



시험에 나오는 것만 공부한다!

**시나공시리즈**

기출문제 & 정답 및 해설  
2019년 3회 정보처리산업기사 필기 B형



## 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(①, ②, ③, ④)에 표기하시오.

### 제1과목 데이터베이스

#### 1. 개념 스키마에 대한 설명으로 옳지 않은 것은?

- ① 조직이나 기관의 총괄적 입장에서 본 데이터베이스의 전체적인 논리적 구조이다.
- ② 실제 데이터베이스가 기억장치 내에 저장되어 있으므로 저장 스키마라고도 한다.
- ③ 모든 응용 프로그램이나 사용자들이 필요로 하는 데이터를 종합한 조직 전체의 데이터베이스 구조이다.
- ④ 데이터베이스 파일에 저장되는 데이터의 형태를 나타낸 것으로 단순히 스키마라고도 한다.

#### 2. 데이터베이스 관리자(DBA)의 역할로 거리가 먼 것은?

- ① 데이터베이스의 스키마 정의
- ② 사용자 통제 및 감시
- ③ 자료의 보안성, 무결성 유지
- ④ 백업 및 회복 전략 정의

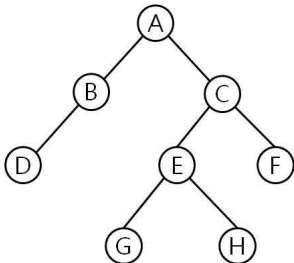
#### 3. 트리 구조에서 각 노드에서 파생된 직계 노드의 수를 의미하는 것은?

- ① Terminal Node                      ② Domain
- ③ Attribute                              ④ Degree

#### 4. E-R 모델에서 사각형은 무엇을 의미하는가?

- ① 관계 타입                              ② 개체 타입
- ③ 속성                                      ④ 링크

#### 5. 다음 트리에 대한 운행 결과의 순서가 "D → B → A → G → E → H → C → F" 일 경우, 적용된 운행 기법은?



- ① Post-Order                              ② In-Order
- ③ Pre-Order                              ④ Last-Order

#### 6. 릴레이션의 기본 키를 구성하는 어떠한 속성 값도 널(Null) 값이나 중복 값을 가질 수 없다는 것을 의미하는 것은?

- ① 참조 무결성 제약 조건              ② 주소 무결성 제약 조건
- ③ 원자값 무결성 제약 조건              ④ 개체 무결성 제약 조건

#### 7. 다음 릴레이션의 차수(Degree)는?

학번	이름	학년	학과
100	강감찬	3	전기
200	홍길동	4	전자
300	이순신	2	전산

- ① 2    ② 3
- ③ 4    ④ 9

#### 8. 관계 데이터 모델에서 애트리뷰트가 취할 수 있는 값들의 집합을 의미하는 것은?

- ① 릴레이션                                  ② 도메인
- ③ 튜플                                        ④ 차수

#### 9. 다음 관계 대수의 의미로 가장 타당한 것은?

π이름 (σ학과 = "물리학과" (학생))

- ① 이름, 학과, 물리학과를 속성으로 하는 전공 테이블 생성
- ② 학생 테이블에서 물리학과인 학생 이름 삭제
- ③ 학생 테이블에서 물리학과인 학생 이름 조회
- ④ 전공 테이블에서 학과의 이름을 물리학으로 변경

#### 10. 다음 SQL 문에서 DISTINCT의 의미는?

"SELECT DISTINCT DEPT FROM STUDENT;"

- ① 검색 결과에서 레코드의 중복 제거
- ② 모든 레코드 검색
- ③ 검색 결과를 순서대로 정렬
- ④ DEPT의 처음 레코드만 검색

#### 11. SQL에서 SELECT 문에 나타낼 수 없는 절은?

- ① HAVING                                      ② GROUP BY
- ③ DROP                                        ④ ORDER BY

#### 12. 릴레이션의 특징이 아닌 것은?

- ① 하나의 릴레이션에서 튜플의 순서는 있다.
- ② 모든 튜플은 서로 다른 값을 갖는다.
- ③ 각 속성은 릴레이션 내에서 유일한 이름을 가진다.
- ④ 모든 속성 값은 원자 값이다.

#### 13. 데이터베이스를 구성하는 데이터 개체, 이들 개체 사이의 속성, 이들 간에 존재하는 관계, 데이터 구조와 데이터 값들이 갖는 제약 조건에 관한 정의를 총칭해서 무엇이라고 하는가?

- ① VIEW                                        ② DOMAIN
- ③ SCHEMA                                      ④ DBA

#### 14. STUDENT 릴레이션에 대한 SELECT 권한을 모든 사용자에게 허가하는 SQL 명령문은?

- ① GRANT SELECT FROM STUDENT TO PUBLIC;
- ② GRANT SELECT ON STUDENT TO PUBLIC;

- ③ GRANT SELECT FROM STUDENT TO ALL;  
④ GRANT SELECT ON STUDENT TO ALL;

15. In computing, this is the process of rearranging an initially unordered sequence of records until they are ordered. What is this?

- ① Debugging                      ② Loading  
③ Sorting                        ④ Compiling

16. 널 값(Null Value)에 대한 설명으로 옳지 않은 것은?

- ① 공백(Space)과는 다른 의미이다.  
② 아직 알려지지 않은 모르는 값이다.  
③ 영(Zero)과 같은 값이다.  
④ 정보의 부재를 나타낼 때 사용하는 특수한 데이터 값이다.

17. 스택에 데이터를 A, B, C, D 순으로 저장했을 경우, 이들 데이터가 출력되는 결과로 가능한 것은?

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

18. 데이터 삽입, 삭제가 Top이라고 부르는 한쪽 끝에서만 이루어지는 후입선출(LIFO) 형태의 자료 구조는?

- ① 스택                      ② 큐                      ③ 데크                      ④ 원형 큐

19. VIEW의 삭제시 사용되는 SQL 명령은?

- ① NULL VIEW ~                      ② KILL VIEW ~  
③ DELETE VIEW ~                      ④ DROP VIEW ~

20. 다음 자료에 대하여 선택(Selection) 정렬을 사용하여 오름차순으로 정렬하고자 할 경우 1회전 후의 결과로 옳은 것은?

- “8, 3, 4, 9, 7”
- ① 3, 4, 8, 7, 9                      ② 3, 8, 4, 9, 7  
③ 3, 4, 9, 7, 8                      ④ 7, 9, 4, 3, 8

## 제2과목 전자계산기구조

21. 데이터 체인(Daisy Chain) 방식과 폴링(Polling) 방식의 설명으로 옳지 않은 것은?

- ① 폴링 방식은 소프트웨어 방식이다.  
② 데이터 체인 방식은 하드웨어 방식이다.  
③ 데이터 체인 방식이 폴링 방식보다 속도가 빠르다.  
④ 폴링 방식이 데이터 체인 방식보다 속도가 빠르다.

22. 폰 노이만(Von Neumann)형 컴퓨터 인스트럭션의 기능에 포함되지 않는 것은?

- ① 전달 기능                      ② 제어 기능  
③ 보존 기능                      ④ 함수 연산 기능

23. 주기억장치에서 접근 시간(Access Time)을 가장 옳게 설명한 것은?

- ① 판독 신호(Read Signal)를 발생한 후 자료를 메모리 주소 레지스터에 옮기기까지의 시간  
② 판독 신호 발생 후 자료를 메모리 버퍼 레지스터에 옮기기까지의 시간  
③ 메모리 주소 레지스터의 내용을 메모리 버퍼 레지스터에 옮기기까지의 시간  
④ 판독 신호를 발생한 후 다음 판독 신호가 발생할 때까지의 시간

24. 기억장치를 여러 모듈로 나누고, 한 번지(Address) 액세스 시에 다음에 사용할 번지를 미리 액세스하여 처리 속도를 향상시키는 접근 방법은?

- ① 인터리빙                      ② 페이징  
③ 세그먼팅                      ④ 스테이징

25. 디스크에서 CAV 방식에 의한 단점에 해당하는 것은?

- ① 저장 공간의 낭비                      ② 처리 속도의 저하  
③ 다수의 단말 장치 필요                      ④ 제한적 오류 검출

26. 다음 마이크로 오퍼레이션과 관계있는 것은?

MAR ← MBR (ADDR)  
MBR ← M (MAR)  
AC ← AC + MBR

- ① AND                      ② BSA  
③ ADD                      ④ JMP

27. 논리 마이크로 연산을 수행하기 위하여 다음과 같은 식이 주어졌을 때 옳지 않은 것은?

$P + Q : R1 \leftarrow R2 \vee R3$

- ① P가 1이면 R1의 내용은 변할 수도 있다.  
② P 또는 Q가 1이면 데이터 전송이 일어난다.  
③ “V”는 논리 마이크로 연산 OR를 나타낸다.  
④ “+”는 덧셈 마이크로 연산을 나타낸다.

28. 간접 주소지정 방식에서 명령어 ADD(47)이 수행되면 다음 중 어느 것이 연산장치로 보내지는가? (단, 기억장소 47번지에는 2002가 저장되어 있다.)

- ① 2002                      ② 2002번지의 내용  
③ 47                      ④ 47번지의 내용

29. 동기 가변식 마이크로 사이클에 관한 설명으로 틀린 것은?

- ① CPU의 시간을 효율적으로 이용할 수 있다.  
② 마이크로 오퍼레이션 수행시간이 현저한 차이를 나타낼 때 사용한다.  
③ 제어기의 구현이 단순하다.  
④ 그룹화된 각 마이크로 오퍼레이션들에 대하여 서로 다른 사이클 시간을 정의한다.

30. 함수 연산 기능 인스트럭션의 수행에 필요한 피연산자를 기억시킬 레지스터의 종류에 따라 컴퓨터 구조를 분류할 때, 이에 속하지 않은 것은?

- ① 스택 컴퓨터 구조  
② AC 컴퓨터 구조  
③ 리스트 컴퓨터 구조  
④ 범용 레지스터 컴퓨터 구조

31. 다음 중 특정 비트를 1로 설정하기 위해서 사용되는 논리 게이트는?

- ① NOT                      ② OR  
③ AND                      ④ EX-OR

32. 다음 중 조합논리회로가 아닌 것은?

- ① 감산기                      ② 디코더  
③ 카운터                      ④ 디멀티플렉서

33. 컴퓨터의 메모리 용량이 64K×32Bit라 하면 MAR(Memory Address Register)과 MBR(Memory Buffer Register)는 각각 몇 비트인가?

- ① MAR : 16, MBR : 16                      ② MAR : 32, MBR : 16  
③ MAR : 8, MBR : 16                      ④ MAR : 16, MBR : 32

34. 주기억장치의 용량이 512KB인 컴퓨터에서 32비트의 가상 주소를 사용하는데, 페이지의 크기가 1K워드이고 1워드가 4바이트라면 주기억장치의 페이지 수는?

- ① 32개                                      ② 64개  
③ 128개                                    ④ 512개

35. Branch 혹은 Jump 명령문은 어느 Register를 수정하는가?

- ① Accumulator  
② MAR(Memory Address Register)  
③ MBR(Memory Buffer Register)  
④ PC(Program Counter)

36. 단항(Unary) 연산을 행하는 것은?

- ① SHIFT                                    ② AND  
③ OR                                        ④ 사칙 연산

37. 키보드(Keyboard)의 키를 눌렀을 때 발생하는 인터럽트의 종류는?

- ① 외부적 인터럽트(External Interrupt)  
② 내부적 인터럽트(Internal Interrupt)  
③ 트랩(Trap)  
④ 소프트웨어 인터럽트(Software Interrupt)

38. 연관 기억장치(Associative Memory)에 대한 설명 중 옳지 않은 것은?

- ① 데이터의 내용에 액세스 되는 메모리 장치이다.  
② 메모리의 각 셀(Cell)은 저장 능력뿐만 아니라 외부의 인수(Argument)와 내용을 비교하기 위한 논리 회로를 갖고 있다.  
③ 데이터를 병렬 검색하는데 알맞게 되어 있으며, 데이터의 검색은 전체 워드를 가지고 시행된다.  
④ 검색 시간이 중요하고, 매우 짧아야 하는 특수한 경우에만 사용된다.

39. 컴퓨터 실행 중 특수한 상태가 발생할 때 제어장치의 조정에 의해 특수한 상태를 처리한 후 먼저 수행하는 프로그램으로 되돌아가는 조작은?

- ① Interrupt                                ② Controlling  
③ Trapping                                ④ Deadlock

40. 어떤 제어 기억장치의 단어 길이가 32비트, 마이크로명령어 형식의 연산필드는 12비트, 조건을 결정하는 플래그의 수는 4개일 때, 제어 기억장치의 최대 용량은 약 몇 Byte인가? (단, 분기 필드는 필요하지 않다고 가정한다.)

- ① 8.4MB                                    ② 4.2MB  
③ 2.4MB                                    ④ 1.1MB

### 제3과목 시스템분석설계

41. HIPO의 3가지 패키지가 아닌 것은?

- ① 도식 목차(Visual Table of Contents)  
② 순서도(Flowchart)  
③ 총괄 도표(Overview Diagram)  
④ 상세 도표(Detail Diagram)

42. 객체 지향 기법에서 데이터와 데이터를 조작하는 연산을 하나로 묶어 하나의 모듈 내에서 결합되도록 하는 것은?

- ① 객체                                        ② 캡슐화  
③ 다형성                                    ④ 추상화

43. 시스템에 대한 정의로 옳지 않은 것은?

- ① 예정된 기능을 수행하기 위하여 설계된 상호작용을 갖는 요소의 유기적 집합체이다.  
② 어떤 목적을 위해 하나 이상의 기능요소가 상호 관련하여 유기적으로 결합된 것이다.  
③ 공통된 목적을 위해 기여할 수 있는 많은 부분으로 구성되는 복잡한 단일체이다.  
④ 상호 관련 없는 구성 요소가 조합되어 특정 목적을 위해 독립적으로 결합된 것이다.

44. 다음에 해당하는 출력 설계 단계는?

- 출력 정보명과 사용 목적을 결정한다.
- 이용자와 이용 경로를 결정한다.
- 이용 주기 및 시기를 결정한다.
- 기밀성의 유무와 보존에 대해 결정한다.

- ① 출력 정보 내용의 설계  
② 출력 정보 이용에 대한 설계  
③ 출력 정보 매체화의 설계  
④ 출력 정보 분배에 대한 설계

45. 파일 설계 단계 중 다음 사항과 연관되는 것은?

- 갱신 빈도와 형태(추가, 삭제, 수정 등)
- 파일의 활동률
- 검색용 키 항목 개수 및 분류용 키 항목

- ① 파일 특성 조사                        ② 파일 매체 검토  
③ 파일 편성법 검토                    ④ 파일 항목 검토

46. 입력 데이터가 기록되는 디스켓, 자기테이프, 디스크, OMR 등의 규격을 결정하는 것은 어느 단계인가?

- ① 입력 매체의 설계                    ② 입력 원표의 설계  
③ 파일 구조의 설계                    ④ 처리 단계의 설계

47. 다음 중 컴퓨터 입력 단계의 체크에 해당하지 않는 것은?

- ① Unmatched Record Check  
② Batch Total Check  
③ Sequence Check  
④ Balance Check

48. 파일 편성 방법 중 순차 파일 편성 방법의 특징이 아닌 것은?

- ① 파일 내 레코드 추가, 삭제 시 파일 전체를 복사할 필요가 없다.  
② 집계용 파일이나 단순한 마스터 파일 등이 대표적인 응용파일이다.  
③ 기본 키 값에 따라 순차적으로 배열되어 있다.  
④ 기억공간의 활용률이 높다.

49. 표준 처리 패턴 중 다음 설명이 의미하는 것은?

- 마스터 파일 내의 데이터를 트랜잭션 파일로 추가, 변경, 삭제하여 항상 올바른 최근 정보를 유지한다.

- ① 갱신                                        ② 병합  
③ 정렬                                        ④ 분배

50. 파일 설계 단계에서 파일 매체 검토 시 고려사항이 아닌 것은?

