



시험에 나오는 것만 공부한다!

**시나공시리즈**

기출문제 & 정답 및 해설  
2021년 1회 정보처리산업기사 필기



## 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 표기하십시오.

### 제1과목 데이터베이스

1. 삽입(Insertion) 정렬을 사용하여 다음의 자료를 오름차순으로 정렬하고자 한다. 2회전 후의 결과는?

5, 4, 3, 2, 1

- ① 4, 5, 3, 2, 1                      ② 2, 3, 4, 5, 1  
③ 3, 4, 5, 2, 1                      ④ 1, 2, 3, 4, 5

2. 애트리뷰트 간에 존재하는 여러 가지 종속 관계를 분석해서 기본적으로 하나의 종속성이 하나의 릴레이션으로 표현되도록 분해하는 과정을 정규화라고 한다. 정규화의 원칙으로 거리가 먼 것은?

- ① 데이터의 종속성이 많아야 한다.  
② 데이터의 중복성이 감소되어야 한다.  
③ 하나의 스키마에서 다른 스키마로 변환시킬 때 정보의 손실이 있어서는 안된다.  
④ 하나의 독립된 관계성은 하나의 독립된 릴레이션으로 분리시켜 표현한다.

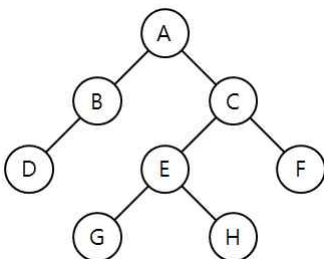
3. 해시에서 서로 다른 두 개 이상의 레코드가 동일한 주소를 갖는 현상을 의미하는 것은?

- ① Synonym                          ② Bucket  
③ Slot                                  ④ Collision

4. 논리적 데이터 모델 중 오너-멤버(Owner-Member) 관계를 가지며, CODASYL DBTG 모델이라고도 하는 것은?

- ① 네트워크 데이터 모델          ② 뷰 데이터 모델  
③ 계층 데이터 모델                  ④ 분산 데이터 모델

5. 다음 그림에서 트리의 차수는?



- ① 1                                      ② 2  
③ 3                                      ④ 8

6. SQL을 정의, 조작, 제어문으로 구분할 경우, 다음 중 나머지 셋과 성격이 다른 것은?

- ① UPDATE                              ② DROP  
③ SELECT                                ④ DELETE

7. 다음의 전위(Prefix) 표기식을 중위(Infix) 표기식으로 옮겨 변환한 것은?

- + \* A B C / D E

- ① B \* D + A - E / C                  ② C \* D + B - A / E  
③ E \* D + C - B / A                  ④ A \* B + C - D / E

8. 다음 중 선형 구조에 해당하지 않는 것은?

- ① 큐(Queue)                              ② 배열(Array)  
③ 그래프(Graph)                        ④ 스택(Stack)

9. 뷰(VIEW)에 대한 설명으로 옳지 않은 것은?

- ① 실제 저장된 데이터 중에서 사용자가 필요한 내용만을 선별해서 볼 수 있다.  
② 실제로는 존재하지 않는 가상의 테이블이다.  
③ 뷰로 구성된 내용에 대하여 삽입, 삭제, 갱신 연산에 제약이 없다.  
④ 데이터 접근 제어로 보안을 제공한다.

10. This is a linear list for which all insertions and deletions, and usually all accesses, are made at one end of the list. What is this?

- ① Graph                                      ② Queue  
③ Stack                                        ④ Array

11. 데이터베이스 설계 단계 중 물리적 설계 단계와 거리가 먼 것은?

- ① 접근 경로 설계  
② 저장 레코드 양식 설계  
③ 레코드 집중의 분석 및 설계  
④ 트랜잭션 모델링

12. 다음 문장을 만족하는 SQL 문장은?

학번이 1000번인 학생을 학생 테이블에서 삭제하십시오.

- ① DELETE FROM 학생 WHERE 학번 = 1000;  
② DELETE FROM 학생 IF 학번 = 1000;  
③ SELECT \* FROM 학생 WHERE 학번 = 1000;  
④ SELECT \* FROM 학생 CONDITION 학번 = 1000;

13. 순수 관계 연산자 중 Select 연산의 연산자 기호는?

- ①  $\pi$ (pi)                                      ②  $\nabla$   
③  $\cup$     ④  $\sigma$ (sigma)

14. 시스템 카탈로그에 대한 설명으로 옳지 않은 것은?

- ① 데이터베이스에 대한 통계정보가 저장될 수 있다.  
② 무결성 확보를 위하여 일반 사용자는 내용을 검색할 수 없다.  
③ 기본 테이블, 뷰, 인덱스, 패키지, 접근 권한 등의 정보를 저장한다.  
④ 시스템 자신이 필요로 하는 스키마 및 여러 가지 객체에 관한 정보를 포함하고 있는 시스템 데이터베이스이다.

15. SQL의 조작성 유형으로 옳지 않은 것은?

- ① SELECT ~ FROM ~ WHERE ~
- ② UPDATE ~ SET ~ WHERE ~
- ③ INSERT ~ FROM ~ SET ~
- ④ DELETE ~ FROM ~ WHERE ~

16. E-R 다이어그램에서 개체를 의미하는 기호는?

- ① 오각형                      ② 타원  
③ 삼각형                    ④ 사각형

17.  $n$ 개의 정점으로 구성된 무방향 그래프의 최대 간선수는?

- ①  $n(n+1)$                       ②  $\frac{n(n-1)}{2}$   
 ③  $\frac{n-2}{2}$                           ④  $n-5$

18. 한 릴레이션의 기본키를 구성하는 어떠한 속성 값도 널(Null) 값이나 중복 값을 가질 수 없음을 의미하는 관계 데이터 모델의 제약 조건은?

- ① 개체 무결성                      ② 외래키 무결성  
③ 릴레이션 무결성                ④ 참조 무결성

19. **트랜잭션의 특성 중 트랜잭션 내의 모든 연산은 반드시 한꺼번에 완료되어야 하며, 그렇지 못한 경우는 한꺼번에 취소되어야 한다는 것은?**

- ① Consistency
- ② Atomicity
- ③ Isolation
- ④ Durability

20. 순서가 A, C, B, D로 정해진 입력 자료를 스택에 입력하였다가 출력한 결과가 될 수 없는 것은? (단, 보기 항에서 좌측 값부터 먼저 출력된 순서이다.)

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

## 제2과목 전자계산기구조

21. 산술연산과 논리연산 동작을 수행한 후 결과를 축적하는 레지스터는?

- ① 누산기                      ② 인덱스 레지스터  
③ 플래그 레지스터        ④ RAM

22. 보조기억장치의 페이지 접근 횟수가 많아 작업 수행 시간보다 페이지 교체 시간이 많아지는 기억공간의 관리 기법은?

- ① 분산 로딩 기법                      ② 페이징(Paging)  
③ 세그먼트                              ④ 연속 로딩 기법

23. 다음의 실행 주기(Execution Cycle)는 어떤 명령을 나타내는 것인가?

```

MAR ← MBR(AD)
MBR ← M
MBR ← MBR + 1
M ← MBR, if(MBR=0) then (PC ← PC+1)

```

- ① JMP                      ② AND  
③ ISZ                     ④ BSA

24. 채널(Channel)을 설명한 것으로 틀린 것은?

- ① CPU의 Idle Time을 줄인다.

- ② I/O 속도를 향상시킨다.
- ③ MODEM의 기능을 갖는다.
- ④ 고속 방식과 저속 방식의 채널이 있다.

25. 다음 ( ) 안에 알맞은 것은? (단, NOT은 고려하지 않는다)

“3×8 Decoder는 (        ) 회로 8개로 만들 수 있다.”

- ① NOR                      ② OR  
③ NAND                  ④ AND

26. 인터럽트 발생 시 프로세스의 상태 보존의 필요성을 가장 옳게 설명한 것은?

- ① 인터럽트를 요청한 해당 장치에 대한 인터럽트 서비스를 완료하고 원래 수행 중이던 프로그램으로 복귀하기 위해
- ② 인터럽트 처리 속도를 향상시키기 위해
- ③ 인터럽트 발생 횟수를 카운트하고 일정 횟수 이상이 되면 시스템을 정지시키기 위해
- ④ 인터럽트 요청 장치와 그 장치의 우선순위를 파악하기 위해

27. 명령어의 Op-Code(명령 코드)는 어느 레지스터에서 이용하는가?

- ① Flag Register
- ② Index Register
- ③ Address Register
- ④ Instruction Register

28. 십진수 6을 4Bit Excess-3 코드로 변환한 후 Gray 코드로 표현한 것은?

- ① 0110                  ② 1101  
③ 1100                  ④ 1001

29. 컴퓨터에서 사용되는 보조기억장치의 특징이 아닌 것은?

- ① 대용량 기억장치이다.
- ② 주기억장치보다 액세스 속도가 빠르다.
- ③ 대형 프로그램을 기억시킬 수 있다.
- ④ 주기억장치보다 비트당 가격이 싸다.

30. 다음 중 입력장치가 아닌 것은?

- ① Scanner                      ② Mouse  
③ Line Printer                ④ Keyboard

31. JK 플립플롭에서  $J_n = 0$ ,  $K_n = 0$ 인 경우의 출력  $Q_{n+1}$ 은?

- ① 0                  ② 1                  ③  $\mathbb{Q}_n$                   ④ 부정

32. 한 명령의 실행 사이클 중에 인터럽트 요청에 의해 인터럽트를 처리한 후 CPU가 다음에 수행하는 사이클은?

- ① Fetch Cycle                      ② Indirect Cycle  
③ Execute Cycle                  ④ Direct Cycle

33. 단항(Unary) 연산을 행하는 것은?

- ① SHIFT                      ② AND  
③ OR                         ④ 사칙 연산

34. 하드웨어의 특성상 주기억장치가 제공할 수 있는 정보전달의 능력 한계를 무엇이라 하는가?

- ① 주기억장치 용량폭                      ② 주기억장치 대역폭  
③ 주기억장치 접근폭                      ④ 주기억장치 전달폭

35. 다음 중 CISC(Complex Instruction Set Computer)형 프로세서의 특징이 아닌 것은?

- ① 명령어의 길이가 일정하다.
- ② 많은 수의 명령어를 갖는다.

- ③ 다양한 주소 모드를 지원한다.  
④ 레지스터와 메모리의 다양한 명령어를 제공한다.

36. 그림과 같은 전가산기(Full Adder)의 입력이 A=1, B=0, C=1일 때 출력 So(합)와 Co(캐리)는?



- ① Co = 0, So = 0                      ② Co = 0, So = 1  
③ Co = 1, So = 0                      ④ Co = 1, So = 1

37. 다음 설명 중 틀린 것은?

- ① 중앙처리장치에서 연산한 결과 등을 일시적으로 저장해 두는 레지스터를 누산기라 한다.  
② 입출력장치는 주변장치에 해당된다.  
③ 레지스터에서 기억장치로 정보를 옮기는 것을 로드(load)라 한다.  
④ 기억장치내의 데이터를 다른 기억장치로 옮기는 것을 전송이라 한다.

38. 입·출력 프로그램의 목적과 가장 거리가 먼 것은?

- ① CPU의 Loading  
② CPU와 I/O의 통신  
③ Interrupt 처리  
④ I/O 장치의 구동

39. 비수치 연산에 속하지 않은 것은?

- ① 논리적 연산  
② 로테이트(Rotate)  
③ 사칙 연산  
④ 시프트(Shift)

40. 여러 개의 CPU(중앙처리장치)를 가지고 동시에 다수 작업을 처리하는 개념은?

- ① Multiprocessing                      ② Multiprogramming  
③ Multiaccessing                      ④ Multitasking

### 제3과목 시스템 분석 및 설계

41. 시스템 개발에 대한 문서화의 효과로 거리가 먼 것은?

- ① 시스템 개발 후 유지보수가 용이하다.  
② 시스템 개발팀에서 운용팀으로 인수인계가 용이하다.  
③ 시스템 개발의 요식적 절차를 부각시킬 수 있다.  
④ 시스템 개발 요령 및 순서를 표준화 할 수 있다.

42. 코드 설계 단계 중 대상 항목에 대하여 설계된 코드의 사용이 컴퓨터 처리 내에 한정되는가, 해당 업무에만 한정되는가, 관련 부분의 업무에 공통으로 사용되는가, 기업 전체에 사용되는가 등을 확정하는 단계는?

- ① 사용 기간의 결정  
② 코드화 방식의 결정  
③ 사용 범위의 결정  
④ 코드 대상 항목 결정

43. 프로세스의 표준 처리 패턴 중 특정의 조건을 제시하여 그 조건에 부합되는 데이터를 파일 중에서 추출해 내는 처리로서, 정보 검색을 위한 필수적인 기능인 것은?

- ① Conversion                      ② Merge  
③ Extract                      ④ Matching

44. 오류 검사의 종류 중 산술 연산시 "0(zero)"으로 나눈 경우의 여부를 검사하는 것은?

- ① Impossible Check  
② Sign Check  
③ Overflow Check  
④ Unmatched Record Check

45. 자료 사전에서 자료의 연결(And)시 사용하는 기호는?

- ① =                      ② { }  
③ ( )                      ④ +

46. 코드 작성 시 유의사항으로 적합하지 않은 것은?

- ① 공통성이 있어야 한다.  
② 복잡성이 있어야 한다.  
③ 체계성이 있어야 한다.  
④ 확장성이 있어야 한다.

47. 객체 지향의 개념에서 하나 이상의 유사한 객체를 묶어서 하나의 공통된 특성을 표현한 것을 무엇이라고 하는가?

- ① 인스턴스                      ② 메소드  
③ 메시지                      ④ 클래스

48. 시간의 흐름에 따른 시스템의 변화상을 보여주는 상태 다이어그램을 작성하는 모형화 단계는?

