Schema) = 전체적인 뷰(View)

- 개념 스키마는 데이터베이스의 전체적인 논리적 구조로서, 모든 응용 프로그램이나 사용자들이 필요로 하는 데이터를 종합한 조직 전체의 데이터베이스로 하나만 존재한다.
- 개념 스키마는 개체 간의 관계와 제약 조건을 나타내고 데이터베이스의 접근 권한, 보안 및 무결성 규칙에 관한 명세를 정의한다.
- 단순히 스키마(Schema)라고하면 개념 스키마를 의미한다.
- 기관이나 조직체의 관점에서 데이터베이스를 정의한 것이다.
- 데이터베이스 관리자(DBA)에 의해서 구성된다.

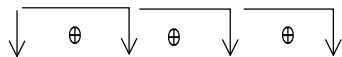
내부 스키마(Internal Schema)

- 내부 스키마는 데이터베이스의 물리적 구조이다.
- 내부 스키마는 데이터의 실제 저장 방법을 기술한다.
- 물리적인 저장장치와 밀접한 계층이다.
- 시스템 프로그래머나 시스템 설계자가 보는 관점의 스키마이다.

- 21 2진수를 Gray Code로 변환하는 방법은 다음과 같습니다.
 ① 그레이 코드의 첫 번째 비트는 2진수를 그대로 내려 씁니다.

- ② 두 번째 그레이 비트부터는 변경할 2진수의 해당 번째 비트와 그 왼쪽의 비트를 XOR 연산하여 씁니다.

2진수 : 1 0 1 1



Gray Code : 1 1 1 0

- 22 2의 보수로 표현된 수치는 먼저 10진수로 변경하여 계산한 후 2의 보수로 변환하면 됩니다.

- ① 첫 번째 비트가 1이므로 음수값입니다. 음수값은 다시 2의 보수를 취해야만 그 값의 크기를 알 수 있습니다.

101011과 100110에 대한 2의 보수를 구한 후 10진수로 변환합니다.

101011 → 010101 → -21

100110 → 011010 → -26

- ② 계산합니다.

-21 - (-26) = -21 + 26 = 5

- ③ 이진수로 변환합니다.

5 → 000101

- 23 연산자의 종류는 OP-Code(연산자 부)의 길이와 관계가 있습니다. OP-Code가 n 비트이면 사용할 수 있는 연산자의 종류는 2^n 개입니다.

- 첫 번째 명령의 OP-Code가 3비트이므로 $2^3 = 8$ 개의 연산자를 사용할 수 있습니다.
- 두 번째 명령의 OP-Code가 6비트이므로 $2^6 = 64$ 개의 연

산자를 사용할 수 있습니다.

중중 출제되는 문제입니다. 꼭 이해하고 넘어가세요.

- 24 그림에 맞게 F로 입력되는 4개의 디코더 출력선을 구해보면 다음과 같습니다.

A	B	번호	출력선	F
0	0	0	\overline{AB}	1
0	1	1	\overline{AB}	0
1	0	2	\overline{AB}	0
1	1	3	AB	1

디코더의 출력선 중 1번과 2번은 절단되었고, 0번이나 3번이 선택되었을 때 출력 F가 1이 되므로 $F = \overline{AB} + AB$ 입니다. 자주 출제되는 문제는 아닙니다. 한번 읽어보고 이해 안되면 답만 기억하고 넘어가세요.

- 25 제어 마이크로 명령이란 마이크로 명령 형식은 없습니다. 마이크로 명령의 형식에는 수평 마이크로 명령, 수직 마이크로 명령, 나노 명령이 있다는 것만 기억해두고 넘어가세요.

- 26 인수의 형(Type)은 매크로의 인수 사용에 해당되지 않습니다. 처음 출제된 문제입니다. 문제와 답만 기억하고 넘어가세요.

- 27 트랩(Trap)은 내부 인터럽트를 의미하므로 내부 인터럽트가 아닌 것을 찾으시면 됩니다. 패리티 오류는 전송된 데이터에 오류가 발생한 것으로 인터럽트와는 무관합니다. 인터럽트의 종류 및 발생 원인은 거의 매회 출제되고 있습니다. 꼭 정리하세요.

외부 인터럽트

외부 인터럽트는 다음과 같이 입·출력 장치, 타이밍 장치, 전원 등 외부적인 요인에 의해 발생한다.

- **전원 이상 인터럽트(Power Fail Interrupt)** : 정전이 되거나 전원 이상이 있는 경우

- **기계 착오 인터럽트(Machine Check Interrupt)** : CPU의 기능적인 오류 동작이 발생한 경우

- **외부 신호 인터럽트(External Interrupt)**

- 타이머에 의해 규정된 시간(Time Slice)을 알리는 경우
- 키보드로 인터럽트 키를 누른 경우
- 외부장치로부터 인터럽트 요청이 있는 경우

- **입·출력 인터럽트(Input-Output Interrupt)**

- 입·출력 Data의 오류나 이상 현상이 발생한 경우
- 입·출력 장치가 데이터의 전송을 요구하거나 전송이 끝났음을 알릴 경우

내부 인터럽트

내부 인터럽트는 다음과 같이 잘못된 명령이나 데이터를 사용할 때 발생하며, 트랩(Trap)이라고도 부른다.

- **명령어 잘못에 의한 인터럽트** : 프로그램에서 명령어를 잘못 사용한 경우

- **프로그램 검사 인터럽트(Program Check Interrupt)** : 0으로 나누거나, Overflow 또는 Underflow가 발생한 경우

소프트웨어 인터럽트

소프트웨어 인터럽트는 프로그램 처리 중 명령의 요청에 의해 발생하는 것으로, 가장 대표적인 형태는 감시 프로그램을 호출하는 SVC(SuperVisor Call) 인터럽트가 있다.

- **SVC(SuperVisor Call) 인터럽트** : 사용자가 SVC 명령을 써서 의도적으로 호출한 경우

- 28 MODEM은 중앙연산처리장치의 하드웨어적인 요소가 아님

니다. 중앙연산처리장치의 주요 구성 요소에는 제어장치, 연산장치, 레지스터, 버스가 있습니다. IR, MAR, PC는 레지스터의 종류입니다. 이 문제에서는 중앙처리장치의 구성 요소만 기억하고 넘어가세요.

- 29 연관기억장치의 목적은 저장 공간의 확대가 아니라 접근 속도의 향상입니다. 연관 기억장치는 개념과 특징을 숙지하세요.

연관 기억장치(Associative Memory)

- 기억장치에서 자료를 찾을 때 주소에 의해 접근하지 않고, 기억된 내용의 일부를 이용하여 Access할 수 있는 기억장치로, CAM(Content Addressable Memory)이라고도 한다.
- 주소에 의해서만 접근이 가능한 기억장치보다 정보 검색이 신속하다.
- 캐시 메모리나 가상 메모리 관리 기법에서 사용하는 Mapping Table에 사용된다.
- 외부의 인자와 내용을 비교하기 위한 병렬 판독 논리 회로를 갖고 있기 때문에 하드웨어 비용이 증가한다.