- ① 파일 활동률                            ② 조작의 용이성  
③ 정보량                                    ④ 처리 시간

51. 프로세스 설계의 유의사항이 아닌 것은?

- ① 프로세스 전개의 사상을 통일한다.  
② 하드웨어의 기기 구성, 처리 성능을 고려한다.

- ③ 운영체제를 중심으로 한 소프트웨어의 효율성을 고려한다.  
 ④ 오류에 대비한 체크 시스템의 고려는 필요 없으며, 분류 처리를 가능한 최대화한다.

## 52. 다음과 같은 코드의 명칭은?

(예) 각 학과별 코드	
X X - X X	→ 대학구분 → 대학내 소속된 학과구분
01 - 01	공대 전자과
01 - 02	공대 전자계산과
02 - 01	사범대 국어교육과
02 - 02	사범대 영어교육과

- ① Block Code                      ② Decimal Code  
 ③ Sequence Code                ④ Significant Digit Code

## 53. 입출력 자료 및 코드의 설계는 다음 시스템 설계 단계의 어느 단계에서 해야 하는가?

- ① 조사 분석 단계                ② 상세 설계 단계  
 ③ 프로그램 작성 단계        ④ 실시 단계

## 54. IPT의 기법과 가장 거리가 먼 것은?

- ① 구조적 설계                    ② HIPO  
 ③ 구조적 코딩                    ④ 상향식 프로그래밍

## 55. 코드 기입 과정에서 “2006”으로 표기해야 하는데 “2060”으로 표기하였을 때의 오류는?

- ① Transcription Error        ② Transposition Error  
 ③ Addition Error                ④ Random Error

## 56. 구조적 설계에서 기능 수행 시 모듈간의 최소한의 상호작용으로 하나의 기능만을 수행하는 정도를 표현하는 용어는?

- ① 응집도                          ② 캡슐화  
 ③ 모듈화                          ④ 정보은폐

## 57. 프로세스 설계 시 유의해야 할 사항으로 가장 거리가 먼 것은?

- ① 사용자의 하드웨어와 프로그래밍에 관한 상식 수준을 고려한다.  
 ② 신뢰성과 정확성을 고려하여 처리 과정을 명확히 표현한다.  
 ③ 시스템의 상태 및 구성 요소, 기능 등을 종합적으로 표시한다.  
 ④ 오류에 대비한 체크 시스템도 고려한다.

## 58. 다음 중 시스템 분석가가 갖추어야 할 능력과 요건으로 가장 거리가 먼 것은?

- ① 기계 중심적인 분석 능력  
 ② 거시적 관점에서 세부적 요소들을 관찰할 수 있는 능력  
 ③ 사용자와 개발 요구자의 환경 이해 능력  
 ④ 서술 또는 구술 형식으로 의사소통할 수 있는 능력

## 59. 시스템의 신뢰성 평가를 위한 검토 항목으로 가장 거리가 먼 것은?

- ① 프로그램 표준화  
 ② 시스템을 구성하고 있는 각 요소의 신뢰도  
 ③ 신뢰성 향상을 위해 이미 시행한 처리에 대한 경제적 효과  
 ④ 시스템 전체의 가동률

## 60. 문서화의 목적으로 거리가 먼 것은?

- ① 시스템 보수 및 운용하는 그룹에 인계·인수 작업이 용이하다.  
 ② 개발자의 순서도 작성, 코딩, 디버깅, 테스트만을 위해서 문서화를 수행한다.  
 ③ 시스템 개발 프로젝트의 관리가 용이하다.  
 ④ 개발 진척 관리의 지표가 될 수 있다.

## 제4과목 운영체제

## 61. 스케줄링 방식에서 평균 대기 시간이 가장 짧은 것은?

- ① Round-Robin                ② SRT  
 ③ SJF                            ④ FIFO

## 62. 보안 메커니즘(Mechanism)의 설계 원칙 중 개방된 설계의 의미를 가장 잘 설명한 것은?

- ① 알고리즘은 알려졌으나, 그 키는 비밀인 암호 시스템의 사용을 의미한다.  
 ② 트로이 목마로부터의 피해를 제한하기 위해 모든 주체는 업무 완수에 필요한 최소한의 특권만을 사용해야 한다.  
 ③ 가능하다면 객체에 대한 접근은 하나 이상의 조건을 만족해야 한다.  
 ④ 가능한 한 기능 검증과 정확한 구현을 쉽게 할 수 있도록 간단히 설계한다.

## 63. 스케줄링의 목적으로 가장 옳지 않은 것은?

- ① 단위 시간당 처리량을 최대화 하기 위하여  
 ② 오버헤드를 최소화 시키기 위하여  
 ③ 응답시간과 자원의 활용간에 균형을 유지하기 위하여  
 ④ 대화식 사용자에게 가능한 빠른 응답을 주기 위하여

## 64. 교착상태가 존재하기 위한 4가지 필요 조건으로 옳지 않은 것은?

- ① 프로세스들이 필요로 하는 자원에 대해 배타적인 통제권을 요구한다.  
 ② 프로세스가 다른 자원을 기다리면서 이미 할당된 자원을 갖고 있다.  
 ③ 자원은 사용이 끝날때까지 이들이 갖고 있는 프로세스로부터 제거할 수 없다.  
 ④ 프로세스가 자원을 선점하기 위한 우선순위를 결정한다.

## 65. O/S 성능평가의 이용 가능도(Availability)를 가장 잘 설명한 것은?

- ① 동일한 시간(단위 시간) 내에서 얼마나 많은 작업량을 처리할 수 있는가의 요인  
 ② 요청한 작업에 대한 결과를 사용자에게 반환 할 때까지 소요되는 시간  
 ③ 작업의 결과를 얼마나 정확하고 믿을 수 있는가의 요인  
 ④ 시스템의 전체 운영 시간 중에서 실제 가동하여 사용 중인 시간의 비율

## 66. 파일 디스크립터에 대한 설명으로 옳지 않은 것은?

- ① 파일마다 독립적으로 존재하며, 시스템에 따라 다른 구조를 가질 수 있다.  
 ② 파일 제어 블록(FCB)이라고도 한다.  
 ③ 사용자가 관리하므로 사용자가 직접 참조할 수 있다.  
 ④ 파일 관리를 위해 시스템이 필요로 하는 정보를 가지고 있다.

## 67. 자원 보호 기법 중 접근 제어 행렬에서 수평으로 있는 각 행들만을 따온 것으로서 각 영역에 대한 권한은 객체와 그 객체에 허용된 연산자로 구성되는 것은?

- ① Global Table                ② Access Control List  
 ③ Capability List              ④ Lock/Key

## 68. 가상기억장치에 대한 설명으로 가장 거리가 먼 것은?

- ① 주기억장치 용량보다 훨씬 큰 프로그램이나 데이터를 저장할 수 있다.  
 ② 프로그램 실행 시 주소변환 작업이 필요하다.  
 ③ 가상기억장치 구현 방법으로 페이징과 세그먼테이션이 있다.



- ④ 수행중인 프로그램에서 사용된 주소가 반드시 주기억장치에  
서 사용가능한 주소이어야 한다.

**69. 프로세스 스케줄링 방법 중 시분할 시스템을 위해 고안되었으며,  
타임 슬라이스라는 작은 단위 시간이 정의되고 이 단위 시간 동안  
CPU를 제공하는 방법은?**

  - ① 선입선출
  - ② 다단계 큐
  - ③ 라운드 로빈
  - ④ 다단계 피드백 큐

**70. 병행 프로세스의 상호 배제 구현 기법으로 거리가 먼 것은?**

  - ① 데커 알고리즘
  - ② 피터슨 알고리즘
  - ③ Test And Set 명령어 기법
  - ④ 은행원 알고리즘

**71. 디스크 스케줄링 중에서 탐색 거리가 가장 짧은 요청이 먼저 서비스  
를 받는 스케줄링 기법은?**

  - ① FCFS
  - ② SCAN
  - ③ SSTF
  - ④ C-SCAN

**72. 분산 운영체제 시스템에 관한 설명으로 옳지 않은 것은?**

  - ① 약결합(Losely-Coupled)으로 볼 수 있다.
  - ② 업무량 증가에 따른 점진적인 확장이 용이하다.
  - ③ 높은 보안성이 유지된다.
  - ④ 제한된 자원을 여러 지역에서 공유 가능하다.

**73. 주기억장치에서 빈번하게 기억장소가 할당되고 반납됨에 따라  
기억장소들이 조각들로 나누어지는 현상은?**

  - ① Compaction
  - ② Fragmentation
  - ③ Coalescing
  - ④ Collision

**74. 시스템 소프트웨어로 가장 거리가 먼 것은?**

  - ① 컴파일러
  - ② 어셈블러
  - ③ 스프레드시트
  - ④ 로더

**75. 세마포어(Semaphore)에 관한 설명 중 옳지 않은 것은?**

  - ① 상호배제 문제를 해결하기 위하여 사용된다.
  - ② 정수의 변수로서 양의 값만을 가진다.
  - ③ 여러 개의 프로세스가 동시에 그 값을 수정하지 못한다.
  - ④ 세마포어에 대한 연산은 처리 도중에 인터럽트되어서는 안된다.

**76. 디스크 대기 큐에 다음과 같은 순서(왼쪽부터 먼저 도착한 순서임)  
로 트랙의 액세스 요청이 대기 중이다. 모든 트랙을 서비스하기  
위하여 FCFS 스케줄링 기법이 사용되었을 때, 모두 몇 트랙의  
헤드 이동이 생기는가?(단, 현재 헤드의 위치는 50 트랙이다.)**

디스크 대기 큐 : 10, 40, 55, 35
---------------------------

  - ① 50
  - ② 85
  - ③ 105
  - ④ 110

**77. 병렬처리의 주종(Master/Slave) 시스템에 대한 설명으로 옳지 않은  
것은?**

  - ① 주프로세서는 입출력과 연산을 수행한다.
  - ② 종프로세서는 입출력 발생시 주프로세서에게 서비스를 요청한  
다.
  - ③ 종프로세서가 운영체제를 수행한다.
  - ④ 비대칭 구조를 갖는다.

**78. UNIX 시스템에서 파일의 권한 모드 설정에 관한 명령어는?**

  - ① Chgrp
  - ② Chmod
  - ③ Chown
  - ④ Cpio

79. 기억장치의 배치(Placement) 전략에 대한 설명으로 가장 옳은 것은?
- ① 새로 반입된 프로그램을 주기억장치의 어디에 위치시킬 것인가를 결정하는 전략
  - ② 주기억장치에 넣을 다음 프로그램이나 데이터를 보조기억장치에서 주기억장치로 언제 가져올 것인가를 결정하는 전략
  - ③ 새로 주기억장치에 배치되어야 할 프로그램이 적재될 장소를 마련하기 위해 어떤 프로그램이나 데이터를 제거할 지 결정하는 전략
  - ④ 실행 중인 프로그램에 의해 참조될 프로그램이나 데이터를 미리 예상하여 적재하는 전략

80. UNIX 시스템의 계층 구조 중 가장 하드웨어와 관련이 없고 사용자와 밀접하므로 사용자의 명령을 입력으로 받아 그 명령을 해석하는 역할을 하는 계층은?
- ① 커널                      ② 셸  
③ 기억장치 관리자        ④ 스케줄러

## 제5과목 : 정보 통신 개론

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

  - ① HDLC
  - ② PPP
  - ③ LLC
  - ④ UDP

82. 데이터 전송의 형태에서 한 문자 전송 시마다 스타트 비트와 스톱 비트를 삽입하여 전송하는 방식은?

  - ① 동기식
  - ② 비동기식
  - ③ 베이스 밴드식
  - ④ 혼합동기식

83. 회선 양쪽 시스템이 처리 속도가 다를 때 데이터양이나 통신 속도를 수신 측이 처리할 수 있는 능력을 넘어서지 않도록 조정하는 기술은?

  - ① 인증 제어
  - ② 흐름 제어
  - ③ 오류 제어
  - ④ 동기화

84. 데이터링크 계층의 기능에 관한 내용으로 틀린 것은?

  - ① 인접노드 간의 흐름 제어와 에러 제어 기능을 수행한다.
  - ② 매체공유를 위한 매체 접근 제어(MAC)를 수행한다.
  - ③ 발신지에서 목적지까지 최적의 패킷 전송경로를 설정한다.
  - ④ 프레임을 노드에서 노드로 전달한다.

85. 샤논-하트레이(Shannon Hartley)의 통신채널용량(bps)은?(단, C : 채널용량, B : 채널의 대역폭, S : Signal, N : Noise)

  - ①  $C = B \log_2(1 + S/N)$
  - ②  $C = B \log_2(1 + N/S)$
  - ③  $C = 2B \log_2(1 + S \times N)$
  - ④  $C = 5B \log_2(1 + N/S)$

86. HDLC 링크구성 방식에 따른 세 가지 동작모드에 해당하지 않는 것은?

  - ① 정규 응답 모드(NRM)
  - ② 동기 응답 모드(SRM)
  - ③ 비동기 응답 모드(ARM)
  - ④ 비동기 균형 모드(ABM)

87. 9600bps의 전송 속도를 갖는 모뎀이 4개의 위상을 갖는 QPSK로 변조될 때 변조 속도(baud)는?

  - ① 4800
  - ② 2400
  - ③ 1200
  - ④ 600

88. 고속의 송신 신호를 다수의 직교하는 협대역 반송파로 다중화시키는 변조 방식으로 가장 옳은 것은?

  - ① TDM
  - ② OFDM
  - ③ FDM
  - ④ SSL

89. 오류를 제어할 때 수신측에서 오류의 검출 기능과 정정 기능을 동시에 갖는 부호는?

- ① Hamming Code                      ② Parity Code  
③ ASCII Code                        ④ EBCDIC Code

90. IPv6의 특징으로 틀린 것은?

- ① IPv6 주소의 길이는 256비트이다.  
② 암호화와 인증 옵션 기능을 제공한다.  
③ 프로토콜의 확장을 허용하도록 설계되었다.  
④ 흐름 레이블(Flow Label)이라는 항목이 추가되었다.

91. 라우팅(Routing) 프로토콜이 아닌 것은?

- ① BGP                                      ② OSPF  
③ SMTP                                    ④ RIP

92. 10 Base T 근거리 통신망의 특성을 올바르게 나타낸 것은?

- ① 10 Mbps, Baseband, Twisted Pair Cable  
② 10 Gbps, Baseband, Twisted Pair Cable  
③ 10 Gbps, Broadband, Coaxial Cable  
④ 10 Mbps, Broadband, Coaxial Cable

93. 다음이 설명하고 있는 것은?

효율적인 전송을 위하여 넓은 대역폭을 가진 하나의 전송 링크를 통하여 여러 신호를 동시에 실어 보내는 기술이다.

- ① 집중화                                    ② 다중화  
③ 복호화                                    ④ 공유화

94. PCM(Pulse Code Modulation) 방식의 구성 절차로 옳은 것은?

- ① 양자화 → 부호화 → 표본화 → 복호화  
② 표본화 → 양자화 → 부호화 → 복호화  
③ 표본화 → 부호화 → 양자화 → 복호화  
④ 양자화 → 표본화 → 복호화 → 부호화

95. OSI-7계층 중 암호화, 데이터 압축, 코드 변환 등의 기능을 수행하는 계층은?

- ① 전송 계층                                ② 응용 계층  
③ 물리 계층                                ④ 프리젠테이션 계층

96. 이동통신망에서 사용되는 다원 접속(Multiple Access) 방식이 아닌 것은?

- ① CDMA                                    ② KDMA  
③ TDMA                                    ④ FDMA

97. HDLC에서 한 프레임(Frame)을 구성하는 요소로 가장 거리가 먼 것은?

- ① Flag                                        ② Address Field  
③ Control Field                            ④ Start/Stop bit

98. LAN의 접근 방식에 따른 분류에 해당하지 않는 것은?

- ① CSMA/CD                                ② 토큰 링  
③ 토큰 버스                                ④ 캐리어 밴드

99. 동일 건물에 있는 다양한 컴퓨터 기기들을 상호 연결하여 정보통신망에 연결된 다른 기기나 주변기기들과 공유할 수 있도록 설계한 네트워크 형태(Topology)는?

