



시험에 나오는것만 공부한다!

시나공시리즈

기출문제 & 정답 및 해설  
2020년 1회 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. 데이터베이스 설계 단계 중 물리적 설계 단계와 거리가 먼 것은?

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

2. 해싱(Hashing)에서 한 개의 레코드를 저장할 수 있는 공간을 의미하는 것은?

- ① Bucket
- ② Synonym
- ③ Slot
- ④ Collision

3. 다음 자료를 버블 정렬을 이용하여 오름차순으로 정렬하고자 할 경우 1회전을 수행한 결과는?

- 9, 4, 5, 1, 3
- ① 4, 5, 1, 3, 9
  - ② 1, 3, 4, 5, 9
  - ③ 4, 1, 3, 5, 9
  - ④ 1, 3, 9, 4, 5

4. SQL 언어의 데이터 제어어(DCL)에 해당하는 것은?

- ① SELECT
- ② INSERT
- ③ UPDATE
- ④ GRANT

5. 데이터베이스 관리 시스템(DBMS)의 필수 기능에 해당하지 않는 것은?

- ① 정의 기능(Definition Facility)
- ② 조작 기능(Manipulation Facility)
- ③ 제어 기능(Control Facility)
- ④ 사전 기능(Dictionary Facility)

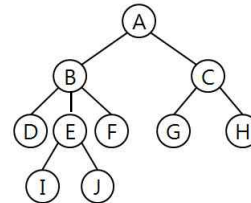
6. 릴레이선의 특징으로 옳지 않은 것은?

- ① 한 릴레이션에 포함된 튜플들은 모두 상이하다.
- ② 모든 속성 값은 세분화가 가능해야 하므로 원자값 이어서는 안된다.
- ③ 한 릴레이션에 포함된 튜플 사이에는 순서가 없다.
- ④ 한 릴레이션을 구성하는 속성 사이에는 순서가 없다.

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

- ① E-R 모델
- ② 관계 데이터 모델
- ③ 계층 데이터 모델
- ④ 네트워크 데이터 모델

8. 다음 그림에서 단말 노드(Terminal Node)의 개수는?



- ① 3
- ② 4
- ③ 6
- ④ 10

9. 다음의 중위(Infix) 표기 방식을 전위(Prefix) 표기 방식으로 옮겨 변환한 것은?

A \* B + C - D / E

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

10. 뷰(View)에 대한 설명으로 옳지 않은 것은?

- ① 실제 저장된 데이터 중에서 사용자가 필요한 내용만을 선별해서 볼 수 있다.
- ② 데이터 접근 제어로 보안을 제공한다.
- ③ 뷰를 제거할 때는 DELETE문을 사용한다.
- ④ 실제로는 존재하지 않는 가상의 테이블이다.

11. 다음의 조건을 모두 만족하는 정규형은?

모든 도메인은 원자 값이고, 기본키가 아닌 모든 속성들이 기본키에 대해 완전 함수 종속적이며, 이행적 함수 종속 관계는 제거되었다.

- ① 제 1 정규형
- ② 제 2 정규형
- ③ 제 3 정규형
- ④ 비정규 릴레이션

12. 다음 설명에 해당되는 것은?

It is a collection of meta data describing the structure and constraint of a database. It defines data entities, attribute, relations, and constraints on data manipulation.

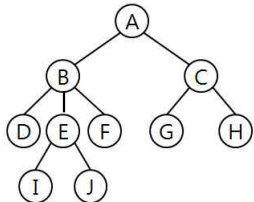
- ① Bubble Sort
- ② Schema
- ③ Key
- ④ DataWare House

13. 데이터베이스 설계 단계 중 논리적 설계 단계에서의 수행사항이 아닌 것은?

- ① 논리적 데이터 모델로 변환
- ② 트랜잭션 인터페이스 설계
- ③ 저장 레코드 양식 설계
- ④ 개념 스키마의 평가 및 정제

14. 다음 SQL 문에서 테이블 생성에 사용되는 문장은?