- ① 객체 모형화(Object Modeling)  
② 동적 모형화(Dynamic Modeling)  
③ 기능 모형화(Function Modeling)  
④ 정적 모형화(Static Modeling)

49. 다음과 같은 특징을 갖는 출력 매체 시스템은?

- 축소 보관과 반영구적인 매체로 사용 가능
- 처리 결과를 마이크로 필름에 기록
- 지도, 설계도면, 학적부, 병원 기록의 보존, 검색, 관리 등에 적합

- ① CRT 출력 시스템                      ② COM 시스템  
③ X-Y 플로터                      ④ 음성 출력 시스템

50. 코드화 대상 항목을 10진 분할하고, 코드 대상 항목의 추가가 용이하며, 무제한적으로 확대할 수 있으나 자리수가 길어질 수 있고, 기계처리에는 적합하지 않은 코드는?

- ① Block Code  
② Decimal Code  
③ Group Classification Code  
④ Sequence Code

51. 파일 설계 단계 중 다음 사항과 관계되는 것은?

- 처리 주기 및 처리 방식
- 갱신 빈도와 형태(추가, 삭제, 수정 등)
- 파일의 활동을

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

52. IPT 기법의 적용 목적으로 가장 거리가 먼 것은?

- ① 개발자의 생산성 향상  
② 프로그래밍의 표준화 유도  
③ 효율적이고 신뢰성 높은 프로그램 개발  
④ 프로그래머 층원 용이

53. 프로세서 설계에 필요한 흐름도 종류 중 처리 내용, 조건, 입출력 데이터의 종류와 출력 등을 논리적으로 표현한 흐름도는?

- ① 블록 차트                      ② 시스템 흐름도  
③ 프로세서 흐름도              ④ 프로그램 흐름도

54. 시스템의 기본 요소 중 처리결과를 평가하여 불충분한 경우 목적 달성을 위해 반복 처리하는 요소는?

- ① feedback                      ② input  
③ output                        ④ process

55. 표준 처리 패턴 중 동일한 파일형식을 가지고 있는 두 개 이상의 파일을 하나의 파일로 통합 처리하는 패턴을 무엇이라고 하는가?

- ① 대조(Match) 패턴  
② 병합(Merge) 패턴  
③ 갱신(Update) 패턴  
④ 생성(Generate) 패턴

56. 시스템의 특성 중 시스템이 정의된 기능을 오류가 없이 정확히 발휘하기 위해 정해진 규정이나 한계, 또는 궤도로부터 이탈되는 사태나 현상을 미리 인식하여 그것을 올바르게 수정해 가는 것을 의미하는 것은?

- ① 목적성                        ② 자동성  
③ 제어성                        ④ 종합성

57. 코드의 기능 중 다음이 설명하는 것은?

정보에 대하여 특정 기준을 만족하는 그룹과 만족하지 않는 그룹으로 나누거나, 코드 대상이 되는 동일 특성을 가진 데이터를 그룹화 하는 기능

- ① 표준화 기능                      ② 분류 기능  
③ 식별 기능                        ④ 연산 기능

58. 시스템을 평가하는 목적으로 거리가 먼 것은?

- ① 시스템 운영 관리의 타당성 파악  
② 시스템의 성능과 유용도 판단  
③ 처리 비용과 효율 면에서 개선점 파악  
④ 시스템 운영 요원의 재훈련

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. LOC 기법에 의해 예측된 모듈의 라인수가 100000 라인이고 개발에 투입되는 프로그래머의 수가 4명, 프로그래머의 월 평균 생산량이 1000 라인이라고 할 때, 이 소프트웨어를 완성하기 위해 개발에 필요한 기간은?

- ① 10개월                        ② 15개월  
③ 20개월                        ④ 25개월

#### 제4과목 운영체제

61. 프로세스의 정의와 거리가 먼 것은?

- ① 하드웨어에 의해 사용되는 입출력 장치  
② 동시에 실행될 수 있는 프로그램들의 집합  
③ 프로그램이 활성화 된 상태  
④ 프로세서에 할당되어 실행될 수 있는 개체

62. UNIX에 대한 설명으로 옳지 않은 것은?

- ① 대부분 C 언어로 작성되어 이식성이 높다.  
② 대화식 시분할 운영체제이다.  
③ 멀티유저, 멀티태스킹을 지원한다.  
④ 리스트 구조의 파일 시스템을 갖는다.

63. 교착상태(Deadlock)의 필요조건에 해당하지 않는 것은?

- ① Circular Wait                      ② Preemption  
③ Hold and Wait                      ④ Mutual Exclusion

64. 페이지 부재(Page Fault)가 계속 발생되어 프로세스가 수행되는 시간보다 페이지 교체에 소비되는 시간이 더 많은 경우를 무엇이라고 하는가?

- ① 스래싱(Thrashing)                      ② 오버레이(Overlay)  
③ 세마포어(Semaphore)                      ④ 페이징(Paging)

65. 분산 처리 시스템의 위상(Topology)에 따른 분류에서 성형(Star) 구조에 대한 설명으로 가장 적합하지 않은 것은?

- ① 사이트의 증가에 따라 통신 회선도 증가한다.  
② 중앙 사이트의 고장 시 모든 통신이 단절된다.  
③ 집중제어로 보수와 관리가 용이하다.  
④ 각 노드들은 Point-to-Point 형태로 모든 노드들과 직접 연결된다.

66. UNIX에서 파일 내용을 화면에 표시하는 명령은?

- ① Chmod                        ② ps  
③ cat                            ④ ls

67. 13K의 작업을 다음 그림의 30K 공백의 작업 공간에 할당했을 경우 사용된 기억장치 배치 전략 기법은?

OS
16K 공백
used
14K 공백
used
5K 공백
used
30K 공백

- ① Last-Fit                        ② First-Fit  
③ Best-Fit                        ④ Worst-Fit

68. 디스크에서 헤드가 70 트랙을 처리하고 60 트랙으로 이동해 왔다. 디스크 스케줄링 기법으로 SCAN 방식을 사용할 때 다음 디스크 대기 큐에서 가장 먼저 처리되는 트랙은?

디스크 대기 큐 : 20, 50, 95, 100

- ① 20                            ② 50  
③ 95                            ④ 100

69. 모니터에 대한 설명으로 옳지 않은 것은?

- ① 모니터 외부의 프로세스는 모니터 내부의 데이터를 직접 액세스 할 수 없다.  
② 모니터에서는 wait와 signal 연산이 사용된다.  
③ 자료 추상화와 정보 은폐 기법을 기초로 한다.  
④ 한 순간에 여러 프로세스가 모니터에 동시에 진입하여 자원을 공유할 수 있다.



## 70. 운영체제의 목적으로 옳지 않은 것은?

- ① 응답 시간 증가
- ② 사용자 인터페이스 제공
- ③ 신뢰성 향상
- ④ 처리량의 향상

## 71. Round-Robin 스케줄링(Scheduling) 방식에 대한 설명으로 옳지 않은 것은?

- ① 시간 할당량이 작아질수록 문맥교환 과부하는 상대적으로 낮아진다.
- ② 적절한 응답시간이 보장되므로 시분할 시스템에 유용하다.
- ③ 할당된 시간(Time Slice) 내에 작업이 끝나지 않으면 대기 큐의 맨 뒤로 그 작업을 배치한다.
- ④ 시간 할당량이 충분히 크면 FIFO 방식과 비슷하다.

## 72. 다음 프로세스에 대하여 HRN 기법으로 스케줄링 할 경우 우선 순위로 옳은 것은?

프로세스	실행(서비스) 시간	대기 시간
A	10	30
B	6	12
C	12	12

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

## 73. 페이지 교체 기법 중 시간 오버헤드를 줄이는 기법으로서 참조 비트(Referenced Bit)와 변형 비트(Modified Bit)를 필요로 하는 방법은?

- ① LRU
- ② NUR
- ③ FIFO
- ④ LFU

## 74. 강결합(Tightly-Coupled) 시스템과 약결합(Loosely-Coupled) 시스템에 대한 설명으로 옳지 않은 것은?

- ① 강결합 시스템은 하나의 운영체제가 모든 처리기와 시스템 하드웨어를 제어한다.
- ② 약결합 시스템은 메시지를 사용하여 상호 통신을 한다.
- ③ 약결합 시스템은 각각의 시스템이 별도의 운영체제를 가진다.
- ④ 강결합 시스템은 각 프로세서마다 독립된 메모리를 가진다.

## 75. 다중 처리기 운영체제 구조 중 주종(Master/Slave) 처리기에 대한 설명으로 옳지 않은 것은?

- ① 주프로세서가 입출력을 수행하므로 비대칭 구조를 갖는다.
- ② 종프로세서는 입출력 발생 시 주프로세서에게 서비스를 요청한다.
- ③ 주프로세서의 고장시 전체 시스템이 멈춘다.
- ④ 주프로세서는 연산만 수행하고 종프로세서는 입출력과 연산을 수행한다.

## 76. 3 페이지가 들어갈 수 있는 기억장치에서 다음과 같은 순서로 페이지가 참조될 때 FIFO 기법을 사용하면 최종적으로 기억공간에 남는 페이지들로 옳은 것은? (단, 현재 기억공간은 모두 비어 있다고 가정한다.)

참조열 : 1, 2, 3, 4, 1, 3, 1

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

## 77. 스케줄링에 대한 설명으로 틀린 것은?

- ① 오버헤드를 최소화시켜야 한다.
- ② 무한 연기는 회피해야 한다.
- ③ 모든 프로세스에게 공정한 적용을 위해 우선 순위는 불필요하다.
- ④ 모든 작업에 대해 공평성을 유지해야 한다.

## 78. SJF(Shortest Job First) 스케줄링에서 작업 도착 시간과 CPU 사용 시간은 다음 표와 같다. 모든 작업들의 평균 대기 시간은 얼마인가?

작업	도착 시간	CPU 사용시간
1	0	20
2	3	32
3	8	7

- ① 15
- ② 12
- ③ 9
- ④ 6

## 79. 교착 상태의 해결 방법 중 Banker's Algorithm과 관계되는 것은?

- ① Recovery
- ② Avoidance
- ③ Prevention
- ④ Detection

## 80. UNIX에서 커널의 기능이 아닌 것은?

- ① 명령어 해독 기능
- ② 입·출력 관리 기능
- ③ 프로세스 관리 기능
- ④ 기억장치 관리 기능

## 제5과목 : 정보 통신 개론

## 81. 전송시간을 일정한 간격의 시간 슬롯(time slot)으로 나누고, 이를 주기적으로 각 채널에 할당하는 다중화 방식은?

- ① 주파수 분할 다중화
- ② 파장 분할 다중화
- ③ 통계적 시분할 다중화
- ④ 동기식 시분할 다중화

## 82. 다음 중 교환 방식에 관한 설명으로 틀린 것은?

- ① 회선 교환 방식은 회선에 융통성이 요구되거나 메시지가 짧은 경우에 적합하다.
- ② 데이터그램 패킷 교환 방식은 부하가 적거나 간헐적인 통신의 경우에 적합하다.
- ③ 패킷 교환 방식은 코드 및 속도 변환이 가능하다.
- ④ 가상회선 패킷 교환 방식은 패킷 도착순서가 고정적이다.

## 83. 컴퓨터의 물리적 자원들이 한 건물 내에 산재해 있을 때 정보자원의 공유를 가능하게 해주는 통신망으로 가장 적합한 것은?

- ① LAN
- ② VAN
- ③ WAN
- ④ ISDN

## 84. 나이퀴스트(Nyquist) Sampling Theorem과 관련이 있는 것은?

- ① 표본화
- ② 양자화
- ③ 부호화
- ④ 복호화

## 85. 다중접속 방식이 아닌 것은?

- ① FDMA
- ② TDMA
- ③ CDMA
- ④ XDMA



## 정답 및 해설

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

- 1 삽입 정렬은 두 번째 자료부터 시작하여 그 앞(왼쪽)의 자료들과 비교하여 삽입할 위치를 지정한 후 자료를 뒤로 옮기고 지정한 자리에 자료를 삽입하여 정렬하는 알고리즘입니다. 즉 두 번째 자료는 첫 번째 자료, 세 번째 자료는 두 번째와 첫 번째 자료, 네 번째 자료는 세 번째, 두 번째, 첫 번째 자료와 비교한 후 자료가 삽입될 위치를 찾습니다.

초기 자료 : 

5	4	3	2	1
---	---	---	---	---

① 1회전 : 

5	4	3	2	1
---	---	---	---	---

 → 

4	5	3	2	1
---	---	---	---	---

두 번째 값 4를 첫 번째 값과 비교하여 첫 번째 자리에 삽입하고 5를 한 칸 뒤로 이동시킵니다.

② 2회전 : 

4	5	3	2	1
---	---	---	---	---

 → 

3	4	5	2	1
---	---	---	---	---

세 번째 값 3을 첫 번째, 두 번째 값과 비교하여 4자리에 삽입하고 4, 5는 한 칸씩 뒤로 이동시킵니다.

③ 3회전 : 

3	4	5	2	1
---	---	---	---	---

 → 

2	3	4	5	1
---	---	---	---	---

네 번째 값 2를 첫 번째, 두 번째, 세 번째 값과 비교하여 3자리에 삽입하고 3, 4, 5는 한 칸씩 뒤로 이동시킵니다.

④ 4회전 : 

2	3	4	5	1
---	---	---	---	---

 → 

1	2	3	4	5
---	---	---	---	---

다섯 번째 값 1을 처음부터 비교하여 2자리에 삽입하고 나머지를 한 칸씩 뒤로 이동시킵니다.

#### [전문가의 조언]

자주 출제되는 내용입니다. 삽입 정렬 과정을 정확하게 파악하고 넘어가세요.