- 30 문제에 제시된 그림은 디코더입니다. 이 문제에서는 보기에 제시된 회로들의 특징들만 간단히 정리하세요.

- **멀티플렉서(Multiplexer)** : 2^n 의 입력선 중 1개를 선택하여 그 선으로부터 입력되는 값을 1개의 출력선으로 출력시키는 회로
- **인코더(Encoder)** : 2^n 개의 입력선으로 입력된 값을 n 개의 출력선으로 코드화해서 출력하는 회로
- **쉬프터(Shifter)** : 특정 비트의 자리를 이동하는 회로

- 31 16진수를 8진수로 변환할 때는 먼저 16진수를 2진수로 변환하고, 변환된 2진수를 다시 8진수로 변환하면 쉽습니다.

① 16진수 1자리를 2진수 4자리로 변환합니다.

(A 4 D)₁₆
 $\downarrow \quad \downarrow \quad \downarrow$
 (1010 0100 1101)₂

② 오른쪽에서 왼쪽 방향으로, 2진수를 3자리씩 묶어 8진수 1자리로 표현합니다.

(101 001 001 101)₂
 $\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 (5 1 1 5)₈

- 32 누산기(Accumulator)는 연산장치에 있는 레지스터(Register)의 하나로 연산 결과를 기억하는 등 연산장치의 핵심 역할을 하는 레지스터입니다. 자주 출제되는 내용입니다. 주요 레지스터의 기능을 숙지하세요.

주요 레지스터의 종류 및 기능

- **프로그램 카운터, 프로그램 계수기(PC, Program Counter)** : 다음 번에 실행할 명령어의 번지를 기억하는 레지스터
- **명령 레지스터(IR, Instruction Register)** : 현재 실행 중인 명령의 내용을 기억하는 레지스터
- **누산기(AC, Accumulator)** : 연산된 결과를 일시적으로 저장하는 레지스터로 연산의 중심이다.
- **상태 레지스터(Status Register), PSWR(Program Status Word Register), 플래그 레지스터** : 시스템 내부의 순간순간의 상태가 기록된 정보를 PSW라고 한다. 오버플로, 언더플로, 자리올림, 인터럽트 등의 PSW를 저장하고 있는 레지스터
- **메모리 주소 레지스터(MAR, Memory Address Register)** : 기억장치를 출입하는 데이터의 번지를 기억하는 레지스터

- **메모리 버퍼 레지스터(MBR, Memory Buffer Register)** : 기억장치를 출입하는 데이터가 잠시 기억되는 레지스터
- **인덱스 레지스터(Index Register)** : 주소의 변경이나 프로그램에서의 반복연산의 횟수를 세는 레지스터
- **데이터 레지스터(Data Register)** : 연산에 사용될 데이터를 기억하는 레지스터
- **Shift Register** : 저장된 값을 왼쪽 또는 오른쪽으로 1Bit씩 자리를 이동시키는 레지스터, 2배 길이 레지스터라고도 함
- **메이저 스테이터스 레지스터(Major Status Register)** : CPU의 메이저 상태를 저장하고 있는 레지스터

- 33 프로그램 수행 도중 서로 다른 번지의 주소를 동시에 지정하는 방식은 인터리빙입니다. 메모리 인터리빙에 대해서는 개념에 대한 문제가 많이 출제됩니다. 개념 및 특징을 꼭 숙지하세요.

메모리 인터리빙(Memory Interleaving)

- 여러 개의 독립된 모듈로 이루어진 복수 모듈 메모리와 CPU 간의 주소 버스가 한 개로만 구성되어 있으면 같은 시각에 CPU로부터 여러 모듈들로 동시에 주소를 전달할 수 없기 때문에, CPU가 각 모듈로 전송할 주소를 교대로 배치한 후 차례대로 전송하여 여러 모듈을 병행 접근하는 기법이다.
- CPU가 버스를 통해 주소를 전달하는 속도는 빠르지만 메모리 모듈의 처리 속도가 느리기 때문에 병행 접근이 가능하다.
- 메모리 인터리빙 기법을 사용하면 기억장치의 접근시간을 효율적으로 높일 수 있으므로 캐시 기억장치, 고속 DMA 전송 등에서 많이 사용된다.

- 34 인터럽트 수행 후에는 인터럽트 처리 시 보존시켰던 PC 및 제어상태 데이터를 PC와 제어상태 레지스터에 복구합니다. 인터럽트 동작 원리를 알아야 풀 수 있는 문제가 자주 출제됩니다. 인터럽트의 개념을 이해하면 쉽게 기억할 수 있습니다.

인터럽트(Interrupt)

프로그램을 실행하는 도중에 예기치 않은 상황이 발생할 경우 현재 실행중인 작업을 즉시 중단하고, 발생된 상황을 우선 처리한 후 실행중이던 작업으로 복귀하여 계속처리하는 것, 일명 “끼어들기”라고도 한다.

- 35 빈번히 참조되는 프로그램이나 데이터를 별도의 메모리에 저장하여 처리하는 것은 캐시 메모리입니다. 가상기억장치의 개요와 특징을 정리하세요.

가상기억장치(Virtual Memory)

- 기억용량이 작은 주기억장치를 마치 큰 용량을 가진 것처럼 사용할 수 있도록 하는 운영체제의 메모리 운영 기법으로, 사용자는 프로그램의 크기에 제한 받지 않고 프로그램의 실행이 가능하다.
- 가상기억장치의 목적은 주기억장치의 용량 확보이다.
- 가상기억장치는 하드웨어적으로 실제로 존재하는 것이 아니고 소프트웨어적인 방법으로 보조기억장치를 주기억장치처럼 사용하는 것이다.
- 사용자 프로그램을 여러 개의 작은 블록으로 나누어서 보조 기억장치상에 보관해놓고 프로그램 실행시 필요한 부분들만 주기억장치에 적재한다.
- 주기억장치의 이용률과 다중 프로그래밍의 효율을 높일 수 있다.
- 가상기억장치 기법에서 사용하는 보조기억장치는 디스크 같은 DASD(직접접근 기억장치) 장치이어야 한다.

- 36 스택(Stack)을 사용하는 주소 방식은 0-주소(Zero Addressing)

-ss) 방식입니다. 다른 주소 방식에서는 어떤 기억 요소를 사용하는지도 알아두세요.

■ 1-주소 : 누산기

■ 2-주소, 3-주소 : GPR(범용 레지스터)

37 인스트럭션의 성능 = 수행시간 / (패치시간+ 준비시간) = 10 / (5+3) = 1.25

처음 출제된 문제입니다. 상식적으로 생각할 수 있는 계산식이니 눈에 익혀두고 넘어가세요.