- ① DROP
- ② INSERT
- ③ SELECT
- ④ CREATE

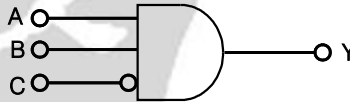
15. 데이터베이스의 설계 단계 순서로 옳은 것은?  
① 개념적 설계 → 물리적 설계 → 논리적 설계  
② 개념적 설계 → 논리적 설계 → 물리적 설계  
③ 물리적 설계 → 개념적 설계 → 논리적 설계  
④ 논리적 설계 → 개념적 설계 → 물리적 설계
16. 로킹에 대한 설명으로 옳지 않은 것은?  
① 로킹의 대상이 되는 객체의 크기를 로킹 단위라고 한다.  
② 로킹은 주요 데이터의 접근을 상호 배타적으로 하는 것이다.  
③ 로킹 단위가 크면 병행성 수준이 높아진다.  
④ 로킹 단위가 작아지면 로킹 오버헤드가 증가한다.
17. E-R 모델에 관한 설명으로 옳지 않은 것은?  
① 개체 타입은 타원, 관계 타입은 사각형, 속성은 선으로 표현한다.  
② 개체 타입과 이들 간의 관계 타입을 이용한다.  
③ E-R 모델에서는 데이터를 개체, 관계, 속성으로 묘사한다.  
④ 현실 세계가 내포하는 의미들이 포함된다.
18. A, B, C, D의 순서로 정해진 입력자료를 스택에 입력하였다가 출력한 결과가 될 수 없는 것은?(단, 왼쪽부터 먼저 출력된 순서이다.)  
① C, B, A, D                      ② C, D, A, B  
③ B, A, D, C                      ④ B, C, D, A
19. 비선형 구조에 해당하는 것은?  
① 그래프                              ② 데크  
③ 스택                                ④ 큐
20. 다음 그림에서 트리의 차수는?  
  
① 3                                      ② 4  
③ 5                                      ④ 10

제2과목 전자계산기구조

21. 보조기억장치의 일반적인 특징 중 틀린 것은?  
① 읽고 쓰는 속도가 느리다.  
② 기억용량을 크게 하기가 용이하다.  
③ 전원공급이 중단되면 기억된 내용이 모두 지워진다.  
④ 기억용량의 상대적인 가격이 주기억장치보다 저렴하다.
22. 버스(Bus)를 구성하는데 사용할 수 있는 논리회로는?  
① Encoder                              ② Multiplexer  
③ Counter                               ④ Comparator
23. ALU의 위치와 기능이 바르게 나열된 것은?  
① CPU, 산술논리연산                  ② ROM, 산술논리연산  
③ CPU, 주소지정                       ④ ROM, 주소지정
24. 전자계산기에서 어떤 특수한 상태가 발생할 때 그것이 원인이 되어 현재 실행하고 있는 프로그램이 일시 중단되고, 그 특수한 상태를 처리하는 프로그램으로 옮겨져 처리한 후 다시 원래의 프로그램을 처리하는 것은?

- ① 인터럽트                              ② 다중 처리  
③ 시분할 시스템                       ④ 다중 프로그램
25. 8진수인 다음식의 연산값은?  

$751_8 + 154_8$

  
① 2151                                  ② 2152  
③ 1251                                  ④ 1125
26. Op-Code가 8비트일 때 생성 될 수 있는 명령어의 수는?  
①  $2^7-1$                                     ②  $2^7$   
③  $2^8$                                       ④  $2^8-1$
27. 채널의 기능이 아닌 것은?  
① 입출력 명령 해독  
② 입출력 명령 지시  
③ 입출력 데이터 저장  
④ 입출력 데이터 실행
28. 비수치적 연산에 속하지 않는 것은?  
① 사칙 연산                              ② 논리적 연산  
③ 로테이트(Rotate)                   ④ 논리적 시프트(Shift)
29. 트랩(Trap)의 발생 원인으로 옳은 것은?  
① 0으로 나눌 때  
② 정해진 시간이 지났을 때  
③ 정보 전송이 끝났음을 알릴 때  
④ 입출력장치가 데이터의 전송을 요구할 때
30. 다음 게이트의 출력은? (단, A = B = S = 1)  
  