- ① 패킷 교환망(PSDN)  
② 부가 가치 통신망(VAN)  
③ 근거리 통신망(LAN)  
④ 공중 전화망(PSTN)

100. M진 PSK에서 반송파 간의 위상차는? (단, M은 진수이다.)

- ①  $\pi \times M$                                       ②  $\frac{2\pi}{3M}$   
③  $\frac{\sqrt{\pi}}{M}$     ④  $\frac{2\pi}{M}$

## 정답 및 해설

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

- 1 개념 스키마는 데이터베이스의 전체적인 논리적 구조를 의미하고 그 구조에 따라 데이터가 저장되지만 개념 스키마를 저장 스키마라고 부르지는 않습니다.

## [전문가의 조언]

스키마에서 출제되는 문제 대부분이 스키마의 3계층을 구분할 수 있는가에 대해 묻고 있습니다. 어떤 경우에도 3계층을 구분할 수 있을 정도로 각각의 개념을 명확히 하세요.

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

- 외부 스키마는 사용자나 응용 프로그래머가 각 개인의 입장에서 필요로 하는 데이터베이스의 논리적 구조를 정의한 것이다.
- 외부 스키마는 전체 데이터베이스의 한 논리적인 부분으로 볼 수 있으므로 서브 스키마(Sub Schema)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 개체간의 관계와 제약 조건을 나타내고 데이터베이스의 접근 권한, 보안 및 무결성 규칙에 관한 명세를 정의한다.
- 단순히 스키마(Schema)라고 하면 개념 스키마를 의미한다.
- 기관이나 조직체의 관점에서 데이터베이스를 정의한 것이다.
- 데이터베이스 관리자(DBA)에 의해서 구성된다.

**내부 스키마(Internal Schema)**

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

- 2 데이터베이스 관리자(DBA)는 데이터베이스 시스템의 모든 관리와 운영에 대한 책임을 지고 있는 것이지 사용자의 통제 및 감시 등은 수행하지 않습니다.

## [전문가의 조언]

DBA의 역할을 묻는 문제가 많이 출제됩니다. DBA의 역할을 모두 기억하기보다는 객관식 문제의 특성을 살려 DBA가 하는 일이 아닌 것을 찾아낼 수 있게 공부하는 것이 바람직합니다.

**데이터베이스 사용자**

**DBA(DataBase Administrator)**

- 데이터베이스 시스템의 모든 관리와 운영에 대한 책임을 지고 있는 사람이나 그룹으로 다음과 같은 역할을 한다.
- 데이터베이스 설계와 조작에 대한 책임
  - 데이터베이스 구성 요소 결정
  - 개념 스키마 및 내부 스키마 정의
  - 데이터베이스의 저장 구조 및 접근 방법 정의
  - 보안 및 데이터베이스의 접근 권한 부여 정책 수립
  - 장애에 대비한 예비(Back Up) 조치와 회복(Recovery)에 대한 전략 수립
  - 무결성을 위한 제약 조건의 지정
  - 데이터 사전의 구성과 유지 관리
  - 사용자의 변화 요구와 성능 향상을 위한 데이터베이스의 재구성
- 행정 책임
  - 사용자의 요구와 불평의 청취 및 해결
  - 데이터 표현 방법의 표준화
  - 문서화에 대한 기준 설정
- 시스템 감시 및 성능 분석
  - 변화 요구에 대한 적응과 성능 향상에 대한 감시
  - 시스템 감시 및 성능 분석
  - 자원의 사용도와 병목 현상 조사
  - 데이터 사용 추세, 이용 형태 및 각종 통계 등의 종합, 분석

**응용 프로그래머**

- 응용 프로그래머는 일반 호스트 언어로 프로그램을 작성할 때 데이터 조작어를 삽입해서 일반 사용자가 응용 프로그램을 사용할 수 있게, 인터페이스를 제공할 목적으로 데이터베이스를 접근하는 사람들이다.
- 응용 프로그래머는 C, COBOL, PASCAL 등의 호스트 언어와 DBMS가 지원하는 데이터 조작어에 능숙한 컴퓨터 전문가이다.

**일반 사용자**

일반 사용자는 보통 터미널을 이용하여 데이터베이스에 있

는 자원을 활용할 목적으로 질의어나 응용 프로그램을 사용하여 데이터베이스에 접근하는 사람들이다.

- 3 트리 구조에서 각 노드가 가진 가지 수, 즉 서브 트리의 수를 그 노드의 디그리(차수)라 하고, 가장 큰 디그리(차수)를 트리의 디그리(차수)라고 합니다.

**[전문가의 조언]**

문제와 보기가 동일하게 다시 출제되었습니다. 트리의 차수와 노드의 차수를 구분해서 기억해 두세요.

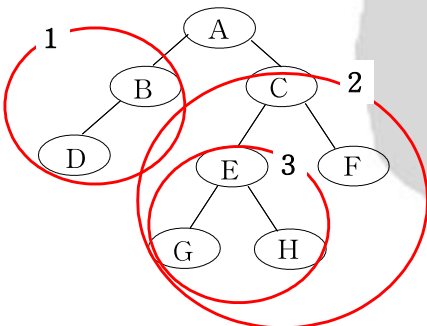
4 **[전문가의 조언]**

**중요해요!** E-R 모델과 관련해서는 E-R 다이어그램에서 사용하는 도형이 가장 많이 출제됩니다. 이 문제에서 확실하게 기억하고 넘어가세요.

**E-R 도형**

다이아몬드(마름모)	관계(Relationship) 타입
사각형	개체 집합
타원	속성(Attribute)
밑줄 타원	기본 키 속성
선, 링크	개체 타입과 속성을 연결

- 5 문제의 트리를 'D → B → A → G → E → H → C → F' 순서대로 방문하는 순회 기법은 Inorder 순회 기법입니다. 문제의 트리를 Inorder로 순회해 보겠습니다. 먼저 서브 트리를 하나의 노드로 생각할 수 있도록 서브 트리 단위로 묶습니다.



- ① Inorder는 Left → Root → Right 이므로 1A2가 됩니다.  
 ② 1은 DB이므로 DBA2가 됩니다.  
 ③ 2는 3CF이므로 DBA3CF가 됩니다.  
 ④ 3은 GEH이므로 DBAGEHCF가 됩니다.

**[전문가의 조언]**

**중요해요!** 이진 트리를 탐색하는 순회법의 이름은 Root의 위치가 어디 있느냐에 따라 정해집니다. 즉 Root가 앞(Pre)에 있으면 Preorder, 안(In)에 있으면 Inorder, 뒤(Post)에 있으면 Postorder입니다. 이 기준으로 문제에 제시된 트리를 Preorder와 Postorder로도 순회해 보세요.

6 **[전문가의 조언]**

참조 무결성과 개체 무결성은 매회 빠지지 않고 출제되고 있습니다. 개체 무결성과 더불어 참조 무결성의 개념도 정리해 두세요.

**참조 무결성**

외래키값은 NULL이거나 참조 릴레이션의 기본키 값과 동일해야 한다. 즉 릴레이션은 참조할 수 없는 외래키값을 가질 수 없다.

- 7 차수(Degree)는 속성의 수, 즉 열의 수를 말합니다.

**[전문가의 조언]**

관계형 데이터베이스의 릴레이션을 구성하는 요소들은 매우

중요합니다. 정확히 숙지해 두세요.

**릴레이션의 구성 요소**

<b>릴레이션 (Relation)</b>	데이터들을 표(Table)의 형태로 표현한 것으로, 구조를 나타내는 릴레이션 스키마와 실제 값들인 릴레이션 인스턴스로 구성됨
<b>인스턴스 (Instance)</b>	데이터 개체를 구성하고 있는 속성들에 데이터 타입이 정의되어 구체적인 데이터 값을 갖고 있는 것을 말함
<b>튜플 (Tuple)</b>	<ul style="list-style-type: none"> <li>릴레이션을 구성하는 각각의 행을 말함</li> <li>속성의 모임으로 구성됨</li> <li>파일 구조에서 레코드와 같은 의미임</li> <li>튜플의 수를 카디널리티(Cardinality) 또는 기수, 대응수라고 함</li> </ul>
<b>속성 (Attribute)</b>	<ul style="list-style-type: none"> <li>데이터베이스를 구성하는 가장 작은 논리적 단위</li> <li>파일 구조상의 데이터 항목 또는 데이터 필드에 해당됨</li> <li>개체의 특성을 기술함</li> <li>속성의 수를 디그리(Degree) 또는 차수라고 함</li> </ul>
<b>도메인 (Domain)</b>	<ul style="list-style-type: none"> <li>하나의 애트리뷰트가 취할 수 있는 같은 타입의 원자(Atomic)값들의 집합</li> <li>실제 애트리뷰트 값이 나타날 때 그 값의 합법 여부를 시스템이 검사하는 데에도 이용됨</li> <li>예) 성별 애트리뷰트의 도메인은 '남'과 '여'로, 그 외의 값은 입력될 수 없음</li> </ul>

8 **[전문가의 조언]**

관계형 데이터베이스에서 릴레이션을 구성하는 용어들은 중요하다고 했죠. 이 문제를 틀렸다면 7번 문제의 [전문가의 조언]을 다시 한번 공부하세요.

- 9  $\pi$ 는 열만 구하는 Project,  $\sigma$ 는 조건에 맞는 튜플만 구하는 Select입니다. 괄호 속의 연산을 먼저 해야 합니다. 그대로 해석하면, '<학생> 릴레이션에서 '학과' 속성의 값이 "물리학과"인 튜플들 중 '이름' 속성만 구하시오'입니다.

**[전문가의 조언]**

자주 출제되는 문제는 아닙니다. 문제와 답만 외우고 넘어가세요.

- 10 SQL 문에서 DISTINCT의 의미는 검색 결과에서 레코드의 중복을 제거하라는 의미로 중복된 레코드가 있으면 그 중 첫 번째 한 개만 검색하여 표시합니다.

**[전문가의 조언]**

중중 출제되고 있는 문제입니다. 이 문제에서는 DISTINCT의 의미만 확실히 기억하고 넘어가세요.

- 11 DROP은 스키마, 도메인, 기본 테이블, 뷰 테이블, 인덱스 등을 삭제하는 데이터 정의어(DDL)입니다.

**[전문가의 조언]**

**중요합니다.** SELECT문의 구조와 각 절의 의미를 숙지하세요. **SELECT 문의 일반 형식**

```
SELECT [테이블명.]속성명1, [테이블명.]속성명2, ...
FROM 테이블명1, 테이블명2, ...
[WHERE 조건]
[GROUP BY 속성명1, 속성명2, ...]
[HAVING 조건]
[ORDER BY 속성명 [ASC|DESC]];
```



### · SELECT절

- **속성명** : 검색하여 불러올 속성(열) 또는 속성을 이용한 수식을 지정한다.
  - ▶ 기본 테이블을 구성하는 모든 속성을 지정할 때는 '\*'를 기술한다.
  - ▶ 두 개 이상의 테이블을 대상으로 검색할 때는 '테이블명.속성명'으로 표현한다.
- **FROM절** : 질의에 의해 검색될 데이터들을 포함하는 테이블명을 기술한다.
- **WHERE절** : 검색할 조건을 기술한다.
- **GROUP BY절** : 특정 속성을 기준으로 그룹화하여 검색할 때 그룹화할 속성을 지정한다.
  - 일반적으로 GROUP BY절은 그룹 함수와 함께 사용된다.
- **HAVING절** : GROUP BY와 함께 사용되며, 그룹에 대한 조건을 지정한다.
- **ORDER BY절** : 특정 속성을 기준으로 정렬하여 검색할 때 사용한다.
  - **속성명** : 정렬의 기준이 되는 속성명을 기술한다.
  - **[ASC|DESC]** : 정렬 방식으로서 'ASC'는 오름차순, 'DESC'는 내림차순이다. 생략하면 오름차순으로 지정된다.

12 한 릴레이션에 포함된 튜플 사이에는 순서가 없습니다.

#### [전문가의 조언]

자주 출제되는 내용입니다. 릴레이션의 특징을 무조건 암기하지 말고 다음에 주어진 예를 <학생> 릴레이션에 적용시켜 보면서 이해하세요.

#### 릴레이션의 특징

##### <학생>

학과	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 학적사항을 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 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 속성명들과 구분되어 유일해야 하지만 '학년' 속성에는 2, 1, 2, 4 등이 입력된 것처럼 동일한 값이 있을 수 있다.
- 릴레이션을 구성하는 튜플을 유일하게 식별하기 위해 속

성들의 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.

### 13 [전문가의 조언]

스키마는 정의, 특징, 3계층 구조에 대한 문제가 자주 출제됩니다. 이 문제에서는 스키마의 정의와 특징을 정리하고 넘어가세요.

#### 스키마(Schema)의 정의

- 스키마는 데이터베이스의 구조와 제약조건에 관한 전반적인 명세(Specification)를 기술(Description)한다.
- 스키마는 데이터베이스를 구성하는 데이터 개체(Entity), 속성(Attribute), 관계(Relationship) 및 데이터 조작시 데이터 값들이 갖는 제약조건 등에 관해 전반적으로 정의한다.
- 스키마는 사용자의 관점에 따라 외부 스키마, 개념 스키마, 내부 스키마로 나뉜다.

#### 스키마의 특징

- 스키마는 데이터 사전(Data Dictionary)에 저장된다.
- 스키마는 현실 세계의 특정한 한 부분의 표현으로서 특정 데이터 모델을 이용해서 만들어진다.
- 스키마는 시간에 따라 불변인 특성을 갖는다.
- 스키마는 데이터의 구조적 특성을 의미하며, 인스턴스에 의해 규정된다.

- 14 GRANT문은 데이터베이스 사용자에게 사용 권한을 부여하는 SQL 명령문입니다. GRANT문의 형식은 'GRANT 권한 ON 객체 TO 사용자'이며, 모든 사용자에게 사용 권한을 부여하려면 사용자에게 'PUBLIC'을 입력하면 됩니다.

#### [전문가의 조언]

문제와 보기가 동일하게 다시 출제되었습니다. GRANT문의 형식을 정확히 기억해 두세요.

- 15 컴퓨터 사용에 있어서, 초기에 비순서적으로 나열된 레코드들이 순서대로 될 때까지 다시 나열하는 과정을 정렬(Sort)이라고 합니다.

#### [전문가의 조언]

동일한 문제가 출제된 적이 있습니다. 이 문제에서는 정렬(Sort)의 개념만 명확히 하고 넘어가세요.

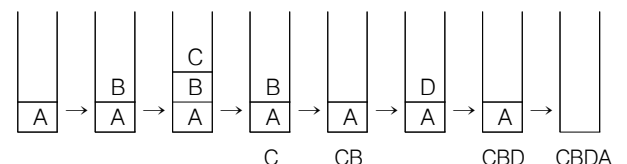
- 16 널(Null)은 이론적으로 아무것도 없는 값을 의미합니다. 0은 숫자 데이터로서 널(Null) 값이 아닙니다.

#### [전문가의 조언]

널(Null) 값의 의미와 관련된 문제가 자주 출제되고 있습니다. 이 문제를 통해 널(Null) 값의 의미를 꼭 기억하세요.

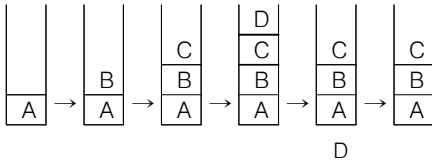
- 17 이 문제는 문제의 자료가 각 보기의 순서대로 출력되는지 스택을 이용해 직접 입·출력을 수행해 보면 됩니다. PUSH는 스택에 자료를 입력하는 명령이고, POP는 스택에서 자료를 출력하는 명령입니다. 먼저 ②번을 살펴볼게요.

PUSH A   PUSH B   PUSH C   POP C   POP B   PUSH D   POP D   POP A



반면 ①번은 D 출력 후에 B를 출력해야 하는데, C를 출력하지 않고는 D를 출력할 수 없으므로 불가능합니다.