- 2 정규화의 원칙 중 하나는 데이터의 종속성을 제거하여 독립성을 높이는 것입니다.

#### [전문가의 조언]

자주 출제되는 내용입니다. 정규화의 개념과 목적을 정리하세요.

#### 정규화의 개념

- 정규화란 함수적 종속성 등의 종속성 이론을 이용하여 잘못된 설계된 관계형 스키마를 더 작은 속성의 세트로 쪼개어 바람직한 스키마로 만들어 가는 과정이다.
- 정규형에는 제1정규형, 제2정규형, 제3정규형, BCNF형, 제4정규형, 제5정규형이 있으며, 차수가 높아질수록 만족시켜야 할 제약 조건이 늘어난다.
- 정규화는 데이터베이스의 논리적 설계 단계에서 수행한다.
- 정규화는 논리적 처리 및 품질에 큰 영향을 미친다.

#### 정규화의 목적

- 데이터 구조의 안정성을 최대화한다.

- 어떠한 릴레이션이라도 데이터베이스 내에서 표현 가능하게 만든다.
- 효과적인 검색 알고리즘을 생성할 수 있다.
- 중복을 배제하여 삽입, 삭제, 갱신 이상의 발생을 방지한다.
- 데이터 삽입 시 릴레이션을 재구성할 필요성을 줄인다.

- 3 Collision과 Synonym을 혼동하지 마세요. 해싱 결과 서로 다른 두 개 이상의 레코드가 같은 주소를 갖는 현상은 Collision이고, Collision으로 인해 같은 주소에 저장된 서로 다른 두 개 이상의 레코드 집합을 Synonym이라고 합니다.

#### [전문가의 조언]

종종 출제되는 내용입니다. 해싱의 개요 그리고 해시 테이블과 관련된 용어를 정리하고 넘어가세요.

#### 해싱(Hashing)

- 해싱은 해시 테이블(Hash Table)이라는 기억 공간을 할당하고, 해시 함수(Hash Function)를 이용하여 레코드 키에 대한 해시 테이블 내의 Home Address를 계산한 후 주어진 레코드를 해당 기억 장소에 저장하거나 검색 작업을 수행하는 방식이다.
- 해싱은 DAM(직접 접근) 파일을 구성할 때 사용되며, 접근 속도는 빠르나 기억 공간이 많이 요구된다.
- 검색 속도가 가장 빠르다.
- 삽입, 삭제 작업의 빈도가 많을 때 유리한 방식이다.

#### 해시 테이블(Hash Table, 해시 표)

버킷(Bucket)	하나의 주소를 갖는 파일의 한 구역을 의미하며, 버킷의 크기는 같은 주소에 포함될 수 있는 레코드 수를 의미함
슬롯(Slot)	1개의 레코드를 저장할 수 있는 공간으로, n개의 슬롯이 모여 하나의 버킷을 형성함
Collision (충돌 현상)	서로 다른 2개 이상의 레코드가 같은 주소를 갖는 현상
Synonym	같은 Home Address를 갖는 레코드들의 집합
Overflow	계산된 Home Address의 Bucket 내에 저장할 기억 공간이 없는 상태로 Bucket을 구성하는 Slot이 여러 개일 때 Collision은 발생해도 Overflow는 발생하지 않을 수 있음

#### 4 [전문가의 조언]

네트워크(망형) 데이터 모델의 개념 및 관련 용어의 의미를 묻는 문제가 종종 출제되니 정리해 두세요.

**망형 데이터 모델(Network Data Model)의 개요**

- CODASYL이 제안한 것으로, CODASYL DBTG 모델이라고도 한다.
- 그래프를 이용해서 데이터 논리 구조를 표현한 데이터 모델이다.
- 상위와 하위 레코드 사이에서 다 대 다(N:M) 대응관계를 만족하는 구조이다.
- 레코드 타입간의 관계는 1:1, 1:N, N:M이 될 수 있다.
- 대표적 DBMS : DBTG, EDBS, TOTAL

**망형 데이터 모델의 표현**

- Entity군 : 동종의 Entity 그룹
- Entity SET : 주종 관계에 있는 Entity군들의 그룹
- SET Membership Type : 일 대 다(1:N) 관계에 연관된 레코드 타입들을 각각 오너(Owner), 멤버(Member)라고 함
- ※ 오너(Owner)는 트리 구조에서의 Parent, 멤버(Member)는 트리 구조에서의 Children과 같은 개념임

- 5 트리(Tree)의 차수(Degree)는 가장 차수가 많은 노드의 차수입니다. 문제에 주어진 트리(Tree)에서 각 노드의 차수는 A=2, B=1, C=2, E=2입니다. 곧 A, C, E 노드의 차수가 2로 가장 크면서 같으므로 트리의 차수는 2입니다.

**[전문가의 조언]**

종종 출제되는 내용입니다. 트리의 차수와 노드의 차수를 구분해서 기억해 두세요.

- 6 DROP만 DDL이고 나머지는 DML입니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. 각 언어의 특징을 파악하고 각각에는 어떤 명령들이 있는지 구분할 수 있도록 공부하세요.

**데이터베이스 언어**

- DDL(Data Define Language, 데이터 정의어)
  - SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의하거나 변경 또는 삭제할 때 사용하는 언어이다.
  - 데이터베이스 관리자나 데이터베이스 설계자가 사용한다.
  - 명령어 : CREATE, ALTER, DROP
- DML(Data Manipulation Language, 데이터 조작어)
  - 데이터베이스 사용자가 응용 프로그램이나 질의어를 통하여 저장된 데이터를 실질적으로 처리하는 데 사용되는 언어이다.
  - 데이터베이스 사용자와 데이터베이스 관리 시스템 간의 인터페이스를 제공한다.
  - 명령어 : SELECT, INSERT, DELETE, UPDATE
- DCL(Data Control Language, 데이터 제어어)
  - 데이터의 보안, 무결성, 회복, 병행 수행 제어 등을 정의하는 데 사용되는 언어이다.
  - 데이터베이스 관리자가 데이터 관리를 목적으로 사용한다.
  - 명령어 : COMMIT, ROLLBACK, GRANT, REVOKE

**7 [전문가의 조언]**

Prefix(전위) 표기란 연산자가 해당 피연산자 2개의 앞(왼쪽)에 표기되어 있는 것을 말합니다. 그러므로 인접한 피연산자 2개와 왼쪽으로 인접한 연산자를 묶은 후 연산자를 피연산자 사이에 옮겨놓으면 됩니다.

- ① 피연산자 2개와 왼쪽으로 인접한 연산자 1개를 묶습니다.

(- (+ (\* A B) C) (/ D E))

- ② 연산자를 피연산자 사이로 이동시킵니다.

→ (((A \* B) + C) - (D / E))

- ③ 불필요한 괄호를 제거합니다.

((A \* B) + C) - (D / E) → A \* B + C - D / E

- 8 그래프(Graph)는 비선형 자료 구조입니다.

**[전문가의 조언]**

**중요해요!** 자료 구조를 선형 구조와 비선형 구조로 구분할 수 있어야 합니다.

**자료 구조의 분류**

비선형 구조	트리(Tree), 그래프(Graph)
선형 구조	리스트(List), 스택(Stack), 큐(Queue), 데크(Deque)

- 9 뷰는 기본 테이블의 기본키를 포함한 속성(열) 집합으로 뷰를 구성해야만 삽입, 삭제, 갱신 연산이 가능하므로 삽입, 삭제, 갱신 연산에는 제한이 있다고 할 수 있습니다.

**[전문가의 조언]**

**중요해요!** 뷰의 의미와 특징을 묻는 문제가 자주 출제되고 있습니다. 확실히 정리하고 넘어가세요.

**뷰(View)의 개념**

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

- 10 모든 삽입과 삭제 그리고 일반적으로 모든 접근이 리스트의 한쪽 끝에서 이루어지는 선형 리스트는 스택(Stack)입니다.

**[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
- 스택의 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

11 트랜잭션 모델링은 개념적 설계 단계에서 수행합니다.

**[전문가의 조언]**

물리적 설계에서는 물리적 데이터베이스의 특징이나 물리적 설계 시 고려할 사항도 종종 시험에 출제되니 한 번 확인하고 넘어가세요.

**물리적 설계(데이터 구조화)**

- 논리적 설계 단계에서 논리적 구조로 표현된 데이터를 디스크 등의 물리적 저장장치에 저장할 수 있는 물리적 구조의 데이터로 변환하는 과정이다.
- 물리적 설계 단계에서는 다양한 데이터베이스 응용에 대해서 처리 성능을 얻기 위해 데이터베이스 파일의 저장 구조 및 액세스 경로를 결정한다.
- 레코드의 형식, 순서, 접근 경로와 같은 정보를 사용하여 데이터가 컴퓨터에 저장되는 방법을 묘사한다.

**물리적 데이터베이스 구조의 특징**

- 기본적인 데이터 단위는 저장 파일이다.
- 데이터베이스 시스템의 성능에 중대한 영향을 미친다.
- 데이터베이스에 포함될 여러 파일 타입에 대한 저장 레코드의 양식, 순서, 접근 경로를 표현한 것이다.

**물리적 설계 시 고려 사항**

- **반응 시간(Response Time)** : 트랜잭션 수행을 요구한 시점부터 처리 결과를 얻을 때까지의 경과 시간
- **공간 활용도(Space Utilization)** : 데이터베이스 파일과 액세스 경로 구조에 의해 사용되는 저장 공간의 양
- **트랜잭션 처리량(Transaction Throughput)** : 단위 시간 동안 데이터베이스 시스템에 의해 처리될 수 있는 트랜잭션의 평균 개수

12 문제의 지문에 제시된 문장을 만족하는 SQL 문은 다음과 같습니다.

- ‘학생’ 테이블에서 삭제하시오. : DELETE FROM 학생
- ‘학번’이 1000번인 학생을 대상으로 하시오. : WHERE 학번 = 1000;

**[전문가의 조언]**

SQL 절의어의 사용법과 관련된 문제가 종종 출제됩니다. 이 문제에서는 DELETE의 의미와 사용법을 알아두세요.

**DELETE**

DELETE문은 기본 테이블에 있는 튜플(행)들 중에서 특정 튜플을 삭제할 때 사용하는 것으로 일반 형식은 다음과 같다.

```
DELETE
FROM 테이블명
WHERE 조건;
```

13 [전문가의 조언]

종종 출제되는 내용입니다. 순수 관계 연산자의 종류 4가지와 함께 각각의 기능을 파악해 두세요.

**순수 관계 연산자**

Select	<ul style="list-style-type: none"> <li>• 릴레이션에 존재하는 튜플중에서 선택 조건을 만족하는 튜플의 부분 집합을 구하여 새로운 릴레이션을 만들</li> <li>• 릴레이션의 행(가로)에 해당하는 튜플을 구하는 것이므로 수평 연산이라고도 함</li> <li>• 연산자 기호 : <math>\sigma</math>(시그마)</li> </ul>
--------	---

Project	<ul style="list-style-type: none"> <li>• 주어진 릴레이션에서 속성 List에 제시된 Attribute만을 추출하는 연산</li> <li>• 릴레이션의 열(세로)에 해당하는 Attribute를 추출하는 것이므로 수직 연산자라고도 함</li> <li>• 연산자 기호 : <math>\pi</math>(파이)</li> </ul>
Join	<ul style="list-style-type: none"> <li>• 공통 속성을 중심으로 2개의 릴레이션을 하나로 합쳐서 새로운 릴레이션을 만드는 연산</li> <li>• 연산자 기호 : <math>\bowtie</math></li> </ul>
Division	<ul style="list-style-type: none"> <li>• <math>X \supset Y</math>인 2개의 릴레이션에서 R(X)와 S(Y)가 있을 때 R의 속성이 S의 속성값을 모두 가진 튜플에서 S가 가진 속성을 제외한 속성만을 구하는 연산</li> <li>• 연산자 기호는 : <math>\div</math></li> </ul>

14 시스템 카탈로그 자체도 테이블(시스템 테이블)로 구성되어 있어 일반 사용자도 SQL을 이용하여 내용을 검색해 볼 수 있습니다. 단, 수정은 불가능합니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. 시스템 카탈로그의 의미와 특징을 꼭 숙지하세요.

**시스템 카탈로그(System Catalog)**

- 시스템 카탈로그는 시스템 그 자체에 관련이 있는 다양한 객체에 관한 정보를 포함하는 시스템 데이터베이스이다.
- 시스템 카탈로그는 데이터베이스에 포함되는 모든 데이터 객체에 대한 정의나 명세에 관한 정보를 유지 관리하는 시스템 테이블이다.
- 데이터 정의어의 결과로 구성되는 기본 테이블, 뷰, 인덱스, 패키지, 접근 권한 등의 데이터베이스 구조 및 통계 정보를 저장한다.
- 카탈로그들이 생성되면 자료 사전(Data Dictionary)에 저장되기 때문에 좁은 의미로는 카탈로그를 자료 사전이라고도 한다.
- 카탈로그에 저장된 정보를 메타 데이터(Meta-Data)라고 한다.
- 시스템 카탈로그 자체도 시스템 테이블로 구성되어 있어 일반 사용자도 SQL을 이용하여 내용을 검색해 볼 수 있다.
- INSERT, DELETE, UPDATE문으로 카탈로그를 갱신하는 것은 허용되지 않는다.
- 카탈로그는 DBMS가 스스로 생성하고, 유지한다.

15 삽입(INSERT)문의 일반 형식은 ‘INSERT ~ INTO ~ VALUE ~’입니다.

**[전문가의 조언]**

종종 출제되는 내용입니다. SQL의 데이터 조작문의 4가지 유형을 정확히 기억해 두세요.