38 하드웨어의 신호에 의해 특정 번지의 서브루틴을 수행한다는 것은 백터에 의한 인터럽트 처리 방식을 의미합니다. 이 문제에서는 인터럽트 백터에 대한 개념만 명확히 하고 넘어가세요.

인터럽트 백터

■ 중앙처리장치는 인터럽트가 발생한 장치번호를 받은 후에는 해당되는 인터럽트 서비스(취급) 루틴으로 분기하게 된다. 이때 기억장치내의 특정한 곳에는 인터럽트 취급 루틴으로 분기하는 명령어들을 기억하는 영역이 있는데, 이를 인터럽트 백터라고 한다.

■ 인터럽트 백터에는 인터럽트가 발생했을 때 프로세서의 인터럽트 서비스가 특정의 장소로 점프하도록 점프할 분기 번지가 기억되어 있다.

39 Fixed Service는 우선순위 인터럽트 운영 방식이 아닙니다. 자주 출제되는 문제는 아닙니다. 문제와 답만 기억하고 넘어가세요.

40 정보량이 매우 적은 시스템은 하드웨어에 의한 우선순위 체제를 사용하는 것이 효과적입니다. 인터럽트 우선순위를 소프트웨어적으로 판별하는 것은 폴링이고, 하드웨어적으로 판별하는 것은 데이지 체인이라는 것을 기억하고 각각의 특징을 정리하세요.

하드웨어적인 인터럽트 우선순위 판별 방식 : 데이지 체인(Daisy-Chain) 방식

■ 인터럽트가 발생하는 모든 장치를 1개의 회선에 직렬로 연결한다.

■ 하드웨어적인 방법으로 판별 과정이 간단해서 응답 속도가 빠르다.

■ 회로가 복잡하고 융통성이 없으며 추가적인 하드웨어가 필요하므로 비경제적이다.

■ 우선순위가 높은 장치를 선두에 위치시키고 나머지를 우선순위에 따라 차례로 연결한다.

■ 백터에 의한 인터럽트 처리 방식이다.

■ 직렬 우선순위 부여 방식을 데이지 체인 방식이라고 한다.

소프트웨어적인 인터럽트 우선순위 판별 방법 : Polling

■ Interrupt 발생 시 우선순위가 가장 높은 인터럽트 자원(Source)부터 차례로 검사하여 찾고 이에 해당하는 인터럽트 서비스 루틴을 수행하는 방식이다.

■ 소프트웨어적인 방식을 폴링이라고 한다.

■ 많은 인터럽트가 있을 때 그들을 모두 조사하는 데 많은 시간이 걸려 반응시간이 느리다는 단점이 있다.

■ 회로가 간단하고 융통성이 있으며 별도의 하드웨어가 필요없으므로 경제적이다.

41 UNIX 파일 시스템의 구조는 부트 블록(Boot Block), 슈퍼블록(Super Block), I-node(Index-node) 블록, 데이터 블록으로 구성됩니다. UNIX 파일 시스템의 4가지 구성 요소를 기억해 두세요.

42 문제에 주어진 프로세스의 선행 순서를 정리하면 P1>P2>

P4, P3>P4가 됩니다. P1 프로세스가 가장 먼저 실행되고, 그 다음 P2 프로세스, 그 다음은 P4 프로세스가 실행됩니다. 그리고 P4 프로세스가 실행되기 전에 P3 프로세스가 실행되어야 하므로 P1과 P3, P2와 P3은 병행 처리가 가능하게 됩니다. 자주 출제되는 문제는 아닙니다. 문제에 주어진 내용만 이해하고 넘어가세요.

43 최초 적합(First Fit)은 프로그램이나 데이터가 들어갈 수 있는 크기의 빈 영역 중에서 첫 번째 분할 영역에 배치시키는 방법으로, 10K는 15K(B) 공백에 배치됩니다. 만약 최적 적합으로 배치한다면 15K 공백에, 최악 적합으로 배치한다면 25K 공백에 배치될 것입니다.

44 3개의 페이지를 수용할 수 있는 주기억장치이므로 다음 그림과 같이 표현할 수 있습니다.

참조 페이지	2	3	2	1	5	2	4	5	3	2	5	2
페이지	2	2	2	2	5	5	5	5	3	3	3	3
		3	3	3	3	2	2	2	2	2	5	5
				1	1	1	4	4	4	4	4	2
부재 발생	●	●		●	●	●	●		●		●	●

참조 페이지가 페이지에 없을 경우 페이지 결함(부재)이 발생합니다. 초기에는 모든 페이지가 비어 있으므로 처음 2, 3 페이지 적재 시 페이지 결함이 발생합니다. FIFO는 각 페이지가 주기억장치에 적재될 때마다 그때의 시간을 기억시켜 가장 먼저 들어와서 가장 오래 있었던 페이지를 교체하는 기법이므로, 참조 페이지 5를 참조할 때에는 2를 제거한 후 5를 가져오게 됩니다. 그러므로 모든 페이지 요청을 처리하고 나면 총 페이지 결함 발생 수는 9번입니다.

45 자원 보호 기법이 아닌 것은 권한 행렬(Capability Matrix)입니다. 자원 보호 기법에는 접근 제어 행렬(Access Control Matrix), 전역 테이블(Global Table), 접근 제어 리스트(Access Control List), 권한 리스트(Capability List), 록-키(Lock-Key)가 있습니다. 자원 보호 기법에 무엇이 있는지 구분할 수 있어야 합니다.

46 제어 프로그램 중 문제의 지문에 설명된 프로그램은 데이터 관리 프로그램(Data Management Program)입니다. 운영 체제는 제어 프로그램과 처리 프로그램으로 분류하며, 제어 프로그램은 다시 감시 프로그램, 작업 제어 프로그램, 데이터 관리 프로그램으로 분류합니다. 제어 프로그램의 종류별 특징을 읽어보고, 정리하세요.

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

47 UNIX 시스템에서 I-node에 포함되는 정보에는 파일 소유자의 사용자 번호 및 그룹 번호, 파일 크기, 파일 유형, 생성 시기, 최종 변경 시기, 최근 사용 시기, 파일의 보호 권한, 파일 링크 수, 데이터가 저장된 블록의 시작 주소 등이 있음

니다. 파일이 가장 처음 변경된 시간은 유닉스의 I-node에 포함되지 않습니다. I-node에 포함되지 않는 것을 구별해낼 수 있어야 합니다.

- 48 교착 상태 예방 기법은 교착 상태가 발생하지 않도록 사전에 시스템을 제어하는 방법으로, 교착 상태 발생의 4가지 조건 중에서 어느 하나를 제거(부정)함으로써 수행됩니다. 그러나 상호 배제 조건의 부정은 실제로 구현하지 않습니다. 그 이유는 공유 자원은 어느 한 시점에서 하나의 프로세스만 사용할 수 있어야 하기 때문입니다. 교착 상태 예방 기법 4가지를 기억하세요.