PUSH A   PUSH B   PUSH C   PUSH D   POP D   POP B(불가능)

**[전문가의 조언]**

자주 출제되는 내용입니다. ③, ④번도 위와 같은 방법으로 스택에 자료를 넣었다 꺼내보면서 그대로 출력이 가능한지 확인해 보세요.

**18 [전문가의 조언]**

스택의 개념과 용도를 묻는 문제가 자주 출제됩니다. 확실히 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 기억 공간에 가장 마지막으로 삽입된 자료가 기억된 공간을 가리키는 요소로서, 스택 포인터(SP, Stack Pointer)라고도 함
- **Bottom** : 스택의 가장 밑바닥임
- 용도
  - 부 프로그램 호출시 복귀 주소를 저장할 때
  - 인터럽트가 발생하여 복귀 주소를 저장할 때
  - 후위 표기법(Postfix Notation)으로 표현된 산술식을 연산할 때
  - 0 주소지정방식 명령어의 자료 저장소
  - 재귀(Recursive) 프로그램의 순서 제어
  - 컴파일러를 이용한 언어번역 시

**19 [전문가의 조언]**

**중요해요!** 뷰를 삭제할 때는 DROP문을 사용한다는 것을 기억해 두세요. 그리고 뷰는 의미, 특징, 장점과 단점이 모두 자주 출제된다는 것을 유념하세요. 뷰의 의미를 충분히 이해하면 특징이나 장·단점도 쉽게 기억됩니다.

**뷰(View)의 의미**

- 뷰는 사용자에게 접근이 허용된 자료만을 제한적으로 보여주기 위해 하나 이상의 기본 테이블로부터 유도된, 이름을 가지는 가상 테이블이다.
- 뷰는 저장장치 내에 물리적으로 존재하지 않지만, 사용자에게는 있는 것처럼 간주된다.

**뷰(View)의 특징**

- 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 독립성을 어느 정도 제공한다.
- 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.
- 일단 정의된 뷰는 다른 뷰의 정의에 기초가 될 수 있다.
- 하나의 뷰를 삭제하면 그 뷰를 기초로 정의된 다른 뷰도

자동으로 삭제된다.

**뷰(View)의 장·단점**

- 장점
  - 논리적 데이터 독립성을 제공한다.
  - 동일 데이터에 대해 동시에 여러 사용자의 상이한 응용이나 요구를 지원해 준다.
  - 사용자의 데이터 관리를 간단하게 해준다.
  - 접근 제어를 통한 자동 보안이 제공된다.
- 단점
  - 독립적인 인덱스를 가질 수 없다.
  - ALTER VIEW문을 사용할 수 없다. 즉 뷰의 정의를 변경할 수 없다.
  - 뷰로 구성된 내용에 대한 삽입, 삭제, 갱신 연산에 제약이 따른다.

- 20** 선택 정렬은 n개의 레코드 중에서 최소값을 찾아 첫 번째 레코드 위치에 놓고, 나머지 n-1개 중에서 다시 최소값을 찾아 두 번째 레코드 위치에 놓는 방식을 반복하여 정렬하는 방식입니다.

• 원본 : 8, 3, 4, 9, 7

① 1회전 : 3, 8, 4, 9, 7 → 3, 8, 4, 9, 7 → 3, 8, 4, 9, 7  
→ 3, 8, 4, 9, 7

② 2회전 : 3, 4, 8, 9, 7 → 3, 4, 8, 9, 7 → 3, 4, 8, 9, 7

③ 3회전 : 3, 4, 8, 9, 7 → 3, 4, 7, 9, 8

④ 4회전 : 3, 4, 7, 8, 9

**[전문가의 조언]**

필기 시험뿐만 아니라 실기 시험에도 출제되는 내용입니다. 꼭 이해하고 넘어가세요.

- 21** 폴링 방식은 소프트웨어적인 인터럽트 판별 방식으로서 하드웨어적인 인터럽트 판별 방식인 페이지 체인 방식보다 속도가 느립니다.

**[전문가의 조언]**

데이터 체인과 폴링의 특징을 구분하는 문제가 자주 출제됩니다. 인터럽트 우선순위를 소프트웨어적으로 판별하는 것은 폴링이고, 하드웨어적으로 판별하는 것은 페이지 체인이라는 것을 기억하고 각각의 특징을 정리하세요.

**하드웨어적인 인터럽트 우선순위 부여 방식 : 페이지 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 페이지 체인 방식이라고 한다.

**소프트웨어적인 인터럽트 우선순위 판별 방법 : Polling**

- Interrupt 발생시 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 필요 없으므로 경제적이다.

- 22 폰 노이만형 컴퓨터란 명령어를 차례차례 순차적으로 처리하는 일반 컴퓨터를 말하며, 이 컴퓨터의 인스트럭션 기능에는 함수 연산 기능, 전달 기능, 입·출력 기능, 제어 기능 등이 있습니다.

#### [전문가의 조언]

폰 노이만(Von Neumann)형 컴퓨터의 연산자 기능은 함수 연산 기능, 전달 기능, 입·출력 기능, 제어 기능이 있다는 것과 각 기능에 해당하는 연산자를 서로 구분해 낼 수 있도록 정리하세요.

#### 연산자(Operation Code)의 기능

##### • 함수 연산 기능

- 산술 연산 : ADD, SUB, MUL, DIV, 산술 Shift 등
- 논리 연산 : NOT, AND, OR, XOR, 논리적 Shift, Rotate, Complement, Clear 등

##### • 자료 전달 기능 : Load, Store, Move, Push, Pop

##### • 제어 기능 : GOTO, Jump(JMP), IF 조건, SPA, SNA, SZA, Call, Return

##### • 입·출력 기능 : INPUT, OUTPUT

- 23 [전문가의 조언]

종종 출제되는 내용입니다. 기억장치 관련 용어들을 정리해 두세요.

#### Access Time

- Access Time은 기억장치에 읽기 요청이 발생한 시간부터 요구한 정보를 꺼내서 사용 가능할 때까지의 시간이다.
- Access Time은 한 Word 단위의 정보를 읽거나 기록하는데 걸리는 시간이다.
- Access Time = Seek Time + Latency Time(또는 Search Time) + Transmission Time

#### Cycle Time

- Cycle Time은 기억장치에 읽기 신호를 보낸 후 다시 읽기 신호를 보낼 수 있을 때까지의 시간 간격을 말한다.
- Cycle Time  $\geq$  Access Time, 즉 Cycle Time이 Access Time보다 길거나 같다.

#### Bandwidth(대역폭, 전송률)

- 대역폭은 메모리로부터 또는 메모리까지 1초 동안 전송되는 최대한의 정보량으로, 기억장치의 자료 처리 속도를 나타내는 단위이다.
- 메모리 워드의 길이가 작을수록 대역폭이 좋다.
- 전송 단위 : Baud(보) = Bps(1초당 전송 가능한 비트 수)

- 24 메모리 인터리빙(Interleaving)은 단위 시간에 여러 메모리의 접근이 가능하도록 병행 접근하는 기법입니다.

#### [전문가의 조언]

메모리 인터리빙에 대해서는 개념에 대한 문제가 주로 출제됩니다. 개념 및 특징을 꼭 숙지하세요.

#### 메모리 인터리빙(Memory Interleaving)

- 인터리빙이란 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 접근 시간을 효율적으로 높일 수 있으므로 캐시 기억장치, 고속 DMA 전송 등에서 많이 사용된다.

- 25 디스크에서 등각속도(CAV) 방식을 사용할 경우 트랙이 있는 위치에 따라 데이터 저장 밀도가 달라지므로 디스크 저장 공간이 비효율적으로 사용됩니다.

#### [전문가의 조언]

동일한 문제가 출제된적이 있습니다. 디스크에서 등각속도(CAV) 방식의 특징을 간단히 정리해 두세요.

#### 디스크에서 등각속도(CAV) 방식의 특징

- 모든 트랙의 저장 밀도가 다르다.
- 디스크 저장 공간이 비효율적으로 사용된다.
- 회전 구동장치가 간단하다.
- 디스크 평판이 일정한 속도로 회전한다.

- 26 메모리의 내용을 가져와서 누산기(AC)의 값과 더하는 것이므로 ADD에 해당됩니다.

#### [전문가의 조언]

이런 종류의 문제는 한 번만 정확하게 이해하면 다음부터는 쉽게 맞힐 수 있습니다. 하나씩 차분하게 생각하면 어렵지 않으니 꼭 이해하고 넘어가세요.

- $MAR \leftarrow MBR(ADDR)$  : MBR이 갖고 있는 주소를 MAR에 전송함
- $MBR \leftarrow M(MAR)$  : 기억장치의 MAR 번지에 있는 자료를 MBR로 가져옴
- $AC \leftarrow AC + MBR$  : AC에 MBR의 내용을 더하여 AC에 저장함
- ※ ADD와 LDA는 ' $AC \leftarrow AC + MBR$ ' 전에 AC를 초기화하는 ' $AC \leftarrow 0$ ' 코드가 있느냐? 없느냐?의 차이입니다. 즉 ' $AC \leftarrow 0$ ' 코드가 있으면 0과 더하는 것이므로 기억장소의 값을 그대로 AC로 전송하는 LDA(Load to Acc)이고, 없으면 AC에 들어 있는 어떤 값과 더하는 것이므로 ADD입니다.

- 27 왼쪽 명령이 실행된 것이 오른쪽의 마이크로 오퍼레이션입니다. 곧 '+'는 논리 마이크로 오퍼레이션 연산인 OR를 나타냅니다.

#### [전문가의 조언]

문제와 보기가 동일하게 다시 출제되었습니다. 보기에 제시된 내용 정도만 이해하고 넘어가세요.

- 28 오퍼랜드 값 47이 간접 주소이므로, 47번지에 있는 값 2002는 오퍼랜드로 사용할 값이 아니라 오퍼랜드로 사용할 값이 들어있는 곳을 가리키는 번지입니다. 즉 47번지에 기억된 2002가 가리키는 곳의 값이 연산장치로 보내질 값입니다.

#### [전문가의 조언]

간접 주소지정방식의 개념이나 개념을 이용한 응용 문제가 종종 출제됩니다. 간접 주소지정방식의 특징을 정리하고 넘어가세요.

#### 간접 주소지정방식(Indirect Mode)

- 명령의 길이가 짧고 제한되어 있어도 긴 주소에 접근 가능한 방식이다.
- 명령어 내의 Operand부에 실제 데이터가 저장된 장소의 번지를 가진 기억장소의 번지를 표현함으로써, 최소한 주 기억장치를 두 번 이상 접근하여 데이터가 있는 기억장소에 도달한다.
- 명령어에 나타낸 주소가 명령어 내에서 데이터를 지정하기 위해 할당된 비트(Operand 부의 비트)수로 나타낼 수 없을 때 사용하는 방식이다.



- 29 동기 가변식은 CPU 시간의 낭비를 줄일 수 있지만 제어회로가 조금 복잡합니다.

**[전문가의 조언]**

마이크로 사이클 타임 부여 방식들의 개별적인 특징을 구분하는 문제가 종종 출제됩니다. 서로 구분할 수 있도록 정리하고 넘어가세요.

**마이크로 사이클 타임(Micro Cycle Time) 부여 방식**

**동기 고정식(Synchronous Fixed)**

- 모든 마이크로 오퍼레이션의 동작시간이 같다고 가정하여 CPU Clock의 주기를 Micro Cycle Time과 같도록 정의하는 방식이다.
- 모든 마이크로 오퍼레이션 중에서 수행시간이 가장 긴 마이크로 오퍼레이션의 동작시간을 Micro Cycle Time으로 정한다.
- 모든 마이크로 오퍼레이션의 동작시간이 비슷할 때 유리한 방식이다.

**동기 가변식(Synchronous Variable)**

- 수행시간이 유사한 Micro Operation끼리 그룹을 만들어, 각 그룹별로 서로 다른 Micro Cycle Time을 정의하는 방식이다.
- 동기 고정식에 비해 CPU 시간 낭비를 줄일 수 있는 반면, 제어기의 구현은 조금 복잡하다.
- 마이크로 오퍼레이션의 동작시간이 차이가 날 때 유리하다(정수배).

**비동기식(Asynchronous)**

- 모든 마이크로 오퍼레이션에 대하여 서로 다른 Micro Cycle Time을 정의하는 방식이다.
- CPU의 시간 낭비는 전혀 없으나, 제어기가 매우 복잡해지기 때문에 실제로는 거의 사용되지 않는다.

- 30 피연산자를 기억시킬 레지스터의 종류에 따라 컴퓨터 구조를 분류하면 GPR(범용 레지스터) 컴퓨터 구조, ACC(누산기) 컴퓨터 구조, 스택(Stack) 컴퓨터 구조가 있습니다.

**[전문가의 조언]**

이 문제의 경우 다시 출제되어도 보기 하나만 달리하여 출제될 확률이 높으므로 컴퓨터 구조의 종류 3가지를 확실히 숙지해 두세요. 그리고 각각의 구조들이 어떤 명령어 형식에서 사용하는지도 알아두세요.

- 3주소 명령어 : GPR(범용 레지스터)
- 2주소 명령어 : GPR(범용 레지스터)
- 1주소 명령어 : ACC(누산기)
- 0주소 명령어 : 스택(Stack)

- 31 [전문가의 조언]

자주 출제되는 문제입니다. 주요 연산자의 기능을 정리하고 넘어가세요.

**주요 연산자의 기능**

**AND(Masking Operation)**

- AND 연산은 특정 문자 또는 특정 비트를 삭제(Clear)시키는 연산으로, Masking 연산이라고도 한다.
- AND 연산은 삭제할 부분의 비트를 0과 AND시켜서 삭제하는 데, 대응시키는 0인 비트를 Mask Bit라 한다.

**OR(Selective-Set)**

- OR 연산은 특정 문자를 삽입하거나 특정 비트에 1을 세트시키는 연산으로, Selective Set 연산이라고도 한다.
- 삽입하거나 세트시킬 비트에 삽입할 문자 코드 또는 1을 OR 연산시킨다.

**XOR(Compare, 비교)**

- XOR 연산은 2개의 데이터를 비교하거나 특정 비트를 반전시킬 때 사용한다.

- 2개의 데이터를 XOR 연산하여 결과에 1Bit라도 1이 있으면 서로 다른 데이터이다.
- 반전시킬 때는 반전시킬 비트와 1을 XOR시킨다.

**NOT(Complement, 보수)**

NOT 연산은 각 비트의 값을 반전시키는 연산으로, 보수를 구할 때 사용한다.

**산술 Shift**

- 산술 Shift는 부호(Sign)를 고려하여 자리를 이동시키는 연산으로,  $2^n$ 으로 곱하거나 나눌 때 사용한다.
- 왼쪽으로 nBit Shift하면 원래 자료에  $2^n$ 을 곱한 값과 같고, 오른쪽으로 nBit Shift하면 원래 자료를  $2^n$ 으로 나눈 것과 같다.

- 32 카운터는 외부로부터의 입력과 현재 상태에 따라 출력이 결정되는 순서논리회로입니다.

**[전문가의 조언]**

단순히 조합논리회로와 순서논리회로의 종류를 구분하는 문제가 종종 출제됩니다. 어느 한쪽의 종류라도 정확히 기억해 두세요.

- **조합논리회로** : 반가산기, 전가산기, 병렬가산기, 반감산기, 전감산기, 디코더, 인코더, 멀티플렉서, 디멀티플렉서, 다수결회로, 비교기 등
- **순서논리회로** : 플립플롭, 카운터, 레지스터, RAM, CPU 등

- 33 64K×32Bit의 의미는 워드의 크기가 32비트인 기억장소가 64K개 있다는 의미입니다.

- K = 1024이므로 64K는 64×1024입니다.
- MAR은 64×1024(0~65536)개를 지정할 수 있는 비트 수, 즉 64K =  $2^6 \times 2^{10}$ 이므로 16비트가 필요합니다. MBR은 워드의 크기와 같으므로 32비트가 필요합니다.