① 0                                        ② 1  
③ AB                                      ④ S
31. 1개의 Full Adder를 구성하기 위해서는 최소 몇 개의 Half Adder가 있어야 하는가?  
① 1                                        ② 2  
③ 3                                        ④ 4

32. 마이크로 오퍼레이션에 관한 설명 중 옳은 것은?  
① 마이크로 오퍼레이션을 동기 시키는 방법으로 동기 고정식과 동기 가변식이 있다.  
② 동기 고정식은 CPU 시간의 효율적 이용은 가능하나 제어가 복잡하다.  
③ 동기 가변식은 CPU 시간의 낭비를 초래하지만 제어회로가 간단하다.  
④ 마이크로 사이클은 마이크로 오퍼레이션과 무관하다.
33. 명령어의 형식 가운데 연산에 사용된 모든 피연산자 값을 상실하는 명령어 형식은?  
① 3-주소 형식 명령어                  ② 2-주소 형식 명령어  
③ 1-주소 형식 명령어                  ④ 0-주소 형식 명령어
34. 컴퓨터 명령어 실행 주기 중에서 인스트럭션의 종류에 대한 판단이 이루어지는 상태는?  
① Fetch                                   ② Execute  
③ Interrupt                               ④ Indirect

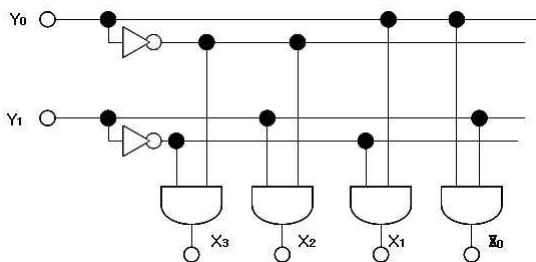
35. 분기 명령이 수행될 때 다음의 레지스터 중 그 내용이 바뀌는 것은?

- ① 누산기
- ② 프로그램 카운터
- ③ 인덱스 레지스터
- ④ 메모리 어드레스 레지스터

36. 중앙처리장치와 주기억장치의 속도 차이가 현저할 때 인스트럭션의 수행속도가 주기억장치에 제한을 받지 않고 중앙처리장치의 속도로 수행되도록 하는 기억장치는?

- ① 캐시 메모리
- ② 인스트럭션 버퍼
- ③ CAM
- ④ 제어기억장치

37. 다음 논리도(Logic Diagram)에서  $Y_0$ 에 1,  $Y_1$ 에 0이 입력되었을 때, 1을 출력하는 단자는?



- ①  $X_1$
- ②  $X_1$ 과  $X_2$
- ③  $X_2$
- ④  $X_2$ 와  $X_3$

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

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

39. 기억장치 계층 구조 상 접근 속도가 가장 빠른 것은?

- ① ROM
- ② RAM
- ③ Register
- ④ Magnetic Disk

40. 중앙 처리 장치를 통하지 않고 직접 주기억장치를 접근하여 입출력을 하는 방식으로, 한 번에 한 블록씩 전송하는 방법은?

- ① DMA
- ② 인터럽트 입출력
- ③ 고정 채널 제어기 입출력
- ④ 가변 채널 제어기 입출력

### 제3과목 시스템분석설계

41. 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

42. 코드 설계 단계 중 다음 설명에 해당하는 것은?

코드 대상 항목에 대하여 설계된 코드의 사용이 컴퓨터 처리에 한정되는가, 해당 업무에만 한정되는가, 관련 부문의 업무에 공통으로 사용되는가, 기업 전체에 사용되는가, 관련 있는 타기업 또는 공공 기관이 공통으로 사용할 것인지 등을 확정해야 한다.

- ① 사용 범위의 결정
- ② 코드 목적의 명확화
- ③ 코드 대상의 특성 분석
- ④ 코드 부여 방식 결정

43. 데이터 파일의 종류 중 마스터 파일을 갱신 또는 조회하기 위해 작성하는 파일은?