- **상호 배제(Mutual Exclusion) 부정** : 한 번에 여러 개의 프로세스가 공유 자원을 사용할 수 있도록 함
- **점유 및 대기(Hold and Wait) 부정** : 프로세스가 실행되기 전 필요한 모든 자원을 할당하여 프로세스 대기를 없애거나 자원이 점유되지 않은 상태에서 자원 요구하도록 함
- **비선점(Non-preemption) 부정** : 자원을 점유하고 있는 프로세스가 다른 자원을 요구할 때 점유하고 있는 자원을 반납하고, 요구한 자원을 사용하기 위해 기다리게 함
- **환형대기(Circular Wait) 부정** : 자원을 선형 순서로 분류하여 고유 번호를 할당하고, 각 프로세스는 현재 점유한 자원의 고유 번호보다 앞이나 뒤 어느 한쪽 방향으로만 자원을 요구하도록 하는 것

- 49 SCAN 스케줄링은 현재 헤드의 위치에서 진행 방향이 결정되면 탐색 거리가 짧은 순서에 따라 그 방향의 모든 요청을 서비스하고, 끝까지 이동한 후 역방향의 요청 사항을 서비스합니다. 그러므로 이동 순서는 70 → 60 → 50 → 20 → 80 → 100 순입니다.

- 50 SRT 스케줄링은 이미 할당된 CPU를 다른 프로세스가 강제로 빼앗아 사용할 수 있는 선점형 스케줄링 기법입니다. 나머지 보기를 통해 SJF와 SRT 기법의 특징을 정리하고 넘어가세요.

- 51 스래싱은 프로세스의 처리 시간보다 페이지 교체 시간이 더 많아지는 현상입니다. 다중 프로그래밍의 정도가 높아지면 어느 시점까지는 CPU의 이용율이 높아지고, 스래싱의 발생 빈도가 낮아지지만 어느 시점을 넘어서면 스래싱의 발생 빈도가 높아져 CPU의 이용율이 낮아집니다. 다중 프로그래밍의 정도가 높다고 무조건 좋은 것은 아닙니다. 문제의 보기를 통해 스래싱 현상을 해결하는 방법에는 무엇이 있는지 정도만 알아두세요.

- 52 지문에 제시된 내용은 다중 접근 버스 연결(Multi Access Bus Connected)형 구조입니다. ‘공유된 하나의 버스’하면 다중 접근 버스 연결 구조입니다. 위상에 대한 문제는 제시된 내용으로 머릿속에 그림을 그리면 쉽게 구분할 수 있습니다. 분산 운영체제의 각 구조를 구분하여 꼭 기억해 두세요.

- **완전 연결(Fully Connection)형** : 각 사이트들이 시스템 내의 다른 모든 사이트들과 직접 연결된 구조로, 사이트의 수가 n 개이면 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

- 53 운영체제의 여러 형태 중 시대적으로 가장 먼저 생겨난 형태는 일괄 처리 시스템입니다. 운영체제 운용 기법의 발달 과정은 ‘일괄 처리 시스템 → 다중 프로그래밍 시스템, 다중 처리 시스템, 시분할 시스템, 실시간 처리 시스템 → 다중 모드 → 분산 처리 시스템’ 순입니다.

- 54 세그멘테이션(Segmentation) 기법에서는 세그먼트 맵 테이블(Segment Map Table)이 필요합니다. 페이지 맵 테이블(Page Map Table)이 필요한 것은 페이징(Paging) 기법입니다. 세그멘테이션 기법의 특징을 정리하고 넘어가세요.

세그멘테이션 기법

- 가상기억장치에 보관되어 있는 프로그램을 다양한 크기의 논리적인 단위로 나눈 후 주기억장치에 적재시켜 실행시키는 기법이다.
- 프로그램을 배열이나 함수 등과 같은 논리적인 크기로 나눈 단위를 세그먼트라고 하며, 각 세그먼트는 고유한 이름과 크기를 갖는다.
- 기억장치의 사용자 관점을 보존하는 기억장치 관리 기법이다.
- 세그멘테이션 기법을 이용하는 궁극적인 이유는 기억 공간을 절약하기 위해서이다.
- 주소 변환을 위해서 세그먼트가 존재하는 위치 정보를 가지고 있는 세그먼트 맵 테이블(Segment Map Table)이 필요하다.
- 세그먼트가 주기억장치에 적재될 때 다른 세그먼트에게 할당된 영역을 침범할 수 없으며, 이를 위해 기억장치 보호 키(Storage Protection Key)가 필요하다.

- 55 주/중 처리기 시스템에서는 주프로세서는 입·출력과 연산을 담당하며, 운영체제를 수행합니다. 종프로세서는 주프로세서의 지시에 따라 주로 연산 위주의 작업을 처리합니다. 주/중(Master/Slave) 처리기 시스템에서의 주/중 프로세서의 역할을 구분할 수 있어야 합니다.

Master/Slave(주/중) 처리기의 특징

- 하나의 프로세서를 Master(주프로세서)로 지정하고, 나머지들은 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 발생시 주프로세서에게 서비스를 요청함

- 56 스레드(Thread)는 시스템의 여러 자원을 할당받아 실행하는 프로그램의 단위로, 하나의 프로세스에 여러 개의 스레드가 존재합니다. 자주 출제되는 문제는 아닙니다. 문제를 통해 스레드의 특징을 간단히 정리하세요.

- 57 운영체제의 성능 평가 기준에는 처리 능력(Throughput), 반환 시간(Turn Around Time), 사용 가능성(Availability), 신뢰도(Reliability)가 있습니다. 4가지 평가 기준과 각 기준

의 의미도 함께 기억해 두세요.

운영체제 성능 평가 기준

- **처리 능력(Throughput)** : 일정 시간 내에 시스템이 처리하는 일의 양
- **반환 시간(Turn Around Time)** : 시스템에 작업을 의뢰한 시간부터 처리가 완료될 때까지 걸린 시간
- **사용 가능도(Availability)** : 시스템을 사용할 필요가 있을 때 즉시 사용 가능한 정도
- **신뢰도(Reliability)** : 시스템에 주어진 문제를 정확하게 해결하는 정도

58 getppid는 부모 프로세스의 id를 얻는 명령어입니다. UNIX 명령어의 기능을 묻는 문제는 자주 출제됩니다. 보기에 제시된 각 명령어의 기능을 꼭 숙지하세요.

59 버퍼링은 일반적으로 단일 작업, 단일 사용자이지만 스펀링은 다중 작업, 다중 사용자이다. 문제에 제시된 내용을 근거로 버퍼링과 스펀링의 차이점을 구분해 두세요.

60 사이클이 허용되고, 불필요한 파일 제거를 위해 참조 카운터가 필요한 디렉터리 구조는 일반 그래프 디렉터리입니다. 디렉터리 구조에서는 각 디렉터리 구조를 구분할 수 있을 정도로만 알아두면 됩니다.