**[전문가의 조언]**

크기만 달리하여 종종 출제되는 내용입니다. 워드의 길이는 워드를 담을 수 있는 MBR(DR), IR의 크기와 같고 주소선의 수는 주소를 지정할 때 사용하는 PC, MAR(AR)의 크기와 같다는 것을 알아두세요.

- 34 주기억장치의 페이지 수는 주기억장치의 용량을 페이지 크기로 나누면 됩니다.

- 페이지의 크기 = 1K워드 = 1K × 4Byte = 4,096Byte = 4K
- 페이지 수 = 512K / 4K = 128, 즉 문제에 주어진 주기억장치의 페이지 수는 128개입니다.

$$\ast K = 2^{10} = 1,024$$

**[전문가의 조언]**

이 문제 역시 크기만 달리하여 종종 출제되고 있습니다. 계산 과정을 정확히 파악하고 넘어가세요.

- 35 프로그램 실행 도중 분기(Branch)가 발생했다면 다음 실행할 명령의 주소를 가지고 있는 프로그램 카운터(PC; Program Counter)의 내용이 변경된 것입니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. 누산기를 중심으로 주요 레지스터의 기능을 꼭 기억하세요.

**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 중심임
- **상태 레지스터(Status Register), PSWR(Program Status Word Register), 플래그 레지스터** : 시스템 내부의 순간순간의 상태가 기록된 정보를 PSW라고 하며, 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

### 36 [전문가의 조언]

**중요해요!** 자주 출제되는 문제입니다. 연산자를 단항 연산자와 이항 연산자로 분류할 수 있어야 합니다.

**피연산자의 수에 따른 연산자의 분류**

NOT A처럼 피연산자가 1개만 필요한 연산자를 단항 연산자라 하고, A+B처럼 피연산자가 2개 필요한 연산자를 이항 연산자라 합니다.

- **단항 연산자(Unary Operator)** : NOT, Complement, Shift, Rotate, MOVE 등
- **이항 연산자(Binary Operator)** : 사칙연산, AND, OR, XOR, XNOR 등

### 37 [전문가의 조언]

자주 출제되는 내용입니다. 인터럽트 종류 및 발생 원인을 명확히 정리하세요.

**인터럽트 종류 및 발생 원인**

**외부 인터럽트(External Interrupt)**

- 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(Internal Interrupt)**

- 내부 인터럽트는 다음과 같이 잘못된 명령어나 데이터를 사용할 때 발생하며, 트랩(Trap)이라고도 부른다.
- **명령어 잘못에 의한 인터럽트** : 프로그램에서 명령어를 잘못 사용한 경우
- **프로그램 검사 인터럽트(Program Check Interrupt)** : 0으로 나누기(Divide by zero)가 발생한 경우나 Overflow 또는 Underflow가 발생한 경우

**소프트웨어 인터럽트(Software Interrupt)**

- 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

- 38 연관 기억장치는 키 레지스터의 Mask Bit와 대응하는 일부 내용에 의해 액세스합니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. 연관기억장치의 개념과 특징을 명확히 숙지하고, 구성 요소는 어떤 것들이 있는지 정도만 알아두세요.

**연관기억장치(Associative Memory)**

- 연관기억장치는 기억장치에서 자료를 찾을 때 주소에 의해 접근하지 않고, 기억된 내용의 일부를 이용하여 접근하는 기억장치로, CAM(Content Addressable Memory)이라고도 한다.
- **구성 요소** : 인수 레지스터, 키 레지스터, 매치 레지스터
- 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 논리회로를 갖고 있기 때문에 하드웨어 비용이 증가한다.

### 39 [전문가의 조언]

인터럽트와 관련된 문제가 자주 출제된다고 했죠. 이 문제를 통해 인터럽트의 정의를 명확히 기억해 두세요.

- 40 제어 기억장치의 최대 메모리 용량은 어드레스 필드를 이용하여 접근할 수 있는 '위드의 개수 × 위드의 크기'이므로 주어진 단서를 이용하여 어드레스 필드의 크기를 구해야 합니다.

- 플래그의 수가 4개라고 했으니 2Bit가 필요하고( $2^2=4$ ), 연산 필드가 12Bit라고 했으므로, 어드레스 필드의 크기는  $32-(12+2) = 18\text{Bit}$ 입니다.
- $2^{18}$ 개의 위드를 지정할 수 있는데 한 개 위드의 크기가 32Bit, 즉 4Byte이므로 최대 메모리 용량은  $2^{18} \times 4 = 1,048,576\text{Byte} = \text{약 } 1.1\text{M}$ 입니다.

※ 1M = 1,000,000Byte

**[전문가의 조언]**

**중요해요!** 명령어의 구성 형식을 꼭 정리하고 넘어가세요. **명령어의 구성**

컴퓨터에서 실행되는 명령어는 크게 연산자에 해당하는 연산자(Operation Code)부와 명령에 필요한 자료의 정보가 표시되는 자료(Operand) 부로 구성된다.

연산자(Operation Code) 부	자료(Operand) 부
-----------------------	---------------

**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.

#### 41 [전문가의 조언]

HIPO에 대해서는 특징과 종류를 묻는 문제가 출제됩니다. HIPO의 3가지 종류를 기억하고 특징을 숙지하세요.

##### HIPO(Hierarchy Input Process Output)

- 시스템 실행 과정인 입력, 처리, 출력을 계층적으로 기술하는 방법이다.
- 시스템을 설계하거나 문서화하기 위한 도구이다.
- 체계적인 문서 작성이 가능하며, 보기 쉽고 알기 쉽다.
- 하향(Top-Down) 방식을 사용하여 나타낸다.
- 개발 과정에서 문서화를 부산물로 얻을 수 있다.
- 도표상에 기능 위주로 입력 내용, 처리 방법, 출력 내용이 제시되므로 시스템을 이해하기 쉽다.
- 기능과 자료의 의존 관계를 동시에 표현할 수 있다.
- 유지보수 및 변경이 용이하다.

##### HIPO의 구성

- **도식 목차(Visual Table of Contents)** : HIPO에서 지정된 기능을 계층적으로 나타낸 도표로, 시스템의 구조와 각 기능의 관계를 도식화한 것
- **총괄 도표(Overview Diagram)** : 시스템 또는 프로그램의 기능을 입력, 처리, 출력 관계로 도표화한 것으로, 사용자 관점에서 본 시스템 또는 프로그램의 기능과 처리 내용을 나타냄
- **상세 도표(Detail Diagram)** : 총괄 도표에 나타난 기능을 구성하는 기본 요소들을 상세히 기술한 도표

#### 42 [전문가의 조언]

객체 지향 기법에서는 각 용어의 의미를 묻는 문제가 자주 출제됩니다. 정확히 숙지하고 넘어가세요.

##### 객체 지향 기법에서 사용되는 용어

- **캡슐화** : 데이터(속성)와 데이터를 처리하는 함수를 하나로 묶는 것
- **정보 은닉(Information Hiding)** : 캡슐화에서 가장 중요한 개념으로, 다른 객체에게 자신의 정보를 숨기고 자신의 연산만을 통하여 접근을 허용하는 것
- **추상화(Abstraction)** : 불필요한 부분을 생각하고 객체의 속성 중 가장 중요한 것에만 중점을 두어 개략화하는 것, 즉 모델화하는 것
- **상속성(Inheritance)** : 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
- **메시지(Message)** : 객체들 간에 상호작용을 하는데 사용되는 수단으로, 객체의 메소드(동작, 연산)를 일으키는 외부의 요구 사항
- **상속성(Inheritance)** : 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

- 43 보기를 잘 읽어보면 답이 보이는 문제입니다. ①번과 ②번에 '상호작용을 갖는~', '상호 관련하여~'라고 되어 있는데, ④

번은 '상호 관련이 없는~'이라고 되어 있습니다. 그러므로 정답은 ④번입니다.

##### [전문가의 조언]

동일한 형태로 종종 출제되는 문제입니다. 나머지 보기를 통해 시스템의 정의를 숙지하고 넘어가세요.

#### 44 [전문가의 조언]

출력 설계 시 각 단계에서 검토해야 할 사항을 묻는 문제가 종종 출제되니 잘 정리해 두세요.

##### 출력 설계

##### ① 출력 정보 내용의 설계

- 출력 정보의 목적을 달성하기 위해 어떤 내용을 출력할 것인지를 설계하는 단계이다.
- 출력할 항목과 명칭을 결정한다.
- 출력 항목의 배열 순서, 크기, 자릿수를 결정한다.
- 출력 항목의 문자(숫자, 영문자, 한글, 한자 등) 표현 방법을 결정한다.
- 출력 항목에 대한 집계(소개, 중계, 합계, 총계 등) 방법을 결정한다.
- 출력 정보의 오류 검사 방법을 결정한다.

##### ② 출력 정보 매체화의 설계

- 정보 처리 결과를 이용자가 요구하는 매체에 기록하기 위한 출력 형식을 설계하는 단계이다.
- 출력 형식을 결정한다.
- 특성과 작동의 용이성을 고려하여 출력 매체 및 장치를 결정한다.
- 출력 정보의 양과 출력 복사의 매수를 결정한다.
- 출력 장소와 출력 시기 및 주기를 결정한다.
- 출력 배열 순서를 결정한다.

##### ③ 출력 정보 분배에 대한 설계

- 출력 정보를 이용자에게 어떻게 전달할 것인지 전달할 경로와 방법에 대해 설계하는 단계이다.
- 분배 책임자, 분배 방법 및 형태, 분배 경로, 분배 주기 및 시기에 대해 결정한다.

##### ④ 출력 정보 이용에 대한 설계

- 어떤 정보를 누가, 언제, 어떻게 이용하느냐에 대해 설계하는 단계이다.
- 출력 정보명과 출력 정보의 사용 목적을 결정한다.
- 이용자와 이용 경로를 결정한다.
- 이용 주기 및 시기를 결정한다.
- 기밀성의 유무와 보존에 대해 결정한다.

#### 45 [전문가의 조언]

파일 설계 순서와 각 단계에서 결정되는 사항들을 묻는 문제가 자주 출제됩니다. 파일 설계 순서를 나열할 수 있도록 기억하고 각각의 단계에서는 어떤 일이 수행되는지 구분할 수 있도록 정리하고 넘어가세요.

##### 파일 설계 순서

- 파일의 성격 검토 → 파일 항목의 검토 → 파일의 특성 조사 → 파일 매체의 검토 → 편성법 검토
- **파일의 성격 검토**
  - 파일의 명칭을 결정한다.
  - 파일의 작성 목적과 종류를 결정한다.
  - 파일이 사용되는 적용 업무를 확인한다.
- **파일 항목의 검토**
  - 항목의 명칭과 저장 형식을 결정한다.
  - 항목의 배열 순서와 자릿수를 결정한다.
  - 레코드의 형식과 크기, 블록의 크기를 결정한다.
- **파일의 특성 조사**
  - 효율적인 파일의 처리 형태를 결정한다.

- 추가, 수정, 삭제의 발생 빈도와 처리 형태를 확인한다.
- 파일의 활동물을 확인한다.
- 순차 처리를 할 것인지 랜덤 처리를 할 것인지 처리 유형을 결정한다.
- **파일 매체의 검토** : 어느 매체가 업무에 가장 적합한가를 충분히 검토하여 매체를 선정해야 함
- **편성법 검토** : 순차 편성, 랜덤 편성, 색인 순차 편성, 리스트 편성 등 파일의 편성 방식을 결정함

- 46 입력 데이터가 기록될 매체에 대해 결정하는 것이므로, 입력 매체의 설계 단계에 속합니다.

**[전문가의 조언]**

자주 출제되는 내용은 아닙니다. 입력 매체의 설계 단계에서 수행하는 내용만 기억하고 넘어가세요.

- 47 불일치 레코드 체크(Unmatched Record Check)는 계산 처리 단계에서의 체크 방법입니다.

**[전문가의 조언]**

입력 단계에서의 검사 방법을 구분하는 문제가 자주 출제됩니다. 무슨 검사법을 말하는지를 맞힐 수 있도록 각각의 특징을 알아두세요.

**입력 단계에서의 오류 검사 방법**

- **균형 검사(Balance Check)** : 차변과 대변의 한계값을 검사하는 방법으로, 대차의 균형이나 가로, 세로의 합계가 일치하는가를 검사함
- **형식 검사(Format Check)** : 입력되는 데이터의 자릿수, 형식, 행, 열, 페이지 번호 등이 규정대로 되어 있는지를 검사하는 방법
- **한계 검사(Limit Check)** : 입력 데이터의 어떤 항목이 규정된 범위 내에 있는지를 검사하는 방법
- **일괄 합계 검사(Batch Total Check = Sum Check)** : 입력 데이터의 특정 항목 합계값을 미리 계산해서 이것을 입력 데이터와 함께 입력하고, 컴퓨터상에서 계산한 결과값과 수동 계산 결과값이 같은지를 검사하는 방법
- **타당성 검사(Validity Check)** : 입력된 데이터에 논리적으로 오류가 있는지를 검사하는 방법
- **숫자 검사(Numeric Check)** : 숫자형의 입력 항목에만 적용하는 기법으로, 입력된 데이터가 모두 숫자인가를 검사하는 방법
- **순차 검사(Sequence Check)** : 입력되는 데이터의 순서가 이미 정해진 순서와 일치하는지를 검사하는 방법
- **대조 검사(Matching Check)** : 입력 데이터와 시스템에 보관된 별도의 코드표를 대조하여 그것이 일치하는지를 검사하는 방법
- **데이터 수 검사(Data Count Check)** : 컴퓨터로 처리할 데이터의 개수를 미리 파악해 두었다가 컴퓨터로 처리한 데이터의 개수와 같은지를 검사하는 방법

- 48 순차 파일 편성은 레코드를 논리적인 처리 순서에 따라 연속된 물리적 저장 공간에 기록하는 것으로, 파일 내 레코드 추가, 삭제 시 파일 전체를 복사해야 합니다.

**[전문가의 조언]**

순차 파일(Sequential File)의 장점과 단점을 묻는 문제가 종종 출제되니 잘 정리해 두세요.

**순차 파일의 장점**

- 기록 밀도가 높아 기억공간을 효율적으로 사용할 수 있다.
- 매체 변환이 쉬워 어떠한 매체에도 적용할 수 있다.
- 레코드를 저장할 때 사용한 키 순서대로 레코드를 처리하는 경우 다른 편성법보다 처리 속도가 빠르다.

**순차 파일의 단점**

- 파일의 중간에서 레코드를 삽입·삭제하는 경우 파일 전체를 복사해야 하므로 시간이 많이 소요된다.
- 데이터 검색 시 항상 처음부터 순차적으로 하기 때문에 검색 효율이 낮다.

**49 [전문가의 조언]**

표준 처리 패턴에 대한 내용은 자주 출제되지만 대부분 문제를 통해 답을 유추할 수 있습니다. 어떤 표준 처리 패턴을 말하는지 구분할 수 있도록 각각의 특징을 파악해 두세요.

**표준 처리 패턴**

<b>변환 (Conversion)</b>	입력 매체상의 데이터에서 오류를 제거하고, 컴퓨터가 처리할 수 있는 형태로 편집하여 파일 매체로 변환(입력 변환)하고, 파일 매체에 저장된 내용을 사람이 확인할 수 있도록 출력 매체로 변환(출력 변환)하는 기능
<b>정렬 (Sort, 분류)</b>	레코드를 처리할 순서에 맞게 오름차순 또는 내림차순으로 재배치하는 기능
<b>병합(Merge)</b>	동일한 파일 형식을 갖는 2개 이상의 파일을 일정한 규칙에 따라 하나의 파일로 통합 처리하는 기능
<b>대조 (Matching)</b>	2개의 파일을 대조시켜 그 기록 순서나 기록 내용을 검사하는 기능
<b>갱신(Update)</b>	마스터 파일의 내용을 변동 파일에 의해 추가, 삭제, 수정 등의 작업을 하여 새로운 내용의 마스터 파일을 생성하는 기능
<b>분배 (Distribution)</b>	하나의 파일 안에서 조건에 맞는 것과 그렇지 않은 것을 분리하는 기능
<b>보고서 (Reporting)</b>	처리 결과를 출력하는 기능
<b>추출(Extract)</b>	파일 안에서 특정 조건에 만족하는 데이터만을 골라내는 기능으로, 정보 검색을 위한 필수 기능
<b>조합(Collate)</b>	2개 이상의 파일에서 조건에 맞는 것을 골라 새로운 레코드로 파일을 만드는 기능

- 50 파일의 활동물 검토는 파일의 특성 조사 단계에서 검토할 사항입니다.