- ① Trailer File
- ② Transaction File
- ③ Summary File
- ④ Source Data File

44. 자료 사전(Data Dictionary)에서 반복을 의미하는 기호는?

- ① +
- ② { }
- ③ [ ]
- ④ ( )

45. 시스템의 특성 중 제어성과 가장 관련 깊은 것은?

- ① 최종 목표에 도달하고자 하는 특성
- ② 시스템 변화에 스스로 대처할 수 있는 특성
- ③ 정해진 목표를 달성하기 위해 오류가 발생하지 않도록 사태를 감시하는 특성
- ④ 관련된 다른 시스템과 상호 의존 관계로 통합되는 특성

46. 다음의 소프트웨어 개발 주기 모형에 대한 설명에 해당하는 것은?

하향식 생명 주기 모형으로 각 단계가 끝나는 시점에서 확인, 검증, 검사를 거쳐 다음 단계로 넘어가거나 이전 단계로 환원하면서 구현 및 운영 단계에 이르는 생명 주기 모형이다.

- ① 단계적 모형
- ② 폭포수 모형
- ③ 구조적 모형
- ④ 객체지향적 모형

47. 모듈 내부의 모든 기능 요소들이 단일한 목적을 위해 수행하는 경우의 응집도는?

- ① Coincidental Cohesion
- ② Functional Cohesion
- ③ Procedural Cohesion
- ④ Temporal Cohesion

48. 다음은 어떤 종류의 코드 오류(Error)인가?

9 8 7 6 5 → 9 8 7 6 4

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

49. 입력된 자료가 처리되어 일단 출력된 후 이용자를 거쳐 다시 재입력되는 방식으로 공과금, 보험료 징수 등의 지로용지를 처리하는데 사용되는 입력 방식은?

- ① 집중 매체화형 시스템
- ② 턴 어라운드 시스템
- ③ 분산 매체화형 시스템
- ④ 직접 입력 시스템

50. 자료 흐름도의 구성 요소 중 대상 시스템의 외부에 존재하는 사람이나 조직체를 나타낸 것은?

- ① Process
- ② Data Flow
- ③ Data Store
- ④ Terminator

51. 다음 표와 같이 시스템이 운영될 때 시스템의 평균 수리 시간(MTTR)은? (단, 상태에서 R = 가동중, S = 고장중이다.)

시간	120	100	280	60	200	80
상태	R	S	R	S	R	S

- ① 80시간
- ② 200시간
- ③ 120시간
- ④ 140시간

52. 프로세스의 표준 처리 패턴 중 어떤 파일에서 특정한 조건에 만족하는 정보를 추출해 내는 처리는?

- ① Matching                      ② Merge  
③ Extract                        ④ Distribution

53. 시스템 개발 시 문서화의 효과에 대한 설명으로 가장 거리가 먼 것은?

- ① 시스템 개발 단계에서의 요식적 행위이다.  
② 효율적인 소프트웨어 개발관리가 용이하다.  
③ 시스템 개발 중 추가 변경에 따른 혼란을 방지한다.  
④ 시스템 개발 후에 유지보수가 용이하다.

54. 모듈의 결합도는 설계에 대한 품질 평가 방법의 하나로써 두 모듈 간의 상호 의존도를 측정하는 것이다. 다음 중 설계 품질이 가장 좋은 결합도는?

- ① Common Coupling            ② Data Coupling  
③ Control Coupling            ④ Content Coupling

55. 순서도와는 달리 논리 기술에 중점을 두고 상자도형을 이용한 설계 도구로 순차, 선택, 반복 등의 제어 구조를 표현하는 도구는?

- ① Waterfall 모델                ② N-S 차트  
③ PAD                            ④ HCP

56. 객체지향기법에서 객체가 메시지를 받아 실행해야 할 때 객체의 구체적인 연산을 정의한 것은?

- ① Instance                        ② Message  
③ Class                            ④ Method

57. 중량, 용량, 거리, 크기, 면적 등의 물리적 수치를 직접 코드에 적용시키는 코드 방식은?