디렉터리 구조

- **1단계 디렉터리** : 가장 간단하고, 모든 파일이 하나의 디렉터리 내에 위치하여 관리되는 구조
- **2단계 디렉터리** : 중앙에 마스터 파일 디렉터리가 있고, 그 아래에 사용자별로 서로 다른 파일 디렉터리가 있는 2계층 구조
- **트리 구조 디렉터리** : 하나의 루트 디렉터리와 여러 개의 종속 디렉터리로 구성된 구조
- **비순환 그래프 디렉터리** : 하위 파일이나 하위 디렉터리를 공동으로 사용할 수 있는 것으로, 사이클이 허용되지 않는 구조
- **일반 그래프 디렉터리** : 트리 구조에 링크를 첨가시켜 순환을 허용하는 그래프 구조

61 S/W Project 일정이 지연된다고 해서 Project 말기에 새로운 인원을 추가 투입하면 Project는 더욱 지연되게 된다는 법칙은 브룩스(Brooks) 법칙입니다. 브룩스(Brooks) 법칙에 관해서는 개념을 묻는 것 한 가지만 계속 출제되고 있습니다. 개념만 정확히 파악합니다.

62 프로토타입 모형의 가장 큰 장점은 의뢰자의 요구사항을 충실히 반영할 수 있다는 것입니다. 요구사항을 충실히 반영할 수 있는 것은 프로토타입 모형이라는 것을 기억하세요. 프로토타입 모형과 함께 폭포수 모형의 특징도 알아두세요.

폭포수 모형(Waterfall Model)

- 폭포에서 한번 떨어진 물은 거슬러 올라갈 수 없듯이 소프트웨어 개발도 각 단계를 확실히 매듭짓고 그 결과를 철저하게 검토하여 승인 과정을 거친 후에 다음 단계를 진행하며 이전 단계로 넘어갈 수 없는 방식이다.
- 폭포수 모형은 소프트웨어 공학에서 가장 오래되고 가장 폭넓게 사용된 전통적인 소프트웨어 생명 주기 모형으로, 고전적 생명 주기 모형이라고도 한다.
- 소프트웨어 개발 과정의 앞 단계가 끝나야만 다음 단계로 넘어갈 수 있는 선형 순차적 모형이다.
- 제품의 일부가 될 매뉴얼을 작성해야 한다.
- 다음 단계를 수행하기 위해 각 단계가 끝난 후에는 결과물이 명확하게 산출되어야 한다.

■ 두 개 이상의 과정이 병행하여 수행되지 않는다.

63 사용 용이성(Usability)은 사용에 필요한 노력을 최소화하고 쉽게 사용할 수 있는 정도를 의미합니다. 전체나 일부 소프트웨어가 다른 응용 목적으로 사용될 수 있는 정도는 재사용성(Reusability)입니다. 나머지 보기로 제시된 품질 목표들의 개념을 꼭 기억하고 넘어가세요.

64 소프트웨어의 개발 영역을 결정하는 주요 요소에는 처리될 데이터와 소프트웨어에 대한 기능, 성능, 제약 조건, 인터페이스 및 신뢰도 등이 있습니다. 소프트웨어의 개발 영역을 결정하는 주요 요소만 기억하세요.

65 정형 기술 검토는 소프트웨어 기술자들에 의해 수행되는 소프트웨어 품질 보증 활동으로, 의제(회의의 주제)와 참가자의 수를 제한해야 합니다. 나머지 보기를 통해 정형 기술 검토 시 지침 사항을 숙지하세요.

66 자료 흐름도(Data Flow Diagram)는 처리(Process), 자료 흐름(Data Flow), 자료 저장소(Data Store), 단말(Terminator)로 구성됩니다. 자료 흐름도의 구성 요소를 숙지해 두세요.

자료 흐름도의 구성 요소

- **처리(Process)** : 입력된 자료를 출력으로 변환하는 것
- **자료 흐름(Data Flow)** : 발생지, 종착지, 처리 및 저장소 사이에서 자료의 흐름을 나타냄
- **자료 저장소(Data Store)** : 시스템 상의 자료 저장소
- **단말(Terminator)** : 시스템에 필요한 자료가 입력되는 발생지와 시스템에서 처리된 자료가 출력되는 종착지

67 객체지향 시스템에서 전통적 시스템의 함수(Function) 또는 프로시저(Procedure)에 해당하는 연산 기능은 메소드(Method)입니다. 각 보기의 용어를 구분할 수 있도록 개념을 읽어보고 넘어가세요.

- **메시지(Message)** : 객체들 간에 상호작용을 하는데 사용되는 수단으로, 객체의 메소드(동작, 연산)를 일으키는 외부의 요구 사항
- **모듈(Module)** : 소프트웨어 구조를 이루는 기본 단위로, 하나 또는 그 이상의 논리적 기능들을 수행하는 컴퓨터 지시어의 집합
- **패키지(Package)** : 소프트웨어 개발이나 업무처리를 지원하기 위해서 개발된 상품형 소프트웨어

68 소프트웨어 개발의 생산성에 영향을 미치는 요소에는 제품의 복잡도, 시스템의 크기, 프로그래머의 능력, 개발 기간, 팀 의사 전달 등이 있습니다. 소프트웨어 사용자의 능력과 소프트웨어 개발 생산성은 무관합니다.

69 자료 사전은 자료 흐름도에 있는 자료를 더 자세히 정의하고 기록한 것으로, 자료 사전에서 반복을 의미하는 기호는 { }입니다. 자료 사전에서 사용하는 표기 기호는 기호의 기능을 구분할 수 있을 정도로만 알아두세요.

자료 사전에서 사용되는 표기 기호

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

70 럼바우의 객체지향 분석 모델링(Modeling)에는 객체 모델링(Object Modeling), 동적 모델링(Dynamic Modeling), 기능 모델링(Functional Modeling)이 있습니다. 럼바우의 3가지 모델링을 묻는 문제가 자주 출제됩니다. 럼바우의 3가지 모델링의 종류와 기능을 기억해 두세요.

- **객체 모델링(Object Modeling)** : 정보 모델링이라고도 하며, 시스템에서 요구되는 객체를 찾아내어 속성과 연관 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

71 재사용 부품의 크기가 작을수록 재사용 확률이 높아집니다. 소프트웨어 재사용의 개념과 이점을 함께 알아두세요.

소프트웨어 재사용

- 이미 개발된 인정받은 소프트웨어의 전체 혹은 일부분을 다른 소프트웨어 개발이나 유지에 사용하는 것이다.
- 소프트웨어 개발의 품질과 생산성을 높이기 위한 방법으로, 기존에 개발한 소프트웨어와 경험, 지식 등을 새로운 소프트웨어에 적용한다.
- 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.

72 소프트웨어 생명 주기 모형 중 Boehm이 제시한 고전적 생명주기 모형으로서 선형 순차적 모델이라고 하는 것은 폭포수 모형입니다. 각 모델을 구분할 수 있도록 개념을 정리하세요.