**[전문가의 조언]**

종종 출제되는 내용입니다. 파일 매체 검토 단계에서 검토할 사항을 확인해 보세요.

**파일 매체의 검토 시 검토할 사항**

<b>기능 검토 사항</b>	<ul style="list-style-type: none"> <li>• 액세스 형태와 처리 방식</li> <li>• 처리 시간과 정보의 양</li> <li>• 작동의 용이성</li> </ul>
<b>종합 검토 사항</b>	<ul style="list-style-type: none"> <li>• 저장 매체와 매체의 개수 결정</li> <li>• 장치의 대수 결정</li> </ul>

- 51 프로세스 설계 시 오류에 대비한 체크 시스템을 고려해야 하며, 분류 처리는 가능한 한 최소화해야 합니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. 프로세스 설계 시 유의사항을 꼭 숙지하세요.

**프로세스 설계 시 유의사항**

- 신뢰성과 정확성을 고려하여 처리 과정을 간결하고 명확히 표현한다.
- 오류에 대비한 검사 시스템을 고려한다.



- 시스템의 상태 및 구성 요소, 기능 등을 종합적으로 표시한다.
- 새로운 시스템의 프로세스 설계뿐만 아니라 기존 시스템의 문제점 분석이 가능하도록 설계한다.
- 정보의 흐름이나 처리 과정을 모든 사람이 이해할 수 있도록 표준화한다.
- 분류 처리는 가능한 한 적게 한다.
- 조작을 간결화, 자동화하여 사용자의 수동 조작을 적게 한다.

- 52 지문에 제시된 코드는 학과별 코드를 ‘공대’, ‘사범대’의 블록으로 구분하고 각 블록 내에서 학과별로 일련 번호를 부여한 것으로 이는 구분 코드(Block Code)에 해당합니다.

#### [전문가의 조언]

무슨 코드인지를 묻는 문제는 자주 출제됩니다. 각 코드의 특징을 반드시 기억해 두세요.

#### 코드의 종류

순서 코드 (Sequence Code)	자료의 발생 순서, 크기 순서 등 일정 기준에 따라서 최초의 자료부터 차례로 일련 번호를 부여하는 방법
구분 코드 (Block Code)	코드화 대상 항목 중에서 공통성이 있는 것끼리 블록으로 구분하고, 각 블록 내에서 일련 번호를 부여하는 방법
그룹 분류식 코드 (Group Classification Code)	코드화 대상 항목을 일정 기준에 따라 대분류, 중분류, 소분류 등으로 구분하고, 각 그룹 안에서 일련 번호를 부여하는 방법
10진 코드 (Decimal Code)	코드화 대상 항목을 0~9까지 10진 분할하고, 다시 그 각각에 대하여 10진 분할하는 방법을 필요한 만큼 반복함
연상 코드 (Mnemonic Code)	코드화 대상 항목의 명칭이나 약호와 관계있는 숫자나 문자, 기호를 이용하여 코드를 부여하는 방법
약자식 코드 (Letter Type Code)	코드화 대상 항목의 약자를 그대로 코드로 사용하는 방법
합성 코드 (Combined Code)	필요한 기능을 하나의 코드로 수행하기 어려운 경우 두 개 이상의 코드를 조합하여 만드는 방법
끝자리 분류 코드 (Final Digit Code)	코드의 끝자리 수에 의미를 부여하는 코드 체계로, 다른 종류의 코드 분류 방법과 조합해서 사용하는 코드 분류 방법

#### 53 [전문가의 조언]

가끔 출제되는 내용입니다. 시스템 설계의 개념과 기본 설계와 상세 설계의 기능을 간단히 정리하고 넘어가세요.

#### 시스템 설계

- 시스템 분석에 의해 정의된 시스템 요구 분석 명세서를 토대로 하여 새로운 시스템을 구체화하는 단계로, 기본 설계와 상세 설계가 있다.
- **기본 설계** : 분석 결과에 따라 사용자 입장에서 시스템 전체를 개괄적으로 설계하는 과정
- **상세 설계** : 각 기능의 논리적인 절차를 확정하고, 구체적인 입·출력 내용 및 파일의 구체적인 내용의 사양을 결정하기 위한 단계

- 54 IPT에서는 하향식 프로그래밍 기법을 사용합니다.

#### [전문가의 조언]

IPT가 하향식 프로그래밍 기법을 사용한다는 것만 알아도 맞힐 수 있는 문제가 종종 출제됩니다. IPT는 하향식 프로그래밍이라는 것을 기억하고, IPT의 기술적 측면을 한번 읽고 넘어가세요.

#### IPT의 기술적 측면

- **설계 분야 - 구조적 설계(Structured Design, 복합 설계)** : 상세 설계의 초기 단계에서 프로그램의 구조를 설계하는 방식으로, 프로그램의 기능을 하향식으로 상세화해서 종속 기능을 모듈에 각각 대응시킴
- **코딩 분야 - 구조적 코딩(Structured Coding)** : 코딩 기법의 표준화 방법으로, GOTO문을 배제하고 순차, 선택, 반복 구조만을 사용하여 신뢰성을 향상시킨 코딩 기법
- **테스트 분야 - 하향식 프로그래밍(Top-Down Programming)** : 프로그램의 코딩과 테스트를 프로그램 구조의 상위 모듈로부터 하위 모듈로 순차적으로 진행시키는 방법

- 55 문제와 같이 좌우 자리가 바뀌어 발생하는 오류는 전위 오류(Transposition Error)입니다.

#### [전문가의 조언]

코드의 오류에 대한 문제는 전위 오류(Transposition Error)와 필사 오류(Transcription Error)를 묻는 문제가 대부분인데, 새로운 형태가 출제되었네요. 두 오류를 중심으로 각각의 오류 현상을 파악하세요.

#### 오류의 종류

- **필사 오류(Transcription Error) = 오자 오류** : 입력 시 임의의 한 자리를 잘못 기록한 경우 발생
- **전위 오류(Transposition Error)** : 입력 시 좌우 자리를 바꾸어 기록한 경우 발생
- **이중 오류(Double Transposition Error)** : 전위 오류가 2개 이상 발생한 경우
- **생략 오류(Omission Error)** : 입력 시 한 자리를 빼놓고 기록한 경우 발생
- **추가 오류(Addition Error)** : 입력 시 한 자리를 더 추가하여 기록한 경우 발생
- **임의 오류(Random Error)** : 오류가 두 가지 이상 결합하여 발생한 경우

#### 56 [전문가의 조언]

응집도에 대해서는 응집도의 의미, 종류, 응집 정도에 따른 순서를 묻는 문제가 자주 출제됩니다. 확실히 숙지해 두세요.

#### 응집도(Cohesion)

- 정보 은닉 개념을 확장한 것으로 모듈 안의 요소들이 서로 관련되어 있는 정도, 즉 모듈이 독립적인 기능으로 정의되어 있는 정도를 의미한다.
- 독립적인 모듈이 되기 위해서는 각 모듈의 응집도가 강해야 한다.
- **응집도의 종류(강함>약함)** : 기능적 응집도 > 순차적 응집도 > 교환(통신)적 응집도 > 절차적 응집도 > 시간적 응집도 > 논리적 응집도 > 우연적 응집도
  - 기능적 응집도(Functional Cohesion) : 모듈 내부의 모든 기능 요소들이 단일 문제와 연관되어 수행될 경우의 응집도
  - 순차적 응집도(Sequential Cohesion) : 모듈 내의 하나의 활동으로부터 나온 출력 데이터를 그 다음 활동의 입력 데이터로 사용할 경우의 응집도
  - 교환(통신)적 응집도(Communication Cohesion) : 동일



한 입력과 출력을 사용하여 서로 다른 기능을 수행하는 구성 요소들이 모였을 경우의 응집도

- 절차적 응집도(Procedural Cohesion) : 모듈이 다수의 관련 기능을 가질 때 모듈 안의 구성 요소들이 그 기능을 순차적으로 수행할 경우의 응집도
- 시간적 응집도(Temporal Cohesion) : 특정 시간에 처리되는 몇 개의 기능을 모아 하나의 모듈로 작성할 경우의 응집도
- 논리적 응집도(Logical Cohesion) : 유사한 성격을 갖거나 특정 형태로 분류되는 처리 요소들로 하나의 모듈이 형성되는 경우의 응집도
- 우연적 응집도(Coincidental Cohesion) : 모듈 내부의 각 구성 요소들이 서로 관련 없는 요소로만 구성된 경우의 응집도

- 57 프로세스 설계는 입력 정보와 파일 정보를 토대로 하여 필요한 출력 정보를 얻기까지의 처리 과정을 설계하는 것으로, 사용자 하드웨어와 프로그래밍에 관한 상식 수준까지 고려할 필요는 없습니다.

#### [전문가의 조언]

이번 회차에서는 프로세스 설계 시 유의사항에 대한 문제가 두 문제나 출제되었네요. 이 문제를 틀렸다면 51번 문제의 [전문가의 조언]을 다시 한번 공부하세요.

- 58 시스템 분석은 시스템을 운영하는 인간을 중심으로 수행해야 합니다.

#### [전문가의 조언]

가끔 출제되는 문제입니다. 시스템 분석을 위해 시스템 분석가가 갖추어야 하는 기본 사항을 간단히 정리하고 넘어가세요.

#### 시스템 분석가가 갖추어야 할 사항

- 기업의 목적과 현행 시스템의 문제점을 정확히 이해하고 해결책을 제시할 수 있어야 한다.
- 업무 내용이나 시스템에 대한 분석 능력이 있어야 한다.
- 컴퓨터 기술과 관리 기법을 알아야 한다.
- 시간 배정과 계획 등을 빠른 시간 내에 파악할 수 있어야 한다.
- 컴퓨터 하드웨어와 소프트웨어에 대한 전반적인 지식을 가져야 한다.
- 업계의 동향 및 관계 법규 등을 파악할 수 있어야 한다.
- 창조력, 응용력이 있어야 한다.
- 현장 분석 경험이 있어야 한다.
- 사용자와 프로그래머, 경영진 간의 의사소통을 원활히 하는 해결사 역할을 수행할 수 있어야 한다.

- 59 [전문가의 조언]

가끔 출제되는 내용인데, 평가 항목이 3가지뿐이라 매번 보기 하나만 달리하여 출제되고 있습니다. 평가 항목 3가지를 정확히 기억해 두세요.

- 60 문서화란 시스템 개발에 관련된 모든 행위를 문서로 만들어 두는 것으로, 개발자의 순서도 작성, 코딩, 디버깅, 테스트를 위한 것은 아닙니다.

#### [전문가의 조언]

거의 매회 출제되는 내용입니다. 문서화의 목적 및 효과가 아닌 것을 찾아낼 수 있도록 정리해 두세요.

#### 문서화의 목적 및 효과

- 시스템 개발팀에서 운용팀으로 인수 인계가 용이하다.
- 개발 후에 시스템의 유지보수가 용이하다.
- 시스템을 쉽게 이해할 수 있다.

- 개발팀을 원활히 운용할 수 있다.
- 시스템 개발 중의 추가 변경 또는 시스템 개발 후의 변경에 따른 혼란을 방지할 수 있다.
- 시스템 개발 방법과 순서를 표준화할 수 있어 효율적인 작업과 관리가 가능하다.
- 복수 개발자에 의한 병행 개발을 가능하게 한다.
- 프로그램을 공유 재산화 할 수 있다.

#### 61 [전문가의 조언]

스케줄링 기법의 종류를 구분하는 문제가 종종 출제됩니다. 나머지 스케줄링 기법의 원리도 정리해 두세요.

- **RR(Round Robin)** : 시분할 시스템(Time Sharing System)을 위해 고안된 방식으로, FCFS 알고리즘을 선점 형태로 변형한 기법이며, FCFS 기법과 같이 준비상태 큐에 먼저 들어온 프로세스가 먼저 CPU를 할당받지만 각 프로세스는 시간 할당량(Time Slice) 동안만 실행한 후 실행이 완료되지 않으면 다음 프로세스에게 CPU를 넘겨주고 준비상태 큐의 가장 뒤로 배치됨
- **SRT(Shortest Remaining Time)** : 비선점 스케줄링인 SJF 기법을 선점 형태로 변경한 기법으로, 현재 실행 중인 프로세스의 남은 시간과 준비상태 큐에 새로 도착한 프로세스의 실행 시간을 비교하여 가장 짧은 실행 시간을 요구하는 프로세스에게 CPU를 할당하며, 시분할 시스템에 유용함
- **FIFO(First In First Out) = FCFS** : 준비상태 큐에 도착한 순서에 따라 차례로 CPU를 할당하는 기법으로, 가장 간단한 알고리즘이며, 먼저 도착한 것이 먼저 처리되어 공정성은 유지되지만 짧은 작업이 긴 작업을, 중요한 작업이 중요하지 않은 작업을 기다리게 됨

- 62 ②번은 최소 특권, ③번은 특권의 분할, ④번은 메커니즘의 경제성을 의미합니다.

#### [전문가의 조언]

문제와 보기가 동일하게 여러 번 출제된 적이 있는 문제입니다. 문제와 답만 기억하고 넘어가세요.

- 63 스케줄링은 프로세스가 생성되어 실행될 때 필요한 시스템의 여러 자원을 해당 프로세스에게 할당하는 작업을 의미하는 것으로, 운영체제의 오버헤드를 최소화하기 위해 사용됩니다.

#### [전문가의 조언]

가끔 출제되는 문제입니다. 스케줄링의 여러 가지 목적에 대해 한번 가볍게 읽고 넘어가세요.

#### 스케줄링의 목적

- **공정성** : 모든 프로세스에 공정하게 할당함
- **처리율(량) 증가** : 단위 시간당 프로세스를 처리하는 비율(량)을 증가시킴
- **CPU 이용률 증가** : 프로세스 실행 과정에서 주기억장치를 액세스한다든지, 입·출력 명령 실행 등의 원인에 의해 발생할 수 있는 CPU의 낭비 시간을 줄이고, CPU가 순수하게 프로세스를 실행하는 데 사용되는 시간 비율을 증가시킴
- **우선순위 제도** : 우선순위가 높은 프로세스를 먼저 실행함
- **오버헤드 최소화** : 오버헤드를 최소화함
- **응답 시간(Response Time, 반응 시간) 최소화** : 작업을 지시하고, 반응하기 시작하는 시간을 최소화함
- **반환 시간(Turn Around Time) 최소화** : 프로세스를 제출한 시간부터 실행이 완료될 때까지 걸리는 시간을 최소화함
- **대기 시간 최소화** : 프로세스가 준비상태 큐에서 대기하는 시간을 최소화함

- **균형 있는 자원의 사용** : 메모리, 입·출력장치 등의 자원을 균형 있게 사용함
- **무한 연기 회피** : 자원을 사용하기 위해 무한정 연기되는 상태를 회피함

64 교착상태(Deadlock)의 4가지 필요 조건은 상호 배제, 점유와 대기, 비선점, 환형 대기입니다. ①번은 상호 배제, ②번은 점유와 대기, ③번은 비선점에 대한 설명입니다. 환형 대기는 공유 자원과 공유 자원을 사용하기 위해 대기하는 프로세스들이 원형으로 구성되어 있어 자신에게 할당된 자원을 점유하면서 앞이나 뒤에 있는 프로세스의 자원을 요구하는 상태를 의미합니다.

#### [전문가의 조언]

교착상태 발생의 4가지 필요 조건과 각각의 의미를 묻는 문제가 자주 출제되니 문제를 통해 꼭 숙지해 두세요.

65 ①번은 처리 능력(Throughput), ②번은 반환 시간(Turn Around Time), ③번은 신뢰도(Reliability)에 대한 설명입니다.