16 [전문가의 조언]

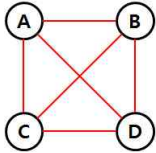
**중요해요!** E-R 모델과 관련해서는 E-R 다이어그램에서 사용하는 도형이 가장 많이 출제됩니다. 이 문제에서 확실하게 기억하고 넘어가세요.

**E-R 도형**

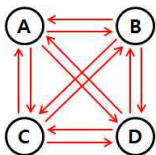
다이아몬드(마름모)	관계(Relationship) 타입
사각형	개체(Entity) 타입
타원	속성(Attribute)
밑줄 타원	기본 키 속성
선, 링크	개체 타입과 속성을 연결

- 17 그래프는 정점(Vertex)과 간선(Edge)의 두 집합으로 이루어진 것으로, 간선의 방향성 유무에 따라 방향 그래프와 무방향 그래프가 있습니다. n개의 정점으로 구성된 무방향 그래프에서 최대 간선 수는  $n(n-1)/2$ 이고, 방향 그래프에서 최대 간선 수는  $n(n-1)$ 입니다. 실제 그래프의 예를 보면 쉽게 이해할 수 있습니다.

예) 정점이 4개인 경우 무방향 그래프와 방향 그래프의 최대 간선 수는 다음과 같습니다.



• 무방향 그래프의 최대 간선 수 :  $4(4-1)/2 = 6$



• 방향 그래프의 최대 간선 수 :  $4(4-1) = 12$

#### [전문가의 조언]

자주 출제되는 문제는 아닙니다. 무방향 그래프와 방향 그래프의 최대 간선 수를 구하는 방법만 간단히 알아두세요.

#### 18 [전문가의 조언]

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

##### 참조 무결성

외래키 값은 NULL이거나 참조 릴레이션의 기본키 값과 동일해야 한다. 즉 릴레이션은 참조할 수 없는 외래키 값을 가질 수 없다.

#### 19 [전문가의 조언]

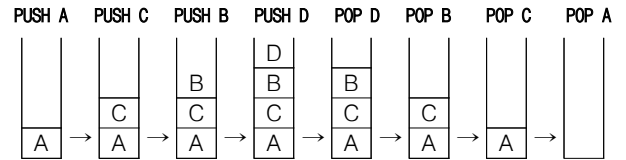
정보처리기사 범위에 포함되는 내용인데 산업기사에도 종종 출제되고 있습니다. 트랜잭션의 특징은 번역을 달리하여 출제되는 경우가 있으므로 영어로도 알아둬야 하며, 4가지 종류는 물론 각각의 의미까지도 꼭 숙지해야 합니다.

##### 트랜잭션의 특징

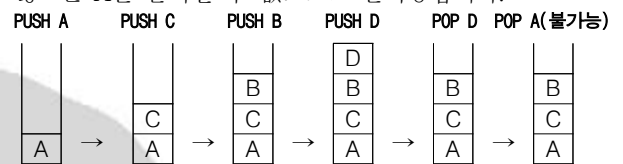
Atomicity (원자성)	<ul style="list-style-type: none"> <li>트랜잭션의 연산은 데이터베이스에 모두 반영되었는지 아니면 전혀 반영되지 않아야 함</li> <li>트랜잭션 내의 모든 명령은 반드시 완벽히 수행되어야 하며, 모두가 완벽히 수행되지 않고 어느 하나라도 오류가 발생하면 트랜잭션 전부가 취소되어야 함</li> </ul>
Consistency (일관성)	<ul style="list-style-type: none"> <li>트랜잭션이 그 실행을 성공적으로 완료하면 언제나 일관성 있는 데이터베이스 상태로 변환함</li> <li>시스템이 가지고 있는 고정 요소는 트랜잭션 수행 전과 트랜잭션 수행 완료 후의 상태가 같아야 함</li> </ul>
Isolation (독립성, 격리성)	<ul style="list-style-type: none"> <li>둘 이상의 트랜잭션이 동시에 병행 실행되는 경우 어느 하나의 트랜잭션 실행 중에 다른 트랜잭션의 연산이 끼어들 수 없음</li> <li>수행 중인 트랜잭션은 완전히 완료될 때까지 다른 트랜잭션에서 수행 결과를 참조할 수 없음</li> </ul>

Durability (영속성, 지속성)	성공적으로 완료된 트랜잭션의 결과는 영구적으로 반영되어야 함
--------------------------	-----------------------------------

- 20 이 문제는 문제의 자료가 각 보기의 순서대로 출력되는지 스택을 이용해 직접 입출력을 수행해 보면 됩니다. PUSH는 스택에 자료를 입력하는 명령이고, POP는 스택에서 자료를 출력하는 명령입니다. 먼저 ①번을 먼저 살펴볼게요.



D DB DBC DBCA  
②번은 D 출력 후에 A를 출력해야 하는데, B와 C를 출력하지 않고는 A를 출력할 수 없으므로 불가능합니다.



#### [전문가의 조언]

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

#### 21 [전문가의 조언]

주요 레지스터의 기능을 묻는 문제가 자주 출제됩니다. 무슨 레지스터를 말하는지 알아낼 수 있도록 각 레지스터들의 개별적인 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 중심임
- 상태 레지스터(Status Register), PSWR(Program Status Word Register), 플래그 레지스터 : 시스템 내부의 순간순간의 상태가 기록된 정보를 PSW라고 함. 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
- 시프트 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

22 [전문가의 조언]

문제와 보기가 동일하게 출제되었던 문제입니다. 문제와 답만 기억하고 넘어가세요.

- 23 ISZ는 주소의 변경이나 프로그램 루프의 실행 횟수를 계산하는데 유용한 명령으로 지정된 주소에 저장된 워드의 내용을 1 증가시킨 다음 그 결과가 0이면 현재 명령을 skip하고, 0이 아니면 그대로 현재 명령을 실행합니다.

[전문가의 조언]

동일하거나 비슷한 유형의 문제가 가끔 출제되는데 코드의 의미를 이해할 수 있으면 어렵지 않게 맞힐 수 있습니다.

MAR ← MBR(AD)	MBR에 있는 명령어의 번지 부분을 MAR에 전송함
MBR ← M	메모리에서 MAR이 지정하는 위치의 값을 MBR에 전송함
MBR ← MBR + 1	MBR의 값을 1 증가시킴
M ← MBR, if(MBR=0) then (PC ← PC+1)	<ul style="list-style-type: none"> <li>• MBR의 값을 메모리의 MAR이 지정하는 위치에 전송함</li> <li>• MBR의 값이 0이면 다음 명령을 수행함</li> </ul>

- 24 채널은 CPU로부터 입·출력 명령을 받으면 주기억장치에서 채널 프로그램을 읽어와 명령을 해독하고 코드를 변환하여 입·출력을 직접 수행하는 장치로서 신호를 변·복조하는 MODEM의 기능은 없습니다.

[전문가의 조언]

채널의 의미, 특징, 종류 모두 중요합니다. 확실히 정리하세요.

채널(Channel)

- Channel은 CPU를 대신하여 주기억장치와 입·출력장치 사이에서 입·출력을 제어하는 입·출력 전용 프로세서(IOP)이다.
- 채널 제어기는 채널 명령어로 작성된 채널 프로그램을 해독하고 실행하여 입·출력 동작을 처리한다.
- 채널은 CPU로부터 입·출력 전송을 위한 명령어를 받으면 CPU와는 독립적으로 동작하여 입·출력을 완료한다.
- 채널은 주기억장치에 기억되어 있는 채널 프로그램의 수행과 자료의 전송을 위하여 주기억장치에 직접 접근한다.
- 채널은 CPU와 인터럽트로 통신한다.
- 채널의 종류
  - Selector Channel(선택 채널) : 고속 입·출력장치(자기 디스크, 자기 테이프, 자기 드럼)와 입·출력하기 위해 사용하며, 특정한 한 개의 장치를 독점하여 입·출력함
  - Multiplexer Channel(다중 채널) : 저속 입·출력장치(카드리더, 프린터)를 제어하는 채널로, 동시에 여러 개의 입·출력장치를 제어함
  - Block Multiplexer Channel : 고속 입·출력장치를 제어하는 채널로, 동시에 여러 개의 입·출력장치를 제어함

- 25 3×8 Decoder는 8개의 AND 게이트가 사용됩니다.

[전문가의 조언]

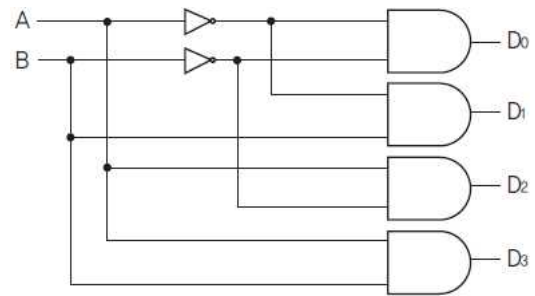
종종 출제되는 내용입니다. 디코더(Decoder)의 그림을 제시하고 무슨 회로나고 묻는 문제가 출제되기도 합니다. 디코더의 특징과 그림을 같이 알아두세요.

디코더(Decoder)

- n Bit의 Code화된 정보를 그 Code의 각 Bit 조합에 따라 2<sup>n</sup>개의 출력으로 번역하는 회로이다.
- 명령어의 명령부나 번지를 해독할 때 사용하며, 주로 AND

게이트로 구성된다.

• 회로



26 [전문가의 조언]

인터럽트 동작 원리를 알면 맞힐 수 있는 문제가 자주 출제됩니다. 무조건 암기하지 말고 현재 작업을 중지한 후 인터럽트를 처리하고 돌아와서 다시 원래의 작업을 수행하는 인터럽트의 개념을 생각하며 이해해 보세요.

인터럽트 동작 원리

- ① 인터럽트 요청 신호 발생

- ② 프로그램 실행을 중단 : 현재 실행중이던 명령어는 끝까지 실행

- ③ 현재의 프로그램 상태를 보존 : 프로그램 상태는 다음에 실행할 명령의 번지를 말하는 것으로서 PC(프로그램 카운터)가 가지고 있음. PC의 값을 메모리의 0번지 또는 스택에 보관함

- ④ 인터럽트 처리 루틴을 실행 : 인터럽트 처리 루틴을 실행하여 인터럽트를 요청한 장치를 식별함

- ⑤ 인터럽트 서비스(취급) 루틴을 실행 : 실질적인 인터럽트를 처리함

- ⑥ 상태 복구 : 인터럽트 요청 신호가 발생했을 때 스택에 보관한 PC의 값을 다시 PC에 저장함

- ⑦ 중단된 프로그램 실행 재개 : PC의 값을 이용하여 인터럽트 발생 이전에 수행중이던 프로그램을 계속 실행함

- 27 Op-Code는 현재 수행중인 명령어를 저장하는 명령 레지스터(Instruction Register)로 이동합니다.

[전문가의 조언]

동일한 문제가 종종 출제되었습니다. Op-Code는 현재 수행 중인 명령어를 저장하는 명령 레지스터(Instruction Register)로 이동한다는 것만 기억하고 넘어가세요.

- 28 먼저 10진수 6을 3초과(Excess-3) 코드로 변환한 후 3초과 코드를 그레이(Gray) 코드로 변환하면 됩니다.

3초과 코드로 변환하기

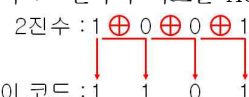
10진수 6에 3을 더한 후 2진수로 변환합니다.

$$6 + 3 \rightarrow (9)_{10} \rightarrow (1001)_2$$

그레이 코드로 변환하기

- 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.



[전문가의 조언]

그레이 코드(Gray Code)와 3초과 코드(Excess-3 Code)는 시험에 자주 출제됩니다. 두 코드의 특징과 변환하는 방법을 숙지하세요.

3초과 코드(Excess-3 Code)

- BCD+3, 즉 BCD 코드에 3<sub>10</sub>(00112)을 더하여 만든 코드이기 때문에 모든 비트가 동시에 0이 되는 경우는 없다.

- 대표적인 자기 보수 코드이며, 비가중치 코드이다.

#### 그레이 코드(Gray Code)

- BCD 코드의 인접하는 비트를 XOR 연산하여 만든 코드이다.
- 이웃하는 코드가 한 비트만 다르기 때문에 코드 변환이 용이하다.
- 입·출력장치, D/A 변환기, 주변장치 등에서 숫자를 표현할 때 사용한다.
- 1Bit만 변화시켜 다음 수치로 증가시키기 때문에 하드웨어적인 오류가 적다.
- Gray Code를 2진수로 변환하는 방법
  - ① 2진수의 첫 번째 비트는 그레이 코드를 그대로 내려 쓴다.
  - ② 두 번째 2진수 비트부터는 왼쪽에 구해 놓은 2진수 비트와 변경할 해당 번째 그레이 비트를 XOR 연산하여 쓴다.

- 29 보조기억장치는 주기억장치보다 액세스 속도가 느립니다.

#### [전문가의 조언]

자주 출제되는 내용은 아니지만 컴퓨터 사용에 있어 상식적인 내용입니다. 보조기억장치의 일반적인 특징을 정리하고 넘어가세요.

#### 보조기억장치

- 보조기억장치는 주기억장치의 단점을 보완하기 위한 기억장치이다.
- 보조기억장치는 주기억장치에 비해 속도는 느리지만 전원이 차단되어도 내용이 그대로 유지되고, 저장 용량이 크다는 장점이 있다.
- 중앙처리장치와 직접 자료 교환이 불가능하다.
- 접근 시간(Access Time)이 오래 걸린다.
- 일반적으로 주기억장치에 데이터를 저장할 때는 DMA 방식을 사용한다.
- CPU가 직접 접근할 수 없다.
- 가격이 비싸고 저장 용량이 적다.
- 전원 공급이 중단되면 기억된 내용이 모두 지워지므로 작업한 문서를 지속적으로 보관할 수 없다.

- 30 라인 프린터(Line Printer)는 출력장치입니다.