- ① 순차 코드(Sequence Code)  
② 표의 숫자 코드(Significant Digit Code)  
③ 블록 코드(Block Code)  
④ 기호 코드(Mnemonic Code)

58. 객체지향시스템 분석에서 사건들을 시나리오로 작성하여 각 시나리오마다 사건 추적도를 그리고 사건 흐름 다이어그램을 작성하는 단계는?

- ① 객체 모형화                    ② 동적 모형화  
③ 기능 모형화                    ④ 사양서 작성

59. 색인 순차 파일(Index Sequential File)에서 데이터 레코드 중의 key 항목만을 모아서 기록하는 인덱스 부분에 해당하지 않는 것은?

- ① Master Index                  ② Cylinder Index  
③ Track Index                    ④ Data Index

60. 마스터 파일의 내용을 변동 파일에 의해 추가, 삭제, 수정 등의 작업을 하여 새로운 파일을 만드는 처리 패턴은?

- ① Merge                            ② Update  
③ Matching                        ④ Conversion

#### 제4과목 운영체제

61. 네트워크를 이용하여 서비스를 요구/제공할 수 있다. 여러 가지 서비스를 요구하는 측을 일컫는 용어는?

- ① Host                              ② Client  
③ Server                            ④ Backbone

62. 프로세스 스케줄링 기법 중 비선점 방식의 SJF에 선점 방식을 도입하여, 현재 실행중인 프로세스 보다 잔여 처리 시간이 짧은 프로세스가 준비 큐에 생기면 실행중인 프로세스를 선점하여 더 짧은 프로세스를 실행시키는 방식은?

- ① 기한부 스케줄링              ② SRT 스케줄링  
③ HRN 스케줄링                ④ 다단계 큐 스케줄링

63. 프로세스에 할당된 페이지 프레임 수가 증가하면 페이지 부재의 수가 감소하는 것이 당연하지만 페이지 프레임 수가 증가할 때 현실적으로 페이지 부재가 더 증가하는 모순(Anomaly) 현상과 가장 관계 있는 페이지 교체 기법은?

- ① LRU                              ② LFU  
③ FIFO                            ④ Optimal

64. 파일의 보호 방법 중 틀린 것은?

- ① 암호화                          ② 접근 제어  
③ 패스워드                        ④ 파일 공유

65. 3개의 페이지 프레임을 갖는 시스템에서 페이지 참조 순서가 1, 2, 1, 0, 4, 1, 3일 경우 FIFO 알고리즘에 의한 최종 페이지 대처 결과는?

- ① 1, 4, 2                            ② 1, 2, 0  
③ 4, 1, 3                            ④ 4, 1, 0

66. 한 프로세스가 다른 프로세스보다 우선순위 등이 낮아 기다리게 되는 경우, 한번 양보하거나 일정 시간이 지나면 우선순위를 한 단계씩 높여 줌으로써 오래 기다린 프로세스를 고려하여 무기한 지연을 해결하는 방법은?

- ① Aging                            ② Priority  
③ Recovery                        ④ Avoidance

67. RR(Round Robin) 방식에 관한 설명으로 옳지 않은 것은?

- ① 시분할 시스템을 위해 고안된 방식이다.  
② 시스템이 사용자에게 적합한 응답시간을 제공해 주는 대화식 시스템에 유용하다.  
③ 시간 할당량이 클 경우 FCFS 기법과 같아지고, 시간 할당량이 작을 경우 문맥 교환 및 오버헤드가 자주 발생될 수 있다.  
④ 프로세스에게 이미 할당된 프로세스를 강제로 빼앗을 수 없고, 그 프로세스의 사용이 종료된 후에 스케줄링 해야 하는 방법을 택하고 있다.

68. 기억장치의 분할 방식 중 틀린 것은?

- ① 분산 분할                        ② 고정 분할  
③ 단일 분할                        ④ 동적 분할

69. 교착상태 발생의 필요 조건에 해당하는 것으로 나열된 것은?