소프트웨어의 생명 주기 모형

- **나선형 모형** : 보헴(Boehm)이 제안한 것으로, 폭포수 모형과 프로토타입 모형의 장점에 위험 분석 기능을 추가한 모형
- **폭포수(Waterfall) 모델** : 폭포에서 한번 떨어진 물은 거슬러 올라갈 수 없듯이 소프트웨어 개발도 각 단계를 확실히 매듭짓고 그 결과를 철저하게 검토하여 승인 과정을 거친 후에 다음 단계를 진행하며 이전 단계로 넘어갈 수 없는 방식
- **RAD 모델** : 소프트웨어의 구성 요소를 사용하여 매우 빠르게 선형 순차적 모델을 적용시킴으로써 빠른 개발 주기를 가지는 점진적 소프트웨어 개발 방식

73 공학적으로 좋은 소프트웨어는 소요 기억 용량을 적게 차지해야 합니다. 보기로 제시된 좋은 소프트웨어의 조건을 한번 읽어보고 다음 문제로 넘어가세요.

74 캡슐화(Encapsulation)는 데이터(속성)와 데이터를 처리하는 함수를 하나로 묶는 것을 의미합니다. 상위 클래스의 모든 속성과 연산을 하위 클래스가 물려받는 것은 상속성(Inheritance)입니다. 문제를 통해 캡슐화의 개념과 특징을 정리해 두세요.

75 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. 프로토타입 모델에 위험 분석 기능을 추가한 생명주기 모형은 나선형 모형입니다. CASE의 개념을 묻는 문제와 CASE 사용의 이점을 묻는 문제가 자주 출제됩니다. CASE 사용의 이점에 대해서도 숙지해 두세요.

CASE 사용의 이점

- 소프트웨어 개발 기간을 단축하고 개발 비용을 절감할 수 있다.
- 자동화된 기법을 통해 소프트웨어 품질이 향상된다.
- 소프트웨어의 유지보수를 간편하게 수행할 수 있다.
- 소프트웨어의 생산성이 향상되고 생산, 운용 활동을 효과적으로 관리·통제할 수 있다.
- 품질과 일관성을 효과적으로 제어할 수 있다.
- 소프트웨어 개발의 모든 단계에 걸친 표준을 확립할 수 있다.

76 화이트 박스 검사는 모듈의 원시 코드를 오픈시킨 상태에서 원시 코드의 논리적인 모든 경로를 검사하여 검사 사례를 설계하는 방법으로, 논리 흐름도, 루프 구조, 순환 복잡도 등의 오류를 찾을 수 있습니다. 자료 구조는 블랙박스 검사를 이용하여 찾을 수 있습니다. 화이트 박스 검사의 개념, 특징, 종류를 꼭 기억해 두세요.

화이트 박스 검사

- 모듈의 원시 코드를 오픈시킨 상태에서 원시 코드의 논리적인 모든 경로를 검사하여 검사 사례를 설계하는 방법이다.
- 설계된 절차에 초점을 둔 구조적 테스트로, 프로시저(절차) 설계의 제어 구조를 사용하여 검사 사례를 설계하며, 테스트 과정의 초기에 적용된다.
- 모듈 안의 작동을 직접 관찰한다.
- 원시 코드(모듈)의 모든 문장을 한 번 이상 수행함으로써 수행된다.
- 프로그램의 제어 구조에 따라 선택, 반복 등의 분기점 부분들을 수행함으로써 논리적 경로를 제어한다.
- 각 조건에서의 참과 거짓의 모든 논리적 결정이 적어도 한 번 이상 실행된다.
- 화이트 박스 테스트 기법에는 기초 경로 검사, 제어 구조 검사(조건 검사, 루프 검사, 데이터 흐름 검사) 등이 있다.

77 소프트웨어 유지보수 유형 중 유지보수 비용을 가장 많이 차지하는 것은 완전 보수(Perfective Maintenance)입니다. 유지보수 유형을 구분하는 문제가 자주 출제됩니다. 각 유지보수 유형은 반드시 구분해 두세요.

- **수정(Corrective) 보수** = **수리·교정·정정·하자 보수** : 시스템을 운영하면서 검사 단계에서 발견하지 못한 오류를 찾아 수정하는 활동
- **적응(Adaptive) 보수** = **환경 적응, 조정 보수** : 소프트웨어의 수명 기간 중에 발생하는 환경의 변화(하드웨어, 운영 체제 등)를 기존의 소프트웨어에 반영하기 위하여 수행하는 활동
- **완전화(Perfective) 보수** = **기능 개선, 기능 보수** : 소프트웨어의 본래 기능에 새로운 기능을 추가하거나 성능을 개선하기 위해 소프트웨어를 확장시키는 활동으로, 유지보수 활동 중 가장 큰 업무 및 비용을 차지하는 활동임
- **예방(Preventive) 보수** : 미래에 유지보수를 용이하게 하거나 기능을 향상시키기 위해 소프트웨어를 변경하는 활동

- 78 효과적인 프로젝트 관리를 위한 3P(3대 요소)는 사람(People), 문제(Problem), 프로세스(Process)입니다. 3P만 암기해 두세요.
- 79 논리적인 조건과 대응하는 행동을 간략히 표현할 수 있도록 하는 검사 사례 설계 기법은 화이트 박스 검사 기법입니다. 동치 분할 검사는 블랙 박스 검사 기법으로 프로그램의 입력 조건에 타당한 입력 자료와 타당하지 않은 입력 자료의 개수를 균등하게 하여 검사 사례를 정하고, 해당 입력 자료에 맞는 결과가 출력되는지 확인하는 방법입니다. 한 회에 화이트 박스 검사와 블랙 박스 검사에 대한 문제가 각각 출제되었네요. 이 문제를 통해서 블랙 박스 검사 기법의 특징을 정리해 두세요.
- 80 공학적으로 잘 작성된 소프트웨어란 품질이 좋은 소프트웨어를 의미합니다. 이와 관련 있는 보기를 찾아보세요. 상식적인 수준에서 풀어보세요.
- 81 전용 전송로를 사용하는 방식은 회선 교환 방식입니다. 회선 교환 방식의 특징을 묻는 문제가 종종 출제됩니다. 회선 교환 방식은 일단 접속이 이뤄지면 접속을 해제할 때까지 전용선처럼 사용할 수 있다는 것에 초점을 맞춰 특징을 이해하세요.
- 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 단말기 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 수행된다.
- 82 문자의 시작과 끝에 START 비트와 STOP 비트가 부가되어 전송의 시작과 끝을 알려 전송하는 방식은 비동기식 전송 방식입니다. 비동기식 전송의 특징을 묻는 문제가 종종 출제됩니다. 비동기식 전송은 Start Bit와 Stop Bit를 사용하여 동기를 맞춘다는 것을 중심으로 특징을 간단하게 정리하세요.
- 비동기식 전송**
- 한 문자를 나타내는 부호(문자 코드) 앞뒤에 Start Bit와 Stop Bit를 붙여서 Byte와 Byte를 구별하여 전송하는 방식이다.
 - 비동기식 전송은 시작 비트, 전송 문자(정보 비트), 정지 비트로 구성된 한 문자를 단위로 하여 전송하며, 오류 검출을 위한 패리티 비트(Parity Bit)를 추가하기도 한다.
 - 문자와 문자 사이의 휴지 시간(Idle Time)이 불규칙하다.
 - 한꺼번에 많은 데이터를 보내면 프레임িং 에러(Framing Error)의 가능성이 높아진다.