#### [전문가의 조언]

가끔 출제되는 문제입니다. 기본적인 내용이지만 혹시 모르는 게 있는지 확인하고 넘어가세요.

#### 입·출력장치의 종류

입력장치	키보드, 마우스, 스캐너, OMR, OCR, MICR, BCR, 마이크로 필름 입력장치(CIM), 라이트 펜, 터치스크린, 디지털타이저 등
출력장치	모니터, 프린터, 플로터, 마이크로 필름 출력장치(COM) 등
보조기억장치 (입·출력 겸용 장치)	자기 디스크, 자기 테이프, 자기 드럼, 하드디스크, 등

- 31 [전문가의 조언]

JK 플립플롭 하면 무(상태 변화 없음), 공(항상 0), 일(항상 1), 보(보수)가 먼저 떠오르도록 특징을 확실히 정리하세요.

#### JK 플립플롭

- RS FF에서  $S = R = 1$ 일 때 동작되지 않는 결점을 보완한 플립플롭이다.
- RS FF의 입력선 S와 R을 JK FF의 입력선 J와 K로 사용한다.

- 다른 모든 플립플롭의 기능을 대응할 수 있으므로 응용 범위가 넓고 집적 회로화 되어 가장 널리 사용된다.
- 특성표

J	K	$Q_{(t+1)}$	상태
0	0	$Q_{(t)}$	상태 변화 없음(무)
0	1	0	Reset(공)
1	0	1	Set(일)
1	1	$\overline{Q_{(t)}}$	반전(보)

- 32 명령 사이클에서 인터럽트를 처리한 후에는 항상 명령어를 읽어오는 Fetch 상태로 전이합니다.

#### [전문가의 조언]

메이저 스테이트 중에서 인터럽트 스테이트는 자주 출제되는 내용입니다. 인터럽트 스테이트의 의미를 숙지하고 동작 순서는 순서대로 나열할 수 있을 정도로만 이해해 두세요.

#### 인터럽트(Interrupt) 단계

- 인터럽트 발생 시 복귀주소(PC)를 저장시키고, 제어순서를 인터럽트 처리 프로그램의 첫 번째 명령으로 옮기는 단계이다.
- 인터럽트 단계를 마친 후에는 항상 Fetch 단계로 변천한다.
- 다음은 Interrupt Cycle의 동작 순서이다.

Micro Operation	의미
$MBR[AD] \leftarrow PC,$ $PC \leftarrow 0$	<ul style="list-style-type: none"> <li>• PC가 가지고 있는, 다음에 실행할 명령의 주소를 MBR의 주소 부분으로 전송함</li> <li>• 복귀주소를 저장할 0번지를 PC에 전송함</li> </ul>
$MAR \leftarrow PC,$ $PC \leftarrow PC + 1$	<ul style="list-style-type: none"> <li>• PC가 가지고 있는, 값 0번지를 MAR에 전송함</li> <li>• 인터럽트 처리 루틴으로 이동할 수 있는 인터럽트 벡터의 위치를 지정하기 위해 PC의 값을 1 증가시켜 1로 세트시킴</li> </ul>
$M[MAR] \leftarrow MBR,$ $IEN \leftarrow 0$	<ul style="list-style-type: none"> <li>• MBR이 가지고 있는, 다음에 실행할 명령의 주소를 메모리의 MAR이 가리키는 위치(0 번지)에 저장함</li> <li>• 인터럽트 단계가 끝날 때까지 다른 인터럽트가 발생하지 않게 IEN에 0을 전송함</li> </ul>
$F \leftarrow 0, R \leftarrow 0$	F에 0, R에 0을 전송하여 Fetch 단계로 변천함

- 33 [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

- 34 [전문가의 조언]

중요한 용어입니다. 꼭 기억하세요.

**Bandwidth(대역폭, 전송률)**



- 대역폭은 메모리로부터 또는 메모리까지 1초 동안 전송되는 최대한의 정보량으로, 기억장치의 자료 처리 속도를 나타내는 단위이다.
- 대역폭은 하드웨어의 특성상 주기억장치가 제공할 수 있는 정보 전달능력의 한계를 의미한다.
- 메모리 워드의 길이가 작을수록 대역폭이 좋다.
- 전송 단위 : Baud(보) = bps(1초당 전송 가능한 비트 수)

35 CISC형 프로세서는 명령어의 길이가 일정하지 않습니다. 명령어 길이가 일정한 것은 RISC형 프로세서입니다.

**[전문가의 조언]**

간혹 출제되는 내용입니다. 문제에 제시된 내용정도만 알아두고 넘어가세요.

36 전가산기는 자리올림(C)과 이진수 두 비트(A, B), 즉 이진수 세 비트를 더해 합과 캐리(자리올림)을 구하는 논리회로입니다. 다음은 전가산기의 진리표입니다.

A	B	C	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

진리표를 참조하지 않더라도 A, B, C 에 입력되는 값을 그대로 더해서 합과 캐리를 구해도 됩니다. A=1, B=0, C=1 이므로 결과는 다음과 같습니다.

$$\begin{array}{r} 1 \\ + 0 \\ \hline 1 \\ 1 \ 0 \end{array}$$

합은 0이 되고 자리올림은 1이 됩니다.

**[전문가의 조언]**

자주 출제되는 문제는 아니지만 또 출제될 수 있는 문제입니다. 계산 방법이 어렵지 않으니 꼭 이해하고 넘어가세요.

37 레지스터에서 기억장치로 정보를 옮기는 것을 스토어(store)라고 합니다. 로드(load)는 기억장치에서 레지스터로 정보를 옮기는 것을 말합니다.

**[전문가의 조언]**

자료 전달 기능에 사용되는 명령어들의 기능을 정리하고 넘어가세요.

**자료 전달 기능**

자료 전달 기능은 CPU와 기억장치 사이에서 정보를 교환하는 기능이다.

Load	기억장치에 기억되어 있는 정보를 CPU의 레지스터로 꺼내오는 명령
Store	CPU에 있는 정보를 기억장치에 기억시키는 명령
Move	레지스터 간에 자료를 전달하는 명령
Push	스택에 자료를 저장하는 명령
Pop	스택에서 자료를 꺼내오는 명령

38 입·출력 프로그램은 입·출력, 즉 주변 장치와 주기억장치간 데이터 입·출력을 처리하기 위한 목적으로 존재합니다. CPU의 로딩과는 무관합니다.

**[전문가의 조언]**

자주 출제되는 문제는 아닙니다. 보기에 제시된 입·출력 프로그램의 목적 정도만 기억해두고 넘어가세요.

39 사칙 연산은 수치적인 연산입니다.

**[전문가의 조언]**

연산자의 4가지 기능인 함수 연산, 자료 전달, 제어, 입·출력을 꼭 외우고, 각각의 기능에는 어떤 연산자들이 있는지 구분할 수 있을 정도로 알아두세요.

**연산자의 기능**

함수 연산 기능	<ul style="list-style-type: none"> <li>• 수치적인 산술 연산과 비수치적인 논리 연산이 있음</li> <li>• 산술 연산 : ADD, SUB, MUL, DIV, 산술 Shift 등</li> <li>• 논리 연산 : NOT, AND, OR, XOR, 논리적 Shift, Rotate, Complement, Clear 등</li> </ul>
자료 전달 기능	<ul style="list-style-type: none"> <li>• CPU와 기억장치 사이에서 정보를 교환하는 기능</li> <li>• Load : 기억장치에 기억되어 있는 정보를 CPU로 꺼내오는 명령</li> <li>• Store : CPU에 있는 정보를 기억장치에 기억시키는 명령</li> <li>• Move : 레지스터 간에 자료를 전달하는 명령</li> <li>• Push : 스택에 자료를 저장하는 명령</li> <li>• Pop : 스택에서 자료를 꺼내오는 명령</li> </ul>
제어 기능	<ul style="list-style-type: none"> <li>• 프로그래머가 명령의 실행 순서를 변경시키는 기능</li> <li>• 무조건 분기 명령 : GOTO, Jump(JMP) 등</li> <li>• 조건 분기 명령 : IF 조건, SPA, SNA, SZA 등</li> <li>• Call : 부 프로그램 호출</li> <li>• Return : 부 프로그램에서 메인 프로그램으로 복귀</li> </ul>
입·출력 기능	<ul style="list-style-type: none"> <li>• CPU와 I/O 장치 또는 메모리와 I/O 장치 사이에서 자료를 전달하는 기능</li> <li>• INPUT : 입·출력장치의 자료를 주기억장치로 입력하는 명령</li> <li>• OUTPUT : 주기억장치의 자료를 입·출력장치로 출력하는 명령</li> </ul>

40 [전문가의 조언]

운영체제의 운용 기법을 구별해내는 문제가 종종 출제됩니다. 각 기법들의 개별적인 특징을 정리하고 넘어가세요.

**운영체제의 운용 기법**

- **일괄 처리 시스템(Batch Processing System)** : 초기의 컴퓨터 시스템에서 사용된 형태로, 일정량 또는 일정 기간 동안 데이터를 모아서 한꺼번에 처리하는 방식
- **다중 프로그래밍 시스템(Multi-Programming System)** : 하나의 CPU와 주기억장치를 이용하여 여러 개의 프로그램을 동시에 처리하는 방식
- **시분할 시스템(Time Sharing System)** : 여러 명의 사용자가 사용하는 시스템에서 컴퓨터가 사용자들의 프로그램을 번갈아 가며 처리해 줌으로써 각 사용자에게 독립된 컴퓨터를 사용하는 느낌을 주는 것이며 라운드 로빈(Round Robin) 방식이라고도 함
- **다중 처리 시스템(Multi-Processing System)** : 여러 개의 CPU와 하나의 주기억장치를 이용하여 여러 개의 프로그램

램을 동시에 처리하는 방식

- **실시간 처리 시스템(Real Time Processing System)** : 데이터 발생 즉시, 또는 데이터 처리 요구가 있는 즉시 처리하여 결과를 산출하는 방식
- **다중 모드(Multi-Mode Processing)** : 일괄 처리 시스템, 시분할 시스템, 다중 처리 시스템, 실시간 처리 시스템을 한 시스템에서 모두 제공하는 방식
- **분산 처리 시스템(Distributed Processing System)** : 여러 개의 컴퓨터(프로세서)를 통신 회선으로 연결하여 하나의 작업을 처리하는 방식

- 41 문서화는 시스템의 개발 요령과 순서 등 시스템 개발에 관련된 모든 행위를 문서로 만들어 두는 것으로, 업무에 실질적인 도움을 주고자 하는 것이지 요식적 절차를 부각시키기 위해서 하는 것은 아닙니다.

#### [전문가의 조언]

시스템 문서화의 효과(목적)를 묻는 문제는 자주 출제되고 있습니다. 어려운 내용이 아니니 읽어보면서 문서화에 대한 효과(목적)를 정리하세요.

#### 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 재산화할 수 있다.
- 타업무 개발에 참고할 수 있다.

- 42 [전문가의 조언]

코드 설계와 관련해서는 특정 단계에서 수행하는 작업에 대한 구체적인 문제보다는 코드 설계 순서를 나열하는 문제가 자주 출제됩니다. 우선 사용 범위 결정 과정에서 수행하는 작업이 무엇인지 알아두고, 다음의 코드 설계 순서를 정확히 기억해 두세요.

#### 코드 설계 순서

코드화 대상 선정 → 코드화 목적의 명확화 → 코드 부여 대상 수 확인 → 사용 범위 결정 → 사용 기간 결정 → 코드화 대상의 특성 분석 → 코드 부여 방식의 결정 → 코드의 문서화

- 43 [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

변환 (Conversion)	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
정렬 (Sort, 분류)	레코드를 처리할 순서에 맞게 오름차순 또는 내림차순으로 재배치하는 기능
병합(Merge)	동일한 파일 형식을 갖는 2개 이상의 파일을 일정한 규칙에 따라 하나의 파일로 통합 처리하는 기능
대조 (Matching)	2개의 파일을 대조시켜 그 기록 순서나 기록 내용을 검사하는 기능
보고서 (Reporting)	처리 결과를 출력하는 기능
추출(Extract)	파일 안에서 특정 조건에 만족하는 데이터만을 골라내는 기능으로, 정보 검색을 위한 필수 기능
조합(Collate)	2개 이상의 파일에서 조건에 맞는 것을 골라 새로운 레코드로 파일을 만드는 기능

- 44 [전문가의 조언]

특정 검사 방법을 묻는 문제보다는 계산 처리 단계에서의 검사와 입력 단계에서의 검사 방법을 구분하는 문제가 더 자주 출제됩니다. 외울 종류가 적은 계산 처리 단계에서의 검사 방법을 확실히 숙지하고 넘어가세요.

#### 계산 처리 단계에서의 검사 방법

- **부호 검사(Sign Check = Plus-Minus Check)** : 계산 결과가 양수 또는 음수인지를 검사하는 방법
- **중복 레코드 검사(Double Record Check)** : 계산 처리하는 과정에서 동일한 레코드가 있는지를 검사하는 방법
- **불일치 레코드 검사(Unmatch Record Check)** : 마스터 파일과 트랜잭션 파일을 조합할 때 키 항목이 일치하는지의 여부를 검사하는 방법
- **오버플로 검사(Overflow Check)** : 계산된 결과가 규정된 자릿수 또는 한계를 초과하는지를 검사하는 방법
- **제로 균형 검사(Zero Balance Check)** : 계산 결과가 0이 되는지를 검사하는 방법
- **불능 검사(Impossible Check)** : 0으로 나누는 경우가 있는지를 검사하는 방법

- 45 [전문가의 조언]

자주 출제되는 내용입니다. 자료 사전에서 사용되는 표기 기호와 각각의 기능을 연결할 수 있어야 합니다.