#### [전문가의 조언]

운영체제의 목적이나 성능평가 기준을 묻는 문제가 자주 출제되니 잘 정리해 두세요.

#### 운영체제 목적

운영체제의 목적에는 처리 능력 향상, 사용 가능성 향상, 신뢰도 향상, 반환 시간 단축 등이 있다. 처리 능력, 반환 시간, 사용 가능성, 신뢰도는 운영체제의 성능을 평가하는 기준이 된다.

#### 운영체제의 성능평가 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 가능성 (Availability)	시스템을 사용할 필요가 있을 때 즉시 사용 가능한 정도
신뢰도(Reliability)	시스템이 주어진 문제를 정확하게 해결하는 정도

66 파일 디스크립터는 파일 시스템이 관리하므로 사용자가 직접 참조할 수 없습니다.

#### [전문가의 조언]

파일 디스크립터의 특징 및 포함하는 정보를 묻는 문제가 종종 출제됩니다. 꼭 숙지하고 넘어가세요.

#### 파일 디스크립터의 특징

- 파일을 관리하기 위해 시스템(운영체제)이 필요로 하는 파일에 대한 정보를 갖고 있는 제어 블록을 의미하며, 파일 제어 블록(FCB; File Control Block)이라고도 한다.
- 파일 디스크립터는 파일마다 독립적으로 존재하며, 시스템에 따라 다른 구조를 가질 수 있다.
- 보통 파일 디스크립터는 보조기억장치 내에 저장되어 있다가, 해당 파일이 Open될 때 주기억장치로 옮겨진다.
- 파일 디스크립터는 파일 시스템이 관리하므로 사용자가 직접 참조할 수 없다.

#### 파일 디스크립터의 정보

- 파일 이름
- 보조기억장치에서의 파일 위치
- **파일 구조** : 순차 파일, 색인 순차 파일, 색인 파일 등
- **보조기억장치의 유형** : 자기 디스크, 자기 테이프 등
- 액세스 제어 정보
- **파일 유형** : 텍스트 파일, 목적 프로그램 파일(2진 파일,

기계어 파일, 실행 파일) 등

- 생성 날짜와 시간, 제거 날짜와 시간
- 최종 수정 날짜 및 시간
- **액세스한 횟수** : 파일 사용 횟수

#### 67 [전문가의 조언]

종종 출제되는 내용입니다. 자원 보호 기법의 종류 및 각각의 특징을 정리해 두세요.

#### 자원 보호 기법

##### • 접근 제어 행렬(Access Control Matrix)

- 접근 제어 행렬은 자원 보호의 일반적인 모델로, 객체에 대한 접근 권한을 행렬로써 표시한 기법이다.
- 행(Row)은 영역(사용자, 프로세스), 열(Column)은 객체, 각 항은 접근 권한의 집합으로 구성된다.

##### • 전역 테이블(Global Table)

- 전역 테이블은 가장 단순한 구현 방법으로, 3개의 순서쌍인 영역, 객체, 접근 권한의 집합을 목록 형태로 구성한 기법이다.
- 테이블이 매우 커서 주기억장치에 저장할 수 없으므로 가상기억장치 기법을 사용해야 하며, 주기억장치에 저장될 경우 공간을 낭비하게 된다.

##### • 접근 제어 리스트(Access Control List)

- 접근 제어 리스트는 접근 제어 행렬에 있는 각 열, 즉 객체를 중심으로 접근 리스트를 구성한 것이다.
- 각 객체에 대한 리스트는 영역, 접근 권한의 순서쌍으로 구성되며, 객체에 대한 접근 권한을 갖는 모든 영역을 정의한다.
- 접근 권한이 없는 영역은 제외된다.

##### • 권한(자격) 리스트(Capability List)

- 권한(자격) 리스트는 접근 제어 행렬에 있는 각 행, 즉 영역을 중심으로 권한 리스트를 구성한 것이다.
- 각 영역에 대한 권한 리스트는 객체와 그 객체에 허용된 조작 리스트로 구성된다.
- 권한 리스트는 영역과 결합되어 있지만 그 영역에서 수행 중인 프로세스가 직접 접근할 수는 없다. 왜냐하면 권한 리스트는 운영체제에 의하여 유지되며 사용자에게 의해서 간접적으로만 접근되는 보호된 객체이기 때문이다.

##### • 록-키(Lock-Key)

- 록-키는 접근 제어 리스트와 권한 리스트를 절충한 기법이다.
- 각 객체는 Lock, 각 영역은 Key라 불리는 유일하고도 독특한 값을 갖고 있어서 영역과 객체가 일치하는 경우에만 해당 객체를 접근할 수 있다.

68 가상 기억장치는 프로그램 실행 시 주소 변환 작업을 수행하므로 수행 중인 프로그램에서 사용된 주소가 반드시 주기억장치에서 사용 가능한 주소일 필요는 없습니다.

#### [전문가의 조언]

가끔씩 출제되는 문제입니다. 가상기억장치에 대한 개념 및 특징을 간단히 정리하고 넘어가세요.

#### 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 가상기억 장치의 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.

### 69 [전문가의 조언]

라운드 로빈(Round Robin) 스케줄링에 대한 문제는 자주 출제됩니다. 이 문제를 통해 라운드 로빈의 개념을 정확히 숙지해 두세요.

- 70 은행원 알고리즘은 교착상태가 발생할 가능성을 배제하지 않고 교착상태가 발생하면 적절히 피해나가는 교착상태 회피 기법입니다.

### [전문가의 조언]

문제와 보기가 동일하게 출제되었습니다. 이 문제에서는 은행원 알고리즘의 기능과 상호 배제 구현 기법의 종류만 기억하고 넘어가세요.

### 상호 배제 기법을 구현하기 위한 방법

- **소프트웨어적 구현 방법** : 데커(Dekker) 알고리즘, 피터슨(Peterson) 알고리즘, Lamport의 빵집 알고리즘
- **하드웨어적 구현 방법** : Test\_And\_Set 기법, Swap 명령어 기법

### 71 [전문가의 조언]

디스크 스케줄링 기법을 구분하는 문제는 종종 출제됩니다. 각 기법의 동작원리를 정확히 파악해 두세요.

### 디스크 스케줄링의 종류

FCFS	가장 간단한 스케줄링으로, 디스크 대기 큐에 가장 먼저 들어온 트랙에 대한 요청을 먼저 서비스하는 기법
SSTF	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
C-SCAN	항상 바깥쪽에서 안쪽으로 움직이면서 가장 짧은 탐색 거리를 갖는 요청을 서비스하는 기법
N-step SCAN	SCAN 기법을 기초로 하며 어떤 방향의 진행이 시작될 당시에 대기중이던 요청들만 서비스하고, 진행 도중 도착한 요청들은 한데 모아서 다음의 반대 방향 진행 때 서비스하는 기법
LOOK	SCAN 기법을 기초로 사용하되 진행 방향의 마지막 요청을 서비스한 후 그 방향의 끝으로 이동하는 것이 아니라 바로 역방향으로 진행하는 기법

- 72 분산 처리 시스템은 독립적인 처리 능력을 가진 컴퓨터 시스템을 통신망으로 연결한 시스템으로 보안에 취약합니다.

### [전문가의 조언]

분산 처리 시스템의 장·단점을 묻는 문제가 종종 출제되니

잘 파악해 두세요.

### 분산 처리 시스템의 장·단점

장점	<ul style="list-style-type: none"> <li>• 여러 사용자들 간에 통신이 용이함</li> <li>• 제한된 장치를 여러 지역의 사용자가 공유할 수 있음</li> <li>• 여러 사용자들이 데이터를 공유할 수 있음</li> <li>• 중앙 컴퓨터의 과부하를 줄일 수 있음</li> <li>• 사용자는 각 컴퓨터의 위치를 몰라도 자원을 사용할 수 있음</li> <li>• 업무량의 증가에 따른 시스템의 점진적인 확장이 용이함</li> <li>• 하나의 일을 여러 시스템이 처리함으로써 연산 속도, 신뢰도, 사용 가능도가 향상되고, 결함 허용이 가능함</li> </ul>
단점	<ul style="list-style-type: none"> <li>• 중앙 집중형 시스템에 비해 소프트웨어 개발이 어려움</li> <li>• 보안 문제가 발생함</li> <li>• 시스템 유지상 통일성을 잃기 쉬움</li> <li>• 시스템의 설계가 복잡하고, 데이터 처리 서비스의 질이 떨어짐</li> </ul>

### 73 [전문가의 조언]

자주 출제되는 문제는 아닙니다. 이 문제에서는 단편화(Fragmentation)의 개념만 기억하고 넘어가세요.

- 74 스프레드시트는 응용 소프트웨어입니다.

### [전문가의 조언]

시스템 소프트웨어와 응용 소프트웨어의 개념과 종류를 구분할 수 있을 정도로만 정리하고 넘어가세요.

### 시스템 소프트웨어

- 시스템 소프트웨어는 시스템 전체를 작동시키는 프로그램으로, 프로그램을 주기억장치에 적재시키거나 인터럽트 관리, 장치 관리, 언어 번역 등의 기능을 담당한다.
- **종류** : 운영체제, 언어 번역 프로그램, 매크로 프로세서, 링커, 라이브러리, 정렬/합병 프로그램, 로더 등

### 응용 소프트웨어

- 응용 소프트웨어는 사용자가 컴퓨터를 이용하여 특정 업무를 처리할 수 있게 개발된 프로그램이다.
- **종류** : 워드프로세서, 스프레드시트, DBMS, 프레젠테이션 등

- 75 세마포어가 항상 양수 값을 갖지는 않습니다. 이진 세마포어는 0 또는 1, 산술 세마포어는 0 또는 양의 정수를 가집니다.

### [전문가의 조언]

세마포어와 관련해서는 개념 및 사용되는 연산 등 다양한 문제가 출제되니 잘 정리해 두세요.

### 세마포어(Semaphore)

- 세마포어는 '신호기', '깃발'을 뜻하며, 각 프로세스에 제어 신호를 전달하여 순서대로 작업을 수행하도록 하는 기법이다.
- 세마포어는 다익스트라(E. J. Dijkstra)가 제안하였으며, P와 V라는 두 개의 연산에 의해서 동기화를 유지시키고 상호 배제의 원리를 보장한다.
- 세마포어의 종류
  - 이진 세마포어(Binary Semaphore) : 세마포어 변수가 0 또는 1을 가지며, 1 이상의 정수로 초기화 됨
  - 산술 세마포어(Counting Semaphore) : 세마포어 변수가 0 또는 양의 정수를 가지며, 0 이상의 정수로 초기화 됨
- S는 P와 V 연산으로만 접근 가능한 세마포어 변수로, 공유



자원의 개수를 나타내며 0과 1 혹은 0과 양의 값을 가질 수 있다.

- 76 FCFS 스케줄링 기법은 준비상태 큐에 도착한 순서에 따라 차례로 CPU를 할당하는 기법이므로 이동 순서는 '50 → 10 → 40 → 55 → 35' 순으로 진행됩니다. 그러므로 총 이동거리는  $40 + 30 + 15 + 20 = 105$ 입니다.

**[전문가의 조언]**

FCFS 스케줄링의 동작 원리를 알면 이동 순서를 찾을 수 있으니 동작 원리를 기억해 두세요.

- 77 주/종 처리기 시스템에서 운영체제를 수행하는 것은 주프로세서입니다.

**[전문가의 조언]**

주/종(Master/Slave) 처리기 시스템에서의 주프로세서와 종프로세서의 역할을 묻는 문제가 자주 출제됩니다. 각각의 역할을 확실히 파악해 두세요.

**주/종 처리기**

주프로세서	<ul style="list-style-type: none"> <li>입·출력과 연산을 담당함</li> <li>운영체제를 수행함</li> </ul>
종프로세서	<ul style="list-style-type: none"> <li>연산만 담당함</li> <li>입·출력 발생 시 주프로세서에게 서비스를 요청함</li> <li>사용자 프로그램만 담당함</li> </ul>

- 78 **[전문가의 조언]**

UNIX 명령어들의 기능을 묻는 문제는 자주 출제됩니다. 나머지 보기로 제시된 명령어의 기능도 꼭 알아두세요.

- **chgrp** : 파일의 사용자 그룹을 변경함
- **chown** : 소유자를 변경함
- **cpio** : 파일을 복사함

- 79 ②번은 반입 전략, ③번은 교체 전략, ④번은 예상 반입 전략에 대한 설명입니다.

**[전문가의 조언]**

배치 전략, 반입 전략, 교체 전략 등 각각의 의미보다는 각 전략의 종류 및 동작 원리를 묻는 문제가 자주 출제됩니다. 이 문제에서는 각 전략의 의미만 간단히 정리하고 넘어가세요.

**기억장치의 관리 전략**

- **반입(Fetch) 전략** : 보조기억장치에 보관중인 프로그램이나 데이터를 언제 주기억장치로 적재할 것인지를 결정하는 전략
- **배치(Placement) 전략** : 새로 반입되는 프로그램이나 데이터를 주기억장치의 어디에 위치시킬 것인지를 결정하는 전략
- **교체(Replacement) 전략** : 주기억장치의 모든 영역이 이미 사용중인 상태에서 새로운 프로그램이나 데이터를 주기억장치에 배치하려고 할 때, 이미 사용되고 있는 영역 중에서 어느 영역을 교체하여 사용할 것인지를 결정하는 전략

- 80 **[전문가의 조언]**

커널(Kernel)과 셸(Shell)의 기능을 구분하는 문제가 자주 출제됩니다. 자원 관리의 커널(Kernel), 명령 해석과 사용자 인터페이스는 셸(Shell)의 기능이라는 것을 중심으로 커널과 셸의 기능을 정확히 파악해 두세요.

**커널(Kernel)**

- UNIX의 가장 핵심적인 부분이다.
- 컴퓨터가 부팅될 때 주기억장치에 적재된 후 상주하면서

실행된다.

- 하드웨어를 보호하고, 프로그램과 하드웨어 간의 인터페이스 역할을 담당한다.
- 프로세스 관리, 기억장치 관리, 파일 관리, 입·출력 관리, 프로세스간 통신, 데이터 전송 및 변환 등 여러 가지 기능을 수행한다.

**셸(Shell)**

- 사용자의 명령어를 인식하여 프로그램을 호출하고 명령을 수행하는 명령어 해석기이다.
- 시스템과 사용자 간의 인터페이스를 담당한다.
- DOS의 COMMAND.COM과 같은 기능을 수행한다.
- 주기억장치에 상주하지 않고, 명령어가 포함된 파일 형태로 존재하며 보조기억장치에서 교체 처리가 가능하다.
- 공용 Shell(Bourne Shell, C Shell, Korn Shell)이나 사용자 자신이 만든 Shell을 사용할 수 있다.

- 81 UDP는 전송 계층 프로토콜입니다.

**[전문가의 조언]**

OSI 7계층에서는 주로 OSI 7계층 가운데 어떤 계층을 설명한 것인지를 묻는 문제가 출제됩니다. 각 계층의 주요 기능을 파악해 두세요.