- 2,000bps(약 2kbps) 이하의 저속, 단거리 전송에 사용한다.
 - 동기화가 단순하고, 가격이 저렴하다.
 - 문자마다 시작과 정지를 알리기 위한 비트가 2~3Bit씩 추가되므로, 전송 효율이 떨어진다.
- 83 문제에 제시된 그림은 성형입니다. 네트워크 토폴로지별 특징을 묻는 문제가 자주 출제되니 확실히 정리하세요.
- **성형(Star)** : 중앙에 중앙 컴퓨터가 있고, 이를 중심으로 단말장치들이 연결되는 중앙 집중식의 네트워크 구성 형태
 - **링형(Ring)** : 컴퓨터와 단말장치들을 서로 이웃하는 것끼리 포인트 투 포인트(Point-to-Point) 방식으로 연결시킨 형태
 - **버스형(Bus)** : 한 개의 통신 회선에 여러 대의 단말장치가 연결되어 있는 형태
 - **계층형(Tree)** : 중앙 컴퓨터와 일정 지역의 단말장치까지는 하나의 통신 회선으로 연결시키고, 이웃하는 단말장치는 일정 지역 내에 설치된 중간 단말장치로부터 다시 연결시키는 형태
 - **망형(Mesh)** : 모든 지점의 컴퓨터와 단말장치를 서로 연결한 형태로, 노드의 연결성이 높음
- 84 OSI 7계층 구조를 최하위 계층부터 최상위 계층까지 나열한 것은 '다'번입니다. OSI 7계층의 순서를 꼭 암기하고 각 계층별 특징은 각 계층을 서로 구분할 수 있을 정도로만 정리하세요.
- 물리 계층(Physical Layer)**
- 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)**
- 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)**
- 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)**
- 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)**
- 송·수신측 간의 관련성을 유지하고 대화 제어를 담당하는 계층이다.

- 대화(회화) 구성 및 동기 제어, 데이터 교환 관리 기능을 한다.
- 송·수신측 간의 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)

- 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)

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

- 85 HDLC는 오류 제어를 위해 Go-Back-N ARQ와 선택적 재전송 ARQ를 사용합니다. 오류 제어를 위해 정지 대기 ARQ를 사용하는 프로토콜은 BSC입니다. HDLC는 가장 많이 사용되는 데이터 링크 제어 프로토콜로 자주 출제되고 있습니다. HDLC의 특징을 확실히 정리하세요.

HDLC(High-level Data Link Control)

- 비트(Bit) 위주의 프로토콜로, 각 프레임에 데이터 흐름을 제어하고 오류를 검출할 수 있는 비트 열을 삽입하여 전송한다.
- 포인트 투 포인트(Point-to-Point) 및 멀티 포인트(Multi-Point), 루프(Loop) 등 다양한 데이터 링크 형태에 동일하게 적용이 가능하다.
- 단방향, 반이중, 전이중 통신을 모두 지원하며, 동기식 전송 방식을 사용한다.
- 오류 제어를 위해 Go-Back-N과 선택적 재전송(Selective Repeat) ARQ를 사용한다.
- 흐름 제어를 위해 슬라이딩 윈도우 방식을 사용한다.
- 전송 제어상의 제한을 받지 않고 자유로이 비트 정보를 전송할 수 있다(비트 투과성).
- 전송 효율과 신뢰성이 높다.

- 86 인터넷 상에서 도메인 주소를 IP 주소로 변환하여 주는 서버를 DNS(Domain Name System)라고 합니다. 이 문제에서는 DNS 서버의 역할만 기억하고 넘어가세요.

- 87 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 꼭 암기하세요.

- 88 호스트 주소는 IP 헤더에 포함되는 정보입니다. TCP 헤더에는 순서 제어를 위한 긴급 pointer, 순서 번호와 오류 제어를 위한 체크섬 등이 포함됩니다. TCP와 IP를 구분할 수 있도록 각각의 특징을 정리하세요.

TCP/IP의 특징

TCP	<ul style="list-style-type: none"> ■ OSI 7 계층의 전송 계층에 해당 ■ 신뢰성 있는 연결형 서비스를 제공 ■ 패킷의 다중화, 순서 제어, 오류 제어, 흐름 제어 기능을 제공
IP	<ul style="list-style-type: none"> ■ OSI 7 계층의 네트워크 계층에 해당 ■ 데이터그램을 기반으로 하는 비 연결형 서비스를 제공 ■ 패킷의 분해/조립, 주소 지정, 경로 선택 기능을 제공

- 89 문제에 제시된 내용은 CSMA/CD 방식에 대한 설명입니다. CSMA/CD 방식의 특징은 세부적인 내용까지 알아야 풀 수 있는 문제도 간혹 출제되니 특징만큼은 확실히 정리해 두어야 합니다.

CSMA/CD 방식

개념

- CSMA 방식에서 충돌이 발생하는 문제점을 해소하기 위해 CSMA 방식에 충돌 검출 기능과 충돌 발생 시 재송신하는 기능을 추가했다.
- CSMA/CD 방식은 통신 회선이 사용 중이면 일정 시간 동안 대기하고, 통신 회선상에 데이터가 없을 때에만 데이터를 송신하며, 송신 중에도 전송로의 상태를 계속 감시한다.
- 송신 도중 충돌이 발생하면 송신을 중지하고, 모든 노드에 충돌을 알린 후 일정 시간이 지난 다음 데이터를 재송신한다.

특징

- 버스형 또는 성형 LAN에 가장 일반적으로 이용된다.
- 전송량이 적을 때 매우 효율적이고 신뢰성이 높다.
- 알고리즘(처리 기법)이 간단하다.
- 노드 장애가 시스템 전체에 영향을 주지 않으며, 장애 처리가 간단하다.
- 일정 길이 이하의 데이터를 송신할 경우 충돌을 검출할 수 없다.
- 전송량이 많아지면 충돌이 잦아져서 채널의 이용률이 떨어지고 전송 지연 시간이 급격히 증가한다.
- 충돌이 발생한 경우 다른 노드에서는 데이터를 전송할 수 없다.
- 충돌이 발생한 경우 지연 시간을 예측하기 어렵다.

- 90 HDLC에서 피기백킹(Piggybacking) 기법을 통해 데이터에 대한 확인 응답을 보낼 때 사용되는 프레임은 정보(Information) 프레임입니다. 처음으로 출제된 문제입니다. 문제와 답만 기억하고 넘어가세요.