- ㉠ 상호 배제(Mutual Exclusion) 조건  
㉡ 환형 대기(Circular Wait) 조건  
㉢ 선점(Preemption) 조건  
㉣ 비선점(Non-Preemption) 조건  
㉤ 재진입 가능(Reentrant) 조건  
㉥ 점유와 대기(Hold and Wait) 조건

- ① ㉠, ㉡, ㉣, ㉥                    ② ㉠, ㉣, ㉤, ㉥  
③ ㉡, ㉢, ㉤, ㉥                    ④ ㉠, ㉢, ㉣, ㉥

70. 파일 디스크립터(Descriptor)가 가지고 있는 정보로 틀린 것은?

- ① 파일의 구조  
② 접근 제어 정보  
③ 파일의 백업 방법  
④ 보조기억장치상의 파일의 위치





89. FM에서 변조지수가 10, 변조신호의 최고 주파수를 4kHz라 할 때 소오 대역폭(kHz)은?

- ① 8                                      ② 40  
③ 88                                    ④ 400

90. TCP 전송 계층 프로토콜을 사용하여 통신하는 데 이용되는 소켓을 무엇이라 하는가?

- ① 스트림 소켓                      ② 데이터그램 소켓  
③ raw 소켓                        ④ 리시빙 소켓

91. 서로 다른 기기들 간의 데이터 교환을 원활하게 수행할 수 있도록 표준화시켜 놓은 통신 규약을 무엇이라 하는가?

- ① 클라이언트                      ② 터미널  
③ 링크                                ④ 프로토콜

92. ARQ(Automatic Repeat reQuest) 방식에 해당하지 않는 것은?

- ① Stop and Wait ARQ              ② Adaptive ARQ  
③ Receive Ready ARQ            ④ Go back N ARQ

93. OSI 7계층 모델에서 기계적, 전기적, 절차적 특성을 정의한 계층은?

- ① 전송 계층                        ② 데이터링크 계층  
③ 물리 계층                        ④ 표현 계층

94. Shannon의 표본화 정리에 의하면 보내려는 신호 성분 중 최고 주파수의 최소 몇 배 이상으로 표본을 행하면 원 신호를 충실하게 재현시킬 수 있는가?

- ① 1                                      ② 2  
③ 4                                      ④ 8

95. 위상변화를 작게 하면서 반송파의 진폭도 바뀌 정보 전송률을 높이려는 변조방식은?

- ① ASK                                ② FSK  
③ PSK                                ④ QAM

96. 다음 내용이 설명하고 있는 LAN의 매체 접근 제어방식은?

- 버스 또는 트리 토폴로지에서 가장 많이 사용된다.
- 전송하고자 하는 스테이션이 전송 매체의 상태를 감지하다가 유히(idle) 상태인 경우 데이터를 전송하고, 전송이 끝난 후에도 계속 매체의 상태를 감지하여 다른 스테이션과의 충돌 발생 여부를 감시한다.

- ① CSMA/CD                        ② Token bus  
③ Token ring                      ④ Slotted ring

97. HDLC 프레임의 구조가 순서대로 옳은 것은?

- ① 플래그 → 주소부 → 제어부 → 정보부 → FCS → 플래그  
② 플래그 → 제어부 → FCS → 정보부 → 주소부 → 플래그  
③ 플래그 → 주소부 → 정보부 → FCS → 제어부 → 플래그  
④ 플래그 → 제어부 → FCS → 주소부 → 정보부 → 플래그

98. 전송 효율을 최대한 높이려고 데이터 블록의 길이를 동적으로 변경시켜 전송하는 ARQ 방식은?

- ① Adaptive ARQ                    ② Stop-And-Wait ARQ  
③ Positive ARQ                    ④ Distrbuted ARQ

99. PCM 방식의 데이터 전송 순서로 맞는 것은?

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

100. Link State 방식의 라우팅 프로토콜은?

- ① RIP                                    ② RIP V2  
③ IGRP                                ④ OSPF

정답 및 해설

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

1 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

2 **[전문가의 조언]**  
종종 출제되는 내용입니다. 해싱의 개요 그리고 해시 테이블과 관련된 용어를 정리하고 넘어가세요.