물리 계층	<ul style="list-style-type: none"> <li>• 전송에 필요한 두 장치 간의 실제 접속과 절단 등 기계적, 전기적, 기능적, 절차적 특성에 대한 규칙을 정의함</li> <li>• 물리적 전송 매체와 전송 신호 방식을 정의하며, RS-232C, X.21 등의 표준이 있음</li> </ul>
데이터 링크 계층	<ul style="list-style-type: none"> <li>• 두 개의 인접한 개방 시스템들 간에 신뢰성 있고 효율적인 정보 전송을 할 수 있도록 함</li> <li>• 송신 측과 수신 측의 속도 차이를 해결하기 위한 흐름 제어 기능을 함</li> <li>• 프레임의 시작과 끝을 구분하기 위한 프레임의 동기화 기능을 함</li> <li>• 오류의 검출과 회복을 위한 오류 제어 기능을 함</li> <li>• 프레임의 순서적 전송을 위한 순서 제어 기능을 함</li> <li>• HDLC, LAPB, LLC, LAPD 등의 표준이 있음</li> </ul>
네트워크 계층	<ul style="list-style-type: none"> <li>• 개방 시스템들 간의 네트워크 연결을 관리하는 기능과 데이터의 교환 및 중계 기능을 함</li> <li>• 네트워크 연결을 설정, 유지, 해제하는 기능을 함</li> <li>• 경로 설정(Routing), 데이터 교환 및 중계, 트래픽 제어, 패킷 정보 전송을 수행함</li> <li>• 관련 표준으로는 X.25, IP 등이 있음</li> </ul>
전송(트랜스포트) 계층	<ul style="list-style-type: none"> <li>• 논리적 안정과 균일한 데이터 전송 서비스를 제공함으로써 종단 시스템(End-to-End) 간에 투명한 데이터 전송을 가능하게 함</li> <li>• OSI 7계층 중 하위 3계층과 상위 3계층의 인터페이스(Interface)를 담당함</li> <li>• 종단 시스템(End-to-End) 간의 전송 연결 설정, 데이터 전송, 연결 해제 기능을 함</li> <li>• 주소 설정, 다중화, 오류 제어, 흐름 제어를 수행함</li> <li>• TCP, UDP 등의 표준이 있음</li> </ul>
세션 계층	<ul style="list-style-type: none"> <li>• 송·수신측 간의 관련성을 유지하고 대화 제어를 담당하는 계층</li> <li>• 대화(회화) 구성 및 동기 제어, 데이터 교환 관리 기능을 함</li> <li>• 송·수신측의 대화(회화) 동기를 위해 전송하는 정보의 일정한 부분에 체크점을 두어 정보의 수신 상태를 체크하며, 이때의 체크점을 동기점이라고 함</li> <li>• 동기점은 오류가 있는 데이터의 회복을 위해 사용하는 것으로, 종류에는 소동기점과 대동기점이 있음</li> </ul>
표현	<ul style="list-style-type: none"> <li>• 응용 계층으로부터 받은 데이터를 세션 계층에 보</li> </ul>



<b>계층</b>	<p>대기 전에 통신에 적당한 형태로 변환하고, 세션 계층에서 받은 데이터는 응용 계층에 맞게 변환하는 기능을 함</p> <ul style="list-style-type: none"> <li>• 서로 다른 데이터 표현 형태를 갖는 시스템 간의 상호 접속을 위해 필요한 계층</li> <li>• 코드 변환, 데이터 암호화, 데이터 압축, 구문 검색, 정보 형식(포맷) 변환, 문맥 관리 기능을 함</li> </ul>
<b>응용 계층</b>	<ul style="list-style-type: none"> <li>• 사용자(응용 프로그램)가 OSI 환경에 접근할 수 있도록 서비스를 제공함</li> <li>• 응용 프로세스 간의 정보 교환, 전자 사서함, 파일 전송 등의 서비스를 제공함</li> </ul>

**82 [전문가의 조언]**

동기식 전송과 비동기식 전송의 특징을 비교하는 문제가 출제됩니다. 비동기식 전송 방식의 특징만이라도 명확히 숙지해 두세요.

**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 프레임링 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 2~3비트씩 추가되므로, 전송 효율이 떨어진다.

**83 [전문가의 조언]**

자주 출제되는 문제는 아닙니다. 이 문제에서는 흐름 제어의 의미만 기억하고 넘어가세요.

**84 경로 선택은 네트워크 계층의 기능입니다.****[전문가의 조언]**

OSI 7계층에서는 특정 계층의 주요 기능을 파악하고 있어야 풀 수 있는 문제가 자주 출제된다고 했죠? 이 문제를 틀렸다면 81번 문제의 [전문가의 조언]을 참조하여 각 계층의 기능을 확실히 기억해 두세요.

**85 [전문가의 조언]**

문제에서는 대역폭을 W 대신 B로 사용했습니다. 통신 용량을 계산하는 식과 더불어 계산식의 구성 요소, 통신 용량을 늘리기 위한 방법 등이 문제로 출제됩니다. 공식을 무조건 암기하려 하지 마세요. 통신 용량을 늘리려면 대역폭(W)과 신호세력(S)은 커져야 하고, 잡음(N)이 줄어야 하는 데, 커져야 하는 것은 분자고 작아져야 하는 것은 분모라는 것을 생각하면 공식을 쉽게 기억할 수 있습니다.

**86 [전문가의 조언]**

데이터 전송 모드와 관련된 문제는 이 문제와 같이 주로 종류를 묻는 문제가 출제됩니다. 데이터 전송 모드의 종류를 기억할 때는 영문 약어도 같이 기억해 두세요. 각각의 개별적인 용도 및 특징은 각각을 구분할 수 있을 정도로만 알아두면 됩니다.

**HDLC의 동작 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 사용한다.

- 중국은 주국의 허가(Poll)가 있을 때에만 송신한다.
- 비동기 응답 모드(ARM, Asynchronous Response Mode)
  - 전이중 통신을 하는 포인트 투 포인트(Point-to-Point) 불균형 링크 구성에 사용한다.
  - 중국은 주국의 허가(Poll) 없이도 송신이 가능하지만, 링크 설정이나 오류 복구 등의 제어 기능은 주국만 한다.
- 비동기 균형(평형) 모드(ABM, Asynchronous Balanced Mode)
  - 포인트 투 포인트(Point-to-Point) 균형 링크에서 사용한다.
  - 혼합국끼리 허가 없이 언제나 전송할 수 있도록 설정한다.

- 87 전송 속도(bps)를 변조 속도(baud)로 변환할 경우에는 전송 속도를 변조 시 상태 변화 수로 나눠서 계산합니다. 4위상이란 위상을 4등분한다는 것으로 각각의 위상에 2비트( $4 = 2^2$ )씩 할당합니다. 즉 변조 속도(baud)는  $9600/2 = 4800$ (baud)입니다.**

**[전문가의 조언]**

bps와 baud의 상호 변환 문제는 자주 출제됩니다. 상호 변환 공식만 알면 쉽게 맞힐 수 있는 문제이니 상호 변환 공식은 반드시 암기하세요.

- 데이터 신호 속도(bps) = 변조 속도(baud) × 변조 시 상태 변화 수

**88 [전문가의 조언]**

처음 출제된 문제입니다. 이 문제에서는 문제로 제시된 OFDM의 개념만 기억하고 넘어가세요.

**89 [전문가의 조언]**

해밍 코드(Hamming Code)의 특징을 묻는 문제가 자주 출제되니 확실히 숙지하세요.

**해밍 코드(Hamming Code)**

- 수신측에서 오류가 발생한 비트를 검출한 후 직접 수정하는 방식이다.
- 오류 검출은 물론 스스로 수정까지 하므로 자기 정정 부호라고도 한다.
- 1비트의 오류만 수정이 가능하며, 정보 비트 외에 잉여 비트가 많이 필요하다.
- 전송 비트 중 1, 2, 4, 8, 16, 32, 64, ...,  $2^n$  번째를 오류 검출을 위한 패리티 비트로 사용하며, 이 비트의 위치는 변하지 않는다.
- 송신한 데이터와 수신한 데이터의 각 대응하는 비트 중 서로 다른 비트의 수를 해밍 거리(Hamming Distance)라고 한다.

**90 IPv6 주소의 길이는 128비트입니다.****[전문가의 조언]**

문제와 보기가 동일하게 출제되었던 문제입니다. IPv6의 특징을 잘 정리하고 넘어가세요.

**IPv6(Internet Protocol version 6)**

- 현재 사용하고 있는 IP 주소 체계인 IPv4의 주소 부족 문제를 해결하기 위해 개발되었다.
- IPv6은 16비트씩 8부분, 총 128비트로 구성되어 있다.
- 각 부분을 16진수로 표현하고, 콜론(:)으로 구분한다.
- 128비트의 긴 주소를 사용하여 주소 부족 문제를 해결할 수 있으며, IPv4에 비해 자료 전송 속도가 빠르다.

- 인증성, 기밀성, 데이터 무결성의 지원으로 보안 문제를 해결할 수 있다.
- IPv4와 호환성이 뛰어나다.
- 주소의 확장성, 융통성, 연동성이 뛰어나며, 실시간 흐름 제어로 향상된 멀티미디어 기능을 지원한다.
- Traffic Class, Flow Label을 이용하여 등급별, 서비스별로 패킷을 구분할 수 있어 품질 보장이 용이하다.

### 91 [전문가의 조언]

자주 출제되는 문제는 아니지만 다시 출제될 것에 대비하여 경로 설정(Routing) 프로토콜의 종류별 특징을 간단히 정리하고 넘어가세요.

#### 라우팅 프로토콜

IGP	하나의 자율 시스템(AS) 내의 라우팅에 사용되는 프로토콜
RIP	<ul style="list-style-type: none"> <li>· 현재 가장 널리 사용되는 라우팅 프로토콜</li> <li>· 소규모 동종의 네트워크(자율 시스템, AS) 내에서 효율적인 방법임</li> <li>· 최대 홉(Hop) 수를 15로 제한하므로, 15 이상의 경우는 도달할 수 없는 네트워크를 의미하며, 이것은 대규모 네트워크에서는 RIP를 사용할 수 없음을 나타냄</li> <li>· 라우팅 정보를 30초마다 네트워크 내의 모든 라우터에 알리며, 180초 이내에 새로운 라우팅 정보가 수신되지 않으면 해당 경로를 이상 상태로 간주함</li> </ul>
OSPF	<ul style="list-style-type: none"> <li>· 경로 수(Hop)에 제한이 없으므로 대규모 네트워크에서 많이 사용되는 라우팅 프로토콜</li> <li>· 라우팅 정보에 변화가 있을 때에, 변화된 정보만 네트워크 내의 모든 라우터에 알림</li> </ul>
EGP	자율 시스템(AS) 간의 라우팅, 즉 게이트웨이 간의 라우팅에 사용되는 프로토콜
BGP	<ul style="list-style-type: none"> <li>· 자율 시스템(AS) 간의 라우팅 프로토콜로, EGP의 단점을 보완하기 위해 만들어짐</li> <li>· 초기에 BGP 라우터들이 연결될 때에는 전체 경로 제어표를 교환하고, 이후에는 변화된 정보만을 교환함</li> </ul>

- 92 10 Base T에서 10은 전송 속도가 10Mbps, Base는 베이스 밴드 방식, T는 전송 매체로 꼬임선(Twisted Pair Wire) 케이블을 사용한다는 의미입니다.

#### [전문가의 조언]

이더넷 시스템의 규격에 대한 문제가 가끔씩 출제되고 있습니다. 다시 출제되더라도 틀리지 않도록 이더넷 시스템의 규격에 대해 정리하고 넘어가세요.

#### 이더넷 시스템의 규격

10 BASE T	10은 전송 속도가 10Mbps, BASE는 베이스 밴드 방식, T는 전송 매체로 꼬임선(Twisted Pair Wire) 케이블을 사용함을 나타냄
10 BASE 2	얇은 동축 케이블을 이용하며, 2는 한 세그먼트의 최장 거리가 약 200m임을 나타냄
10 BASE 5	굵은 동축 케이블을 이용하며, 5는 한 세그먼트의 최장 거리가 500m임을 나타냄
10 BASE F	F는 광섬유 케이블을 이용하는 이더넷임을 나타냄

### 93 [전문가의 조언]

다중화의 의미를 묻는 문제가 자주 출제되고 있습니다. 명확하게 기억해 두세요.

### 94 [전문가의 조언]

펄스 코드 변조(PCM) 방식과 관련해서는 개념보다는 펄스 코드 변조(PCM) 순서를 묻는 문제가 자주 출제됩니다. 순서를 외을 때는 각 단계의 영문 표현까지 같이 외우세요.

#### 펄스 코드 변조(PCM)

펄스 코드 변조 순서 : 송신 측(표본화 → 양자화 → 부호화) → 수신 측(복호화 → 여과화)

표본화 (Sampling)	음성, 영상 등의 연속적인 신호 파형을 일정 시간 간격으로 검출하는 과정
양자화 (Quantizing)	표본화된 PAM 신호를 유한 개의 부호에 대한 대표값으로 조정하는 과정
부호화 (Encoding)	양자화된 PCM 펄스의 진폭 크기를 2진수(1과 0)로 표시하는 과정
복호화 (Decoding)	수신된 디지털 신호, 즉 PCM 신호를 PAM 신호로 되돌리는 과정
여과화 (Filtering)	PAM 신호를 원래의 입력 신호인 아날로그 데이터로 복원하는 과정

### 95 [전문가의 조언]

OSI 7계층과 관련된 문제가 한 회에 3문제나 출제되었네요. 아직도 계층별 주요 기능을 암기하지 못했다면, 81번 문제의 [전문가의 조언]을 참조하여 확실히 기억해 두세요.

### 96 [전문가의 조언]

다원 접속 방식의 종류가 아닌 것을 묻는 문제가 가끔씩 출제되고 있습니다. 다원 접속 방법의 종류를 기억하고 각각의 개념을 간단히 정리해 두세요.

#### 다원 접속 방식

- FDMA(Frequency Division Multiple Access) : 주파수 대역을 일정 간격으로 분할하는 방식
- TDMA(Time Division Multiple Access) : 사용 시간을 분할하는 방식
- CDMA(Code Division Multiple Access) : 주파수나 시간을 모두 공유하면서 각 데이터에 특별한 코드를 부여하는 방식

### 97 [전문가의 조언]

HDLC와 관련해서는 특징, 프레임 구조, 동작 모드 등이 다양하게 자주 출제되고 있습니다. 이 문제를 통해 프레임 구성 요소들의 개별적인 기능을 정리하세요.

#### HDLC 프레임의 구조

플래그 (Flag)	프레임의 시작과 끝을 나타내는 고유한 비트 패턴(01111110)
주소부 (Address Field)	<ul style="list-style-type: none"> <li>• 송·수신국을 식별하기 위해 사용</li> <li>• 불특정 다수에게 전송하는 방송용(Broadcast)은 '11111111', 시스템에 의해 임의로 수신국이 지정되는 시험용(No Station)은 '00000000'을 사용</li> </ul>
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
FCS(Frame Check Sequence Field, 프레임 검사 순서 필드)	프레임 내용에 대한 오류 검출을 위해 사용되는 부분으로, 일반적으로 CRC 코드가 사용됨

- 98 LAN에서 사용되는 매체 액세스 제어(Access Control) 기법에는 CSMA, CSMA/CD(이더넷), 토큰 버스, 토큰 링 등이 있습니다.

**[전문가의 조언]**

자주 출제되는 문제는 아닙니다. LAN에서 사용되는 매체 액세스 제어 기법의 종류만 기억해 두세요.

**99 [전문가의 조언]**

LAN은 주로 정의나 특징을 묻는 문제가 출제됩니다. 잘 정리해 두세요.

**LAN의 특징**

- 광대역 통신망과는 달리 학교, 회사, 연구소 등 한 건물이나 일정 지역 내에서 컴퓨터나 단말기들을 고속 전송 회선으로 연결하여 프로그램 파일 또는 주변장치를 공유할 수 있도록 한 네트워크 형태이다.
- 단일 기관의 소유, 제한된 지역 내의 통신이다.
- 광대역 전송 매체의 사용으로 고속 통신이 가능하다.
- 공유 매체를 사용하므로 경로 선택 없이 매체에 연결된 모든 장치로 데이터를 전송할 수 있다.
- 오류 발생률이 낮으며, 네트워크에 포함된 자원을 공유할 수 있다.
- 네트워크의 확장이나 재배치가 쉽다.
- 전송 매체로 꼬임선, 동축 케이블, 광섬유 케이블 등을 사용한다.
- 망의 구성 형태에 따라서 성형, 버스형, 링형, 계층형(트리형)으로 분류할 수 있다.

**100 [전문가의 조언]**

문제와 보기가 동일하게 출제되었던 문제입니다. 다시 출제되더라도 동일하게 출제될 가능성이 있으니 M진 PSK에서 반송파 간의 위상차는  $\frac{2\pi}{M}$  라는 것만 기억하고 넘어가세요.