#### 자료 사건의 기호

기호	의미
=	자료의 정의 : ~로 구성되어 있다(is composed of).
+	자료의 연결 : 그리고(and)
( )	자료의 생략 : 생략 가능한 자료(Optional)
[ ]	자료의 선택 : 다중 택일(Selection)
{ }	자료의 반복

- 46 코드는 복잡하지 않게 짧고 단순해야 합니다.

#### [전문가의 조언]

자주 출제되는 내용입니다. 코드 설계 시 유의사항이 아닌

것을 찾아낼 수 있도록 숙지해 두세요.

#### 코드 설계 시 유의사항

- 컴퓨터 처리에 적합해야 한다(기계 처리의 용이성).
- 사용자가 취급하기 쉬어야 한다(취급의 용이성).
- 공통성과 체계성이 있어야 한다(분류의 편리성).
- 확장하기 쉬워야 한다(확장성).
- 단순하고 짧게 한다(단순성).
- 대상 자료와 일 대 일로 대응되도록 설계한다(고유성).
- 연상하기 쉬워야 한다(표의성).

#### 47 [전문가의 조언]

클래스뿐만 아니라 나머지 용어들도 중요합니다. 각각의 개념을 정확하게 알아두고 넘어가세요.

- **인스턴스(Instance)** : 클래스에 속한 각각의 객체를 의미하며, 클래스로부터 새로운 객체를 생성하는 것을 인스턴스화(Instantiation)라고 함
- **메소드(Method)** : 객체에 정의된 연산을 의미하며, 객체의 상태를 참조하거나 변경하는 수단이 됨
- **메시지(Message)** : 외부로부터 하나의 객체에 전달되는 메소드(연산)의 요구를 의미함

#### 48 [전문가의 조언]

럼바우 객체 분석 기법의 3가지 모델링의 종류와 제시된 내용이 무슨 모델링인지를 묻는 문제가 출제됩니다. 확실히 파악하고 넘어가세요.

#### 럼바우(Rumbaugh)의 분석 기법

<b>객체 모델링 (Object Modeling)</b>	정보 모델링이라고도 하며, 시스템에서 요구되는 객체를 찾아내어 속성과 연산 식별 및 객체들 간의 관계를 규정하여 객체 다이어그램으로 표시하는 것
<b>동적 모델링 (Dynamic Modeling)</b>	상태 다이어그램(상태도)을 이용하여 시간의 흐름에 따른 객체들 사이의 제어 흐름, 상호 작용, 동작 순서 등의 동적인 행위를 표현하는 모델링
<b>기능 모델링 (Functional Modeling)</b>	자료 흐름도(DFD)를 이용하여 다수의 프로세스들 간의 자료 흐름을 중심으로 처리 과정을 표현한 모델링

#### 49 [전문가의 조언]

종종 출제되는 문제입니다. COM 시스템의 개념을 꼭 기억하세요. 그리고 나머지 용어의 개념은 가볍게 한번 읽어 보고 넘어가세요.

- **CRT 출력 시스템** : 정보처리 결과 또는 출력 정보를 CRT 모니터를 통해 화면에 출력하는 방식
- **X-Y 플로터** : 초기에 사용하던 펜 플로터로, 펜(Pen)이나 종이를 X축과 Y축으로 움직이면서 인쇄하는 방식, 주로 지도, 통계 도표, 설계 도면 등을 출력하는 데 사용됨
- **음성 출력 시스템** : 정보처리 결과를 소리 형태로 출력하는 방식

#### 50 [전문가의 조언]

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

#### 코드의 종류

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

#### 51 [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

52 IPT 기법은 프로그래머의 충원이 아니라 교대 근무를 용이하게 합니다.

#### [전문가의 조언]

거의 출제되지 않는 내용입니다. IPT 기법의 적용 목적과 함께 IPT 기법의 개념 정도만 알아두세요.

#### **IPT(Improved Programming Technique) 기법**

- 소프트웨어의 품질 개선과 생산성 향상을 위해 사용되는 프로그램 개발 기법을 의미한다.
- 보다 효율적이고 신뢰성 높은 프로그램을 개발하기 위한 각종 기법의 총칭이다.
- 프로그램을 개발할 때 기술적, 관리적 측면에서 모두 우수한 개발 작업이 되도록 한다.

#### **53 [전문가의 조언]**

종종 출제되는 내용입니다. 프로그램 흐름도와 함께 나머지 용어의 의미도 정리하고 넘어가세요.

- **블록 차트(Block Chart)** : 시스템의 목적을 달성하는 데 필요한 모든 기능 및 각 부서를 블록으로 표시하는 차트로, 블록 다이어그램이라고도 함
- **시스템 흐름도(System Flowchart)** : 자료 발생부터 결과를 얻기까지 시스템의 전 과정을 나타내는 흐름도
- **프로세스 흐름도(Process Flowchart)** : 컴퓨터의 입력, 처리, 출력 과정을 나타내는 흐름도로, 오퍼레이터에게 처리 공정을 알려주고 컴퓨터의 전체적인 논리 구조의 파악, 컴퓨터의 사용 시간 계산 등에 사용됨

#### **54 시스템의 기본 요소에는 입력(Input), 처리(Process), 출력(Output), 제어(Control), 피드백(FeedBack)이 있습니다.**

##### **[전문가의 조언]**

시스템의 기본 요소 5가지와 각 요소의 작업 내역을 묻는 문제가 자주 출제됩니다. 꼭 숙지해 두세요.

##### **시스템의 기본 요소**

<b>입력(Input)</b>	처리할 데이터, 처리 방법, 처리 조건을 시스템에 투입하는 것
<b>처리(Process)</b>	입력된 데이터를 처리 방법과 조건에 따라 처리하는 것
<b>출력(Output)</b>	처리된 결과를 시스템에서 산출하는 것
<b>제어(Control)</b>	자료가 입력되어 출력될 때까지의 처리 과정이 올바르게 행해지는지 감독하는 것
<b>피드백(FeedBack)</b>	출력된 결과가 예정된 목적을 만족시키지 못한 경우 목적 달성을 위해 반복 처리하는 것

#### **55 [전문가의 조언]**

표준 처리 패턴에 대한 내용은 자주 출제된다고 했죠? 43번 문제의 [전문가의 조언]을 확실히 숙지하세요.

#### **56 [전문가의 조언]**

자주 출제되는 내용입니다. 시스템의 특성 4가지와 각각의 의미를 꼭 기억해 두세요.

##### **시스템의 특성**

<b>자동성</b>	어떤 조건이나 상황의 변화에 대응하여 자동으로 적절한 처리를 수행함
<b>목적성</b>	서로 다른 기능을 가지고 있는 시스템의 각 구성 요소들은 어떤 하나의 공통된 목적을 위해 존재함
<b>제어성</b>	시스템이 오류 없이 그 기능을 발휘하기 위하여 정해진 규정이나 한계, 또는 궤도로부터 이탈되는 사태나 현상의 발생을 사전에 감지하여 그것을 바르게 수정해 가는 것
<b>종합성</b>	항상 다른 관련 시스템과 상호 의존 관계를 유지함

#### **57 [전문가의 조언]**

코드에 관해서는 코드의 기능이나 각각의 기능이 의미하는 것이 무엇인지를 묻는 문제가 출제됩니다. 코드 기능들은 각각의 명칭을 통해 쉽게 추측할 수 있으니 명칭을 통해 기능을 파악해 두세요.

##### **코드의 기능**

<b>3대 기능</b>	<b>그 밖의 기능</b>
<ul style="list-style-type: none"> <li>• 분류 기능</li> <li>• 식별 기능</li> <li>• 배열 기능</li> </ul>	<ul style="list-style-type: none"> <li>• 간소화 기능</li> <li>• 암호화 기능</li> <li>• 연상 기능(표의성)</li> <li>• 추출 기능</li> <li>• 표준화 기능</li> <li>• 단순화 기능</li> <li>• 오류 검출 기능</li> </ul>

#### **58 시스템 평가는 구체화된 새로운 시스템이 본래의 목적에 만족하는가를 평가하는 것으로 시스템 운영 요원의 재훈련과는 관련이 없습니다.**

##### **[전문가의 조언]**

동일한 문제가 출제된 적이 있습니다. 시스템 평가의 목적을 읽으면서 정리하세요.

##### **시스템 평가의 목적**

- 시스템의 성능과 유용도를 판단할 수 있다.
- 처리 비용과 처리 효율 면에서 개선점을 파악할 수 있다.
- 시스템 운용 관리의 타당성을 파악할 수 있다.
- 다른 시스템을 개발할 때 원활한 진행을 위한 참고 자료가 될 수 있으며, 동일한 실수를 하지 않게 된다.

#### **59 [전문가의 조언]**

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

#### **60 프로그래머의 수가 4명이고, 1인당 평균 생산성은 월간 1000 라인이므로, 4명은 월간 4000(1000×4) 라인을 생산할 수 있습니다. 그러므로 이 4명이 100000 라인을 개발하려면 100000/ 4000 = 25 개월이 소요됩니다.**

##### **[전문가의 조언]**

공식을 몰라도 문제를 통해 답을 찾을 수 있는 문제입니다. 이런 문제는 틀리지 마세요.

#### **61 [전문가의 조언]**

프로세스의 여러 가지 정의에 대한 문제가 자주 출제되니 명확히 숙지해 두세요.

##### **프로세스의 정의**

- 실행중인 프로그램
- PCB를 가진 프로그램
- 실기억장치에 저장된 프로그램
- 프로세서가 할당되는 실제
- 프로시저가 활동중인 것



- 비동기적 행위를 일으키는 주체
- 지정된 결과를 얻기 위한 일련의 계통적 동작
- 목적 또는 결과에 따라 발생하는 사건들의 과정

62 UNIX 시스템은 리스트 구조가 아니라 트리 구조로 이루어져 있습니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. UNIX 시스템의 특징은 세부적인 내용까지 알고 있어야 합니다.

**UNIX의 특징**

- 시분할 시스템(Time Sharing System)을 위해 설계된 대화식 운영체제로, 소스가 공개된 개방형 시스템(Open System)이다.
- 대부분 C 언어로 작성되어 있어 이식성이 높으며 장치, 프로세스 간의 호환성이 높다.
- 크기가 작고 이해하기가 쉽다.
- 다중 사용자(Multi-User), 다중 작업(Multi-Tasking)을 지원한다.
- 많은 네트워킹 기능을 제공하므로 통신망(Network) 관리용 운영체제로 적합하다.
- 트리 구조의 파일 시스템을 갖는다.
- 전문적인 프로그램 개발에 용이하다.
- 다양한 유틸리티 프로그램들이 존재한다.

63 교착상태(DeadLock) 발생의 필요조건에는 상호 배제(Mutual Exclusion), 점유와 대기(Hold and Wait), 비선점(Non-Preemption), 환형 대기(Circular Wait)가 있습니다.

**[전문가의 조언]**

교착상태 발생의 4가지 필요조건과 각각의 의미를 묻는 문제가 자주 출제되니 꼭 숙지해 두세요.

**교착상태 발생의 필요충분조건**

- **상호 배제(Mutual Exclusion)** : 한 번에 한 개의 프로세스만이 공유 자원을 사용할 수 있어야 함
- **점유와 대기(Hold and Wait)** : 최소한 하나의 자원을 점유하고 있으면서 다른 프로세스에 할당되어 사용되고 있는 자원을 추가로 점유하기 위해 대기하는 프로세스가 있어야 함
- **비선점(Non-preemption)** : 다른 프로세스에 할당된 자원은 사용이 끝날 때까지 강제로 빼앗을 수 없어야 함
- **환형 대기(Circular Wait)** : 공유 자원과 공유 자원을 사용하기 위해 대기하는 프로세스들이 원형으로 구성되어 있어 자신에게 할당된 자원을 점유하면서 앞이나 뒤에 있는 프로세스의 자원을 요구해야 함

64 [전문가의 조언]

스래싱은 이 문제와 같이 개념을 묻는 문제가 출제되므로 개념만 명확히 알아두면 됩니다. 나머지 보기로 제시된 용어의 개념도 알아두세요.

- **오버레이(Overlay) 기법** : 오버레이 기법은 주기억장치보다 큰 사용자 프로그램을 실행하기 위한 기법
- **세마포어(Semaphore)** : ‘신호기’, ‘깃발’을 뜻하며, 각 프로세스에 제어 신호를 전달하여 순서대로 작업을 수행하도록 하는 기법
- **페이징(Paging) 기법** : 가상 기억장치에 보관되어 있는 프로그램과 주기억장치의 영역을 동일한 크기로 나눈 후 나뉜 프로그램(페이지)을 동일하게 나뉜 주기억장치의 영역(페이지 프레임)에 적재시켜 실행하는 기법

65 성(Star)형은 모든 노드가 하나의 중앙 노드에 직접 연결되어 있고, 그 외 다른 노드와는 연결되어 있지 않은 구조입니다.

각 노드들이 Point-to-Point 형태로 모든 노드들과 직접 연결되는 것은 완전 연결(Fully Connection)형 구조입니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. 특징에 따른 분산 운영체제의 위상을 구분할 수 있도록 정리해 두세요.