- 91 HDLC는 데이터 링크 제어 프로토콜입니다. 라우팅 프로토콜에서는 주로 라우팅 프로토콜의 종류를 묻는 문제가 출제됩니다. 이 문제에서는 IP의 라우팅 프로토콜의 종류인 IGP(RIP, OSPF), EGP, BGP만 확실하게 기억하고 넘어가세요.

- 92 전송 시간을 일정한 간격의 시간 슬롯(Time Slot)으로 나누고, 이를 주기적으로 각 채널에 할당하는 다중화 방식은 동기식 시분할 다중화 방식입니다. 시분할 다중화 방식은 주로 동기식 시분할 다중화와 비동기식 시분할 다중화의 특징을 비교하는 문제가 출제됩니다. 동기식과 비동기식 시분할 다중화 방식의 특징을 구분해서 정리하세요.

동기식 시분할 다중화기

- 동기식 시분할 다중화기(STM, Synchronous TDM)는 일반적인 다중화기를 말하는 것으로, 모든 단말기에 균등한(고정된) 시간폭(Time Slot)을 제공한다.
- 전송되는 데이터의 시간폭을 정확히 맞추기 위한 동기 비트

가 더 필요하다.

- 통신 회선의 데이터 전송률이 전송 디지털 신호의 데이터 전송률을 능가할 때 사용한다.
 - 다중화기의 내부 속도와 단말기의 속도 차이를 보완해 주는 버퍼가 필요하다.
 - 전송할 데이터가 없는 경우에도 시간폭(Time Slot)이 제공되므로 효율성이 떨어진다.
- 비동기식 시분할 다중화기**
- 비동기식 시분할 다중화기(ATDM, Asynchronous TDM)는 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 전송률이 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.

- 93 전송 오류 제어 중 오류가 발생한 프레임뿐만 아니라 오류 검출 이후의 모든 프레임을 재전송하는 ARQ 방식은 Go-Back-N ARQ입니다. 자동 반복 요청(ARQ)의 의미와 종류, 각 수행 방식의 원리 등을 묻는 문제가 다양하게 출제됩니다. 자동 반복 요청(ARQ) 방식의 종류와 개별적인 기능을 확실히 정리하세요.

자동 반복 요청(ARQ)

자동 반복 요청(ARQ, Automatic Repeat reQuest)은 오류 발생 시 수신측은 오류 발생을 송신측에 통보하고, 송신측은 오류 발생 블록을 재전송하는 모든 절차를 의미한다.

정지-대기 ARQ

- 정지-대기(Stop-and-Wait) ARQ는 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 ARQ

연속(Continuous) ARQ는 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 전송

의 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 ARQ

- 적응적(Adaptive) ARQ는 전송 효율을 최대로 하기 위해서 데이터 블록의 길이를 채널의 상태에 따라 그때그때 동적으로 변경하는 방식이다.
- 전송 효율이 가장 좋다.
- 제어 회로가 매우 복잡하고 비용이 많이 들어, 현재 거의 사용되지 않고 있다.

- 94 펄스 코드 변조(PCM)는 ‘표본화 → 양자화 → 부호화 → 복호화 → 여과화’ 단계를 거쳐 변조가 수행됩니다. 단순히 펄스 코드 변조(PCM) 순서를 묻는 문제가 자주 출제됩니다. 꼭 기억해 두세요.

- 95 주파수 분할 다중화에서 인접한 채널 간의 간섭을 방지하기 위한 대역은 보호 대역(Guard Band)입니다. 주파수 분할 다중화에서는 보호 대역의 용도와 주파수 분할 다중화기의 특징을 묻는 문제가 자주 출제됩니다. 꼭 정리하고 넘어가세요.
- 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.

- 96 호스트의 물리 주소를 통해 논리 주소인 IP 주소를 얻어오기 위해 사용되는 프로토콜은 RARP입니다. 자주 출제되는 내용은 아닙니다. ARP와 RARP의 기능만 구분해서 알아두세요.

- **ARP** : 주소 분석 프로토콜이라고 하며, 호스트의 IP 주소를 호스트와 연결된 네트워크 접속 장치의 물리적 주소(MAC Address)로 바꿈
- **RARP** : ARP와 반대로 물리적 주소를 IP 주소로 변환하는 기능을 함

- 97 멀티 포인터(Multi-Point) 방식에서 보조국(Secondary Station)이 주국(Primary Station)에게 보낼 데이터를 갖고 있는지 주국에서 확인하는 방식은 폴링(Polling)입니다. 폴링이나 셀렉션의 의미를 묻는 문제가 종종 출제됩니다. 폴링과 셀렉션의 의미를 구분해서 파악해 두세요.

폴링/셀렉션(Polling/Selection)

주 컴퓨터에서 송·수신 제어권을 가지고 있는 방식이다.

- **폴링(Polling)** : 주컴퓨터에서 단말기에게 전송할 데이터가

있는지를 물어 전송할 데이터가 있다면 전송을 허가하는 방식으로, 단말기에서 주컴퓨터로 보낼 데이터가 있는 경우에 사용함

- **선택전(Selection)** : 주컴퓨터가 단말기로 전송할 데이터가 있는 경우 그 단말기가 받을 준비가 되었는가를 묻고, 준비가 되어 있다면 주컴퓨터에서 단말기로 데이터를 전송하는 방식

- 98 실시간 전송이나, 다량의 데이터 전송에 가장 적합한 것은 회선 교환 방식입니다. 자주 출제되는 문제입니다. 패킷 교환 방식의 특징을 확실히 정리해 두세요.

패킷 교환 방식

패킷 교환 방식(Packet Switching)은 메시지를 일정한 길이의 패킷으로 잘라서 전송하는 방식이다.

- 패킷은 장애 발생시의 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.

- 99 Bps는 '변조 속도(Baud) × 변조 시 상태 변화 수'이므로 Bps는 $400 \times 4 = 1,600\text{Bps}$ 입니다. 신호 속도(Bps)를 계산하는 문제가 자주 출제됩니다. 신호 속도(Bps) 계산 공식과 상태 변화 비트 수의 종류를 확실히 숙지하세요.

- **데이터 신호 속도(bps)** = 변조 속도(baud) × 변조 시 상태 변화 수
- **변조 속도(baud)** = 데이터 신호 속도(bps) / 변조 시 상태 변화 수
- **변조 시 상태 변화 수**
 - 모노비트(Monobit) : 1비트
 - 디비트(Dibit) : 2비트
 - 트리비트(Tribit) : 3비트
 - 쿼드비트(Quadbit) : 4비트

- 100 아날로그 데이터를 디지털 신호로 변환할 때는 펄스 코드 변조(PCM) 방식이 사용되며, 이는 코덱(CODEC)에 의해 수행됩니다. 자주 출제되는 문제는 아닙니다. 아날로그 데이터를 디지털 신호로 변환할 때 코덱과 PCM 방식이 사용된다는 것만 기억하세요.