**위상에 따른 분산 운영체제의 분류**

- **망형 - 완전 연결(Fully Connection)형** : 각 사이트들이 시스템 내의 다른 모든 사이트들과 직접 연결된 구조
- **망형 - 부분 연결(Partially Connection)형** : 시스템 내의 일부 사이트들 간에만 직접 연결하는 것으로, 직접 연결되지 않은 사이트는 연결된 다른 사이트를 통해 통신하는 구조
- **트리(Tree) 또는 계층(Hierarchy)형** : 분산 처리 시스템의 가장 대표적인 형태로, 각 사이트들이 트리 형태로 연결된 구조
- **스타(Star)형 = 성형** : 모든 사이트가 하나의 중앙 사이트에 직접 연결되어 있고, 그 외 다른 사이트와는 연결되어 있지 않은 구조
- **링형(Ring) = 환형** : 시스템 내의 각 사이트가 인접하는 다른 두 사이트와만 직접 연결된 구조
- **다중 접근 버스 연결(Multi Access Bus Connection)형** : 시스템 내의 모든 사이트들이 공유 버스에 연결된 구조

66 [전문가의 조언]

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

- **Chmod** : 파일의 보호 모드를 설정하여 파일의 사용 허가를 지정함
- **ps** : 현재 작업중인 프로세스의 상태 정보를 확인함
- **ls** : 현재 디렉터리 내의 파일 목록을 확인함

67 13K 작업을 최초 적합(First Fit)으로 할당할 경우 16K 공백에, 최적 적합(Best Fit)으로 할당할 경우 14K 공백에, 최악 적합(Worst Fit)으로 할당할 경우 30K 공백에 할당됩니다.

**[전문가의 조언]**

배치 전략의 종류와 각각의 배치원리에 대한 문제는 자주 출제됩니다. 확실히 알아두세요.

**배치 전략**

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

68 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 → 95 → 100’ 순이며, 가장 먼저 처리되는 트랙은 50입니다.

**[전문가의 조언]**

종종 출제되는 내용입니다. 문제를 통해 SCAN 스케줄링의 동작 원리를 숙지해 두세요.

69 모니터(Monitor)는 동기화를 구현하기 위한 특수 프로그램 기법으로, 모니터에는 한 순간에 하나의 프로세스만 진입하여 자원을 사용할 수 있습니다.

**[전문가의 조언]**

모니터의 의미와 특징을 묻는 문제가 종종 출제되니 꼭 정리해 두세요.

**모니터(Monitor)**

- 모니터는 동기화를 구현하기 위한 특수 프로그램 기법으로 특정 공유 자원을 프로세스에게 할당하는 데 필요한 데이터와 이 데이터를 처리하는 프로시저로 구성된다.
- 자료 추상화와 정보 은폐 개념을 기초로 하며 공유 자원을 할당하기 위한 병행성 구조로 이루어져 있다.
- 모니터 내의 공유 자원을 사용하려면 프로세스는 반드시 모니터의 진입부를 호출해야 한다.
- 외부의 프로시저는 직접 액세스할 수 없다.
- 모니터의 경계에서 상호 배제가 시행된다.
- 모니터에는 한순간에 하나의 프로세스만 진입하여 자원을 사용할 수 있다.
- 모니터에서는 Wait와 Signal 연산이 사용된다.

- 70 운영체제의 목적 중 하나는 응답 시간 및 반환 시간의 단축입니다.

**[전문가의 조언]**

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

**운영체제 목적**

운영체제의 목적에는 처리 능력 향상, 사용 가능성도 향상, 신뢰도 향상, 반환 시간 단축 등이 있다. 처리 능력, 반환 시간, 사용 가능성도, 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 가능성도 (Availability)	시스템을 사용할 필요가 있을 때 즉시 사용 가능한 정도
신뢰도(Reliability)	시스템이 주어진 문제를 정확하게 해결하는 정도

- 71 라운드 로빈은 시간 할당량이 작아질수록 문맥교환 과부하는 증가합니다.

**[전문가의 조언]**

자주 출제되는 내용입니다. 시간 할당량에 따른 특징을 중심으로 Round-Robin 스케줄링 기법을 정리해 두세요.

**라운드 로빈(RR: Round-Robin)**

- 시분할 시스템(Time Sharing System)을 위해 고안된 방식으로, FCFS 알고리즘을 선점 형태로 변형한 기법이다.
- FCFS 기법과 같이 준비상태 큐에 먼저 들어온 프로세스가 먼저 CPU를 할당받지만 각 프로세스는 시간 할당량(Time Slice, Quantum) 동안만 실행한 후 실행이 완료되지 않으면 다음 프로세스에게 CPU를 넘겨주고 준비상태 큐의 가장 뒤로 배치된다.
- 할당되는 시간이 클 경우 FCFS 기법과 같아지고, 할당되는 시간이 작을 경우 문맥교환 및 오버헤드가 자주 발생된다.
- 할당되는 시간의 크기가 작으면 작은 프로세스들에게 유리하다.

- 72 HRN 기법의 우선순위 계산식은 '(대기 시간 + 서비스 시간) / 서비스 시간'입니다.

- A 작업 :  $(30 + 10) / 10 = 4$

- B 작업 :  $(12 + 6) / 6 = 3$

- C 작업 :  $(12 + 12) / 12 = 2$

계산된 숫자가 클수록 우선순위가 높으므로 우선순위는 'A → B → C'입니다.

**[전문가의 조언]**

HRN 스케줄링의 우선 순위 계산식은 물론 직접 계산하는 문제도 자주 출제됩니다. 꼭 숙지하고 넘어가세요.

**73 [전문가의 조언]**

페이지 교체 알고리즘을 구분하는 문제가 자주 출제됩니다. 무슨 페이지 교체 알고리즘을 말하는지 구분할 수 있을 정도로 각각의 특징을 정리해 두세요.

**페이지 교체 알고리즘**

- OPT(OPTimal replacement, **최적 교체**) : 앞으로 가장 오랫동안 사용하지 않을 페이지를 교체하는 기법
- FIFO(First In First Out) : 각 페이지가 주기억장치에 적재될 때마다 그때의 시간을 기억시켜 가장 먼저 들어와서 가장 오래 있었던 페이지를 교체하는 기법
- LRU(Least Recently Used) : 최근에 가장 오랫동안 사용하지 않은 페이지를 교체하는 기법으로, 각 페이지마다 계수기(Counter)나 스택(Stack)을 두어 현 시점에서 가장 오랫동안 사용하지 않은, 즉 가장 오래 전에 사용된 페이지를 교체함
- LFU(Least Frequently Used) : 사용 빈도가 가장 적은 페이지를 교체하는 기법
- NUR(Not Used Recently) : 최근에 사용하지 않은 페이지를 교체하는 기법으로, 최근의 사용 여부를 확인하기 위해서 각 페이지마다 두 개의 비트, 즉 참조 비트(Reference Bit)와 변형 비트(Modified Bit, Dirty Bit)가 사용됨
- SCR(Second Chance Replacement) : 가장 오랫동안 주기억장치에 있던 페이지 중 자주 사용되는 페이지의 교체를 방지하기 위한 것으로, FIFO 기법의 단점을 보완하는 기법

- 74 강결합 시스템은 여러 개의 프로세서가 하나의 메모리를 공유하여 사용하는 시스템입니다. 각 프로세서마다 독립된 메모리를 갖는 것은 약결합 시스템입니다.

**[전문가의 조언]**

약결합 시스템과 강결합 시스템을 구분하는 문제가 종종 출제됩니다. 각각의 특징을 파악해 두세요.

**약결합 시스템(Loosely Coupled System)**

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

**강결합 시스템(Tightly Coupled System)**

- 동일 운영체제 하에서 여러 개의 프로세서가 하나의 메모리를 공유하여 사용하는 시스템이다.
- 하나의 운영체제가 모든 프로세서와 시스템 하드웨어를 제어한다.
- 프로세서 간의 통신은 공유 메모리를 통해서 이루어진다.
- 하나의 메모리를 사용하므로 프로세서 간의 결합력이 강하다.

- 공유 메모리를 차지하려는 프로세서 간의 경쟁을 최소화해야 한다.

75 주/종 처리기 시스템에서 주프로세서는 입·출력과 연산을 수행하고, 종프로세서는 연산만 수행합니다.

[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>

76 3개의 페이지 프레임에 갖는 주기억장치이므로 다음 그림과 같이 표현할 수 있습니다.

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

※ ● : 페이지 부재 발생

참조 페이지가 페이지 프레임에 없을 경우는 페이지 결함(부재)이 발생합니다. 초기에는 모든 페이지가 비어 있으므로 처음 1, 2, 3 페이지 적재 시 페이지 결함이 발생합니다. FIFO 기법은 각 페이지가 주기억장치에 적재될 때마다 그때의 시간을 기억시켜 가장 먼저 들어와서 가장 오래 있었던 페이지를 교체하는 기법이므로 참조 페이지 4를 참조할 때에는 가장 먼저 들어와 가장 오래 있었던 1을 제거한 후 4를 가져오게 됩니다. 이런 방법으로 요청된 페이지를 모두 처리하면 총 페이지 결함 발생 수는 5회이고, 최종적으로 남아 있는 페이지는 3, 1, 4입니다.

[전문가의 조언]

자주 출제되는 내용입니다. 문제를 통해 FIFO의 동작 원리와 페이지 부재 발생 횟수 계산 방법을 기억해 두세요.

77 스케줄링은 프로세스에게 우선 순위를 지정하여 우선 순위가 높은 프로세스를 먼저 실행합니다.

[전문가의 조언]

스케줄링에 관한 문제는 주로 스케줄링의 목적을 묻는 문제가 출제됩니다. 스케줄링의 목적을 정리하고 넘어가세요.

스케줄링의 목적

- **공정성** : 모든 프로세스에 공정하게 할당함
- **처리율 증가** : 단위 시간당 프로세스를 처리하는 비율을 증가시킴
- **CPU 이용률 증가** : 프로세스 실행 과정에서 주기억장치를 액세스한다든지, 입·출력 명령의 실행 등의 원인에 의해서 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
- **회전 시간(Turn Around Time) 최소화** : 프로세스를 제출

한 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

78 SJF 기법은 가장 짧은 작업을 먼저 수행하므로, 다음과 같은 순서로 수행됩니다.

진행 시간	0	10	20	30	40	50	60
1	0 도착 실행 시작	20 실행	20 완료	7 실행			
3		8 도착	12 대기	20 실행 시작	27 완료		
2		3 도착	24 대기	27 실행 시작	32 실행	59 완료	

평균 대기 시간은  $(0 + 12 + 24) / 3 = 12$  시간이 됩니다.

[전문가의 조언]

종종 출제되는 내용입니다. SJF 기법의 동작 원리를 알면 어렵지 않으니 동작 원리를 꼭 기억해 두세요.

79 [전문가의 조언]

은행원 알고리즘(Banker's Algorithm)이 회피 기법(Avoidance)이라는 것만 알아도 맞힐 수 있는 문제가 종종 출제됩니다. '은행원 알고리즘'은 교착 상태 해결 기법 중 회피 기법이라는 것을 꼭 기억하세요.

80 명령어 해독은 셸(Shell)의 기능입니다.

[전문가의 조언]

커널(Kernel)과 셸(Shell)의 기능을 구분하는 문제가 자주 출제됩니다. 자원 관리는 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 [전문가의 조언]

시분할 다중화기의 특징이나 동기식과 비동기식을 구분하는 문제가 자주 출제됩니다. 각각의 특징을 정확히 정리해 두세요



요.

### 시분할 다중화기

- 시분할 다중화기(TDM, Time Division Multiplexer)는 통신 회선의 대역폭을 일정한 시간폭(Time Slot)으로 나누어 여러 대의 단말장치가 동시에 사용할 수 있도록 한 것이다.
- 디지털 회선에서 주로 이용하며, 대부분의 데이터 통신에 사용된다.

### 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 비트가 더 필요하다.
- 통신 회선의 데이터 전송률이 전송 디지털 신호의 데이터 전송률을 능가할 때 사용한다.
- 전송할 데이터가 없는 경우에도 시간폭(Time Slot)이 제공되므로 효율성이 떨어진다.

### 비동기식 시분할 다중화기

- 비동기식 시분할 다중화기(ATDM, Asynchronous TDM)는 마이크로 프로세서를 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.
- 데이터 전송량이 많아질 경우 전송 지연이 생길 수 있다.
- 동기식 시분할 다중화기에 비해 접속하는 데 소요되는 시간이 길다.
- 주소 제어, 흐름 제어, 오류 제어 등의 기능을 하므로 복잡한 제어 회로와 임시 기억장치가 필요하며 가격이 비싸다.
- 지능 다중화기, 확률적 다중화기, 통계적 시분할 다중화기라고도 한다.

- 82 회선 교환 방식은 일단 접속이 되고 나면 전용 회선에 의한 통신처럼 데이터가 전달되는 고정 대역 전송이므로 메시지가 긴 경우에도 적합합니다.

### [전문가의 조언]

회선 교환 방식의 특징을 묻는 문제가 종종 출제되고 있습니다. ‘일단 접속이 이루어지면 접속을 해제할 때까지 전용선처럼 사용할 수 있다’는 것을 중심으로 특징을 정리하세요.

### 회선 교환 방식

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

유지된다.

- 전송된 데이터의 오류 제어나 흐름 제어는 사용자에게 의해서 수행된다.

### 83 [전문가의 조언]

LAN에서는 주로 LAN의 정의와 특징을 묻는 문제가 출제됩니다. 꼭 정리해 두세요.

### LAN

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), 망형으로 분류할 수 있다.

- 84 나이퀴스트(Nyquist) Sampling Theorem을 나이퀴스트 샤논의 이름과 연결지어 샤논의 표본화 이론이라고도 합니다.

### [전문가의 조언]

문제와 보기가 동일하게 출제되었던 문제입니다. 먼저 문제와 답을 기억해 두세요. 그리고 샤논의 표본화 이론이 무엇인지 가볍게 읽어보세요.

### 샤논(Nyquist Shanon)의 표본화 이론

어떤 신호  $f(t)$ 가 의미를 지니는 최고의 주파수보다 2배 이상의 속도로 균일한 시간 간격 동안 채집된다면, 이 채집된 데이터는 원래의 신호가 가진 모든 정보를 포함한다.

### 85 [전문가의 조언]

위성 통신의 다원 접속 방법과 관련해서는 CDMA의 의미를 묻는 문제가 가끔 출제되었었는데, 이번에는 다원 접속 방법의 종류를 묻는 문제가 출제되었네요. 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

### 86 [전문가의 조언]

PAD(Packet Assembler Disassembler)의 기능을 묻는 문제가 종종 출제되고 있습니다. 기억해 두세요.

### 87 [전문가의 조언]

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

물리 계층	• 전송에 필요한 두 장치 간의 실제 접속과 절단 등 기계적, 전기적, 기능적, 절차적 특성에 대한 규칙을 정의함
-------	---



	<ul style="list-style-type: none"> <li>물리적 전송 매체와 전송 신호 방식을 정의하며, RS-232C, X.21 등의 표준이 있음</li> </ul>
<b>데이터 링크 계층</b>	<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>HDL, LAPB, LLC, LAPD 등의 표준이 있음</li> </ul>
<b>네트워크 계층</b>	<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>
<b>전송(트랜스포트) 계층</b>	<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>
<b>세션 계층</b>	<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>
<b>표현 계층</b>	<ul style="list-style-type: none"> <li>응용 계층으로부터 받은 데이터를 세션 계층에 보내기 전에 통신에 적당한 형태로 변환하고, 세션 계층에서 받은 데이터는 응용 계층에 맞게 변환하는 기능을 함</li> <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, 가상 터미널(TELNET), 파일 전송(FTP) 등의 서비스를 제공함</li> </ul>

## 88 [전문가의 조언]

통신 방식과 관련해서는 종류와 원리는 물론 사용 예를 포함한 문제가 다양하게 출제되고 있으니 확실히 정리해 두세요.  
통신 방식

<b>단방향 통신</b>	한쪽 방향으로만 전송이 가능한 방식 (예) 라디오, TV
<b>반이중 통신</b>	양방향 전송이 가능하지만 동시에 양쪽 방향에서 전송할 수 없는 방식 (예) 무전기, 모뎀을 이용한 데이터 통신
<b>전이중 통신</b>	동시에 양방향 전송이 가능한 방식 (예) 전화, 전용선을 이용한 데이터 통신

## 89 FTP는 응용 계층 프로토콜입니다.

## [전문가의 조언]

OSI 7계층에서는 각 계층의 주요 기능을 파악하고 있어야 풀 수 있는 문제가 자주 출제된다고 했죠? 이 문제를 틀렸다면, 87번 문제의 [전문가의 조언]을 참조하여 계층별 주요 기능을 다시 한 번 암기하세요.

## 90 비트 펄스 간에 0 전위를 유지하지 않는다는 것은 다른 말로 0으로(to Zero) 복귀(Return)하지 않는다(Non)는 의미입니다.

## [전문가의 조언]

자주 출제되는 문제는 아닙니다. 이 문제에서는 Non Return To Zero(0으로 돌아가지 않는다), 즉 0 전위를 유지하지 않고 +V와 -V의 양극성 전압으로 펄스를 전송하는 NRZ의 원리만 기억하고 넘어가세요.

## 91 [전문가의 조언]

자주 출제되는 내용입니다. 광섬유 케이블과 관련해서는 세부적인 특징에 관한 문제까지 출제되니 자세하게 정리하세요.

## 광섬유 케이블(Optical Fiber Cable)

- 유리를 원료로 하여 제작된 가느다란 광섬유를 여러 가닥 묶어서 케이블의 형태로 만든 것으로, 광 케이블이라고도 한다.
- 데이터를 전기 신호가 아닌 빛으로 바꾸어 빛의 전반사 원리를 이용하여 전송한다.
- 유선 매체 중 가장 빠른 속도와 높은 주파수 대역폭을 제공한다.
- 넓은 대역폭을 제공하므로 데이터의 전송률이 높다.
- 대용량, 장거리 전송이 가능하다.
- 가늘고 가벼워 취급이 용이하다.
- 도청하기 어려워 보안성이 뛰어나다.
- 광섬유 케이블의 원료인 유리는 절연성이 좋아 전자 유도의 영향을 받지 않으므로(무유도성), 전자기적인 문제가 최소화되어 안정된 통신 및 누화 방지가 가능하다.
- 감쇠율이 적어 리피터의 설치 간격이 넓으므로 리피터의 소요가 적다.
- 설치 비용이 비싸지만 리피터의 소요가 적고, 대용량 전송이 가능하여 단위 비용은 저렴하다.
- 광섬유 간의 연결이 어려워 설치 시 고도의 기술이 필요하다.
- 전화 교환망뿐만 아니라 화상 전송, 근거리(LAN)와 광역 통신망, 군사용, 국가 간의 해저 케이블 등 거의 모든 분야에서 사용이 증가하고 있다.
- 광섬유 케이블은 원통형으로 코어(Core), 클래딩(Cladding), 재킷(Jacket)의 세 부분으로 구성된다.

## 92 [전문가의 조언]

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

## 펄스 코드 변조(PCM)

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

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

### 93 [전문가의 조언]

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

**HDLCL 프레임의 구조**

<b>플래그 (Flag)</b>	프레임의 시작과 끝을 나타내는 고유한 비트 패턴(01111110)
<b>주소부 (Address Field)</b>	<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>
<b>제어부 (Control Field)</b>	프레임의 종류를 식별하기 위해 사용. 제어부의 첫 번째, 두 번째 비트를 사용하여 구별함
<b>정보부 (Information Field)</b>	실제 정보 메시지가 들어 있는 부분으로, 송·수신측 간의 협의에 따라 길이와 구성이 정해짐
<b>FCS(Frame Check Sequence Field, 프레임 검사 순서 필드)</b>	프레임 내용에 대한 오류 검출을 위해 사용되는 부분으로, 일반적으로 CRC 코드가 사용됨

### 94 [전문가의 조언]

자주 출제되는 내용입니다. ARQ 종류는 물론 각각의 전송 원리도 파악하고 넘어가세요.

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

오류 발생 시 수신 측은 오류 발생을 송신 측에 통보하고, 송신 측은 오류 발생 블록을 재전송하는 모든 절차를 의미한다.

<b>정지-대기 ARQ</b>	<p>정지-대기(Stop-and-Wait) ARQ는 송신 측에서 한 개의 블록을 전송한 후 수신 측으로부터 응답을 기다리는 방식</p> <ul style="list-style-type: none"> <li>수신 측의 응답이 긍정 응답(ACK)이면 다음 블록을 전송하고, 부정 응답(NAK)이면 앞서 송신했던 블록을 재전송함</li> <li>블록을 전송할 때마다 수신 측의 응답을 기다려야 하므로 전송 효율이 가장 낮음</li> <li>오류가 발생한 경우 앞서 송신했던 블록만 재전송하면 되므로 구현 방법이 가장 단순함</li> </ul>				
<b>연속 ARQ</b>	<p>연속(Continuous) ARQ는 정지-대기 ARQ가 갖는 오버헤드를 줄이기 위해 연속적으로 데이터 블록을 보내는 방식으로, 수신 측에서는 부정 응답(NAK)만을 송신함</p> <table border="1"> <tr> <td>Go-Back-N ARQ</td><td> <ul style="list-style-type: none"> <li>여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내 오면 송신 측이 오류가 발생한 블록 이후의 모든 블록을 재전송함</li> <li>전송 오류가 발생하지 않으면 쉬지 않고 연속적으로 송신이 가능함</li> <li>오류가 발생한 부분부터 모두 재전송하므로 중복 전송의 단점이 있음</li> </ul> </td></tr> <tr> <td>선택적 재전송(Selective Repeat) ARQ</td><td> <ul style="list-style-type: none"> <li>여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내오면 송신 측이 오류가 발생한 블록만을 재전송함</li> <li>수신 측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리 회로와 큰 용량의 버퍼가 필요함</li> </ul> </td></tr> </table>	Go-Back-N ARQ	<ul style="list-style-type: none"> <li>여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내 오면 송신 측이 오류가 발생한 블록 이후의 모든 블록을 재전송함</li> <li>전송 오류가 발생하지 않으면 쉬지 않고 연속적으로 송신이 가능함</li> <li>오류가 발생한 부분부터 모두 재전송하므로 중복 전송의 단점이 있음</li> </ul>	선택적 재전송(Selective Repeat) ARQ	<ul style="list-style-type: none"> <li>여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내오면 송신 측이 오류가 발생한 블록만을 재전송함</li> <li>수신 측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리 회로와 큰 용량의 버퍼가 필요함</li> </ul>
Go-Back-N ARQ	<ul style="list-style-type: none"> <li>여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내 오면 송신 측이 오류가 발생한 블록 이후의 모든 블록을 재전송함</li> <li>전송 오류가 발생하지 않으면 쉬지 않고 연속적으로 송신이 가능함</li> <li>오류가 발생한 부분부터 모두 재전송하므로 중복 전송의 단점이 있음</li> </ul>				
선택적 재전송(Selective Repeat) ARQ	<ul style="list-style-type: none"> <li>여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내오면 송신 측이 오류가 발생한 블록만을 재전송함</li> <li>수신 측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리 회로와 큰 용량의 버퍼가 필요함</li> </ul>				
<b>적응적 ARQ</b>	<p>적응적(Adaptive) ARQ는 전송 효율을 최대로 하기 위해 데이터 블록의 길이를 채널의 상태에 따라 그때그때 동적으로 변경하는 방식</p> <ul style="list-style-type: none"> <li>전송 효율이 제일 좋음</li> <li>제어 회로가 매우 복잡하고 비용이 많이 들어 현재 거의 사용되지 않고 있음</li> </ul>				

### 95 [전문가의 조언]

코덱의 기능을 묻는 문제가 종종 출제되는데 코덱은 모뎀과 기능이 유사하여 혼동될 수 있습니다. 두 장치의 기능을 명확히 구분해서 알아두세요.

- **모뎀(MODEM)** : 디지털 데이터를 아날로그 통신 회선에 적합한 아날로그 신호로 변환하는 변조(MODulation)와 그 반대의 복조(DEMODulation) 과정을 수행함
- **코덱(CODEC)** : 아날로그 데이터를 디지털 통신 회선에 적합한 디지털 신호로 변환(CODer)하거나 그 반대의 과정(DECoder)을 수행함

### 96 [전문가의 조언]

셀룰러 시스템의 특징과 관련된 문제들이 종종 출제되고 있습니다. 셀룰러 시스템의 특징을 정리하세요.

**셀룰러 시스템의 특징**

- **주파수 재사용(Frequency Reuse)** : 인접하지 않는 셀은 같은 주파수를 사용함으로써 통화량을 늘리고, 회선의 사용을 극대화할 수 있음
- **핸드오프(Hand-off, Hand-over)** : 가입자가 서비스 중인

기지국 영역을 벗어나 다른 기지국으로 이동할 때, 통화가 단절되지 않도록 통화 채널을 자동으로 전환하는 기능

- **로밍(Roaming) 서비스** : 가입자가 자신의 홈 교환국을 벗어나 타 교환국에 있어도 서비스를 받을 수 있는 것을 의미함. 로밍은 한 사업자의 교환국 사이에서만 아니라 사업자 간, 국가 간에도 가능함

#### 97 [전문가의 조언]

문제와 보기가 동일하게 출제되었던 문제입니다. 토큰(Token)은 회선에 대한 사용 권한을 의미한다는 것만 기억해 두세요.

#### 98 [전문가의 조언]

망의 구성 형태와 관련해서는 주로 무슨 망인지를 묻는 문제가 출제됩니다. 망의 모양을 염두에 두고 특징을 읽어보면 쉽게 기억됩니다.

**망(Network)의 구성 형태**

<b>성형 (Star)</b>	중앙에 중앙 컴퓨터가 있고, 이를 중심으로 단말장치들이 연결되는 중앙 집중식의 네트워크 구성 형태
<b>링형 (Ring)</b>	컴퓨터와 단말장치들을 서로 이웃하는 것끼리 포인트 투 포인트(Point-to-Point) 방식으로 연결시킨 형태
<b>버스형 (Bus)</b>	한 개의 통신 회선에 여러 대의 단말장치가 연결되어 있는 형태
<b>계층형 (Tree)</b>	중앙 컴퓨터와 일정 지역의 단말장치까지는 하나의 통신 회선으로 연결시키고, 이웃하는 단말장치는 일정 지역 내에 설치된 중간 단말장치로부터 다시 연결시키는 형태
<b>망형 (Mesh)</b>	모든 지점의 컴퓨터와 단말장치를 서로 연결한 형태로, 노드의 연결성이 높음

#### 99 데이터를 전송하고 확인 신호(ACK)를 보내는 단계는 정보의 전송 단계입니다.

**[전문가의 조언]**

단순히 전송 제어 절차의 순서를 묻는 문제가 자주 출제됩니다. '회로 연결 → 데이터 링크 설정 → 데이터 전송 → 데이터 링크 종료 → 회선 절단' 순으로 진행되는 전송 제어 절차의 순서를 꼭 기억하고 넘어가세요.

#### 100 [전문가의 조언]

종종 출제되는 문제입니다. TCP/IP 상에서 운용되는 프로토콜에 대해 정리하세요.

**TCP/IP의 구조**

TCP/IP는 응용 계층, 전송 계층, 인터넷 계층, 링크 계층으로 이루어져 있다.

OSI	TCP/IP	기능
응용 계층, 표현 계층, 세션 계층	응용 계층	• 응용 프로그램 간의 데이터 송·수신 제공 • TELNET, FTP, SMTP, SNMP 등
전송 계층	전송 계층	• 호스트들 간의 신뢰성 있는 통신 제공 • TCP, UDP
네트워크 계층	인터넷 계층	• 데이터 전송을 위한 주소 지정, 경로 설정을 제공 • IP, ICMP, IGMP, ARP, RARP
데이터 링크 계층, 물리 계층	링크 계층	• 실제 데이터(프레임)를 송·수신하는 역할 • Ethernet, IEEE 802, HDLC, X.25, RS-232C 등