**해싱(Hashing)**

- 해싱은 Hash Table이라는 기억 공간을 할당하고, 해시 함수(Hash Function)를 이용하여 레코드 키에 대한 Hash Table 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

3 버블 정렬은 주어진 파일에서 인접한 두 개의 레코드 키 값을 비교하여 그 크기에 따라 레코드 위치를 서로 교환하는 정렬 방식으로 다음과 같은 과정으로 진행됩니다.

초기 상태 : 9, 4, 5, 1, 3

① 1회전 : 4, 9, 5, 1, 3 → 4, 5, 9, 1, 3 → 4, 5, 1, 9, 3 → 4, 5, 1, 3, 9

② 2회전 : 4, 5, 1, 3, 9 → 4, 1, 5, 3, 9 → 4, 1, 3, 5, 9

③ 3회전 : 1, 4, 3, 5, 9 → 1, 3, 4, 5, 9

④ 4회전 : 1, 3, 4, 5, 9

**[전문가의 조언]**  
주요 정렬 알고리즘의 정렬 원리는 실기 시험에서도 출제되는 내용입니다. 이 문제를 통해 버블 정렬의 정렬 과정을 확실히 이해하고 넘어가세요.

4 GRANT만 DCL이고 나머지는 DML입니다.

**[전문가의 조언]**  
자주 출제되는 내용입니다. 각 언어의 특징을 파악하고 각각에는 어떤 명령들이 있는지 구분할 수 있도록 공부하세요.

**데이터베이스 언어**

- **DDL(데이터 정의어)**
  - DDL(Data Define Language)은 SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의하거나 변경 또는 삭제할 때 사용하는 언어이다.

- 데이터베이스 관리자나 데이터베이스 설계자가 사용한다.
- 명령어 : CREATE, ALTER, DROP
- **DML(데이터 조작어)**
  - DML(Data Manipulation Language)은 데이터베이스 사용자가 응용 프로그램이나 질의어를 통하여 저장된 데이터를 실질적으로 처리하는 데 사용되는 언어이다.
  - 데이터베이스 사용자와 데이터베이스 관리 시스템 간의 인터페이스를 제공한다.
  - 명령어 : SELECT, INSERT, DELETE, UPDATE
- **DCL(데이터 제어어)**
  - DCL(Data Control Language)은 데이터의 보안, 무결성, 회복, 병행 수행 제어 등을 정의하는 데 사용되는 언어이다.
  - 데이터베이스 관리자가 데이터 관리를 목적으로 사용한다.
  - 명령어 : COMMIT, ROLLBACK, GRANT, REVOKE

## 5 [전문가의 조언]

DBMS의 필수 기능 및 각각의 특징을 묻는 문제가 자주 출제되고 있습니다. DBMS의 필수 기능 3가지와 각각의 특징을 꼭 기억하세요.

**DBMS의 필수 기능**

<b>정의 (조직)</b>	데이터베이스에 저장될 데이터의 유형(Type)과 구조에 대한 정의, 이용 방식, 제약 조건 등을 명시하는 기능
<b>조작</b>	데이터 검색, 갱신, 삽입, 삭제 등을 체계적으로 처리하기 위해 데이터 접근 수단 등을 정하는 기능
<b>제어</b>	데이터의 정확성과 안전성을 유지하기 위한 무결성, 보안 및 권한 검사, 병행수행 제어 등의 기능을 정하는 기능

- 6 한 릴레이션에서 나타난 모든 속성 값은 논리적으로 더 이상 분해할 수 없는 원자값이어야 합니다. 예를 들어 성별 속성에는 '남' 또는 '여' 값만을 가져야지 '남여' 또는 '여남' 값을 가져서는 안 됩니다.

## [전문가의 조언]

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

## 릴레이션의 특징

<학생>

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

- 한 릴레이션에 포함된 튜플들은 모두 상이하다.  
예) <학생> 릴레이션을 구성하는 홍길동 레코드는 홍길동에 대한 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 삽입하거

나, 기존 학생에 대한 레코드를 삭제함으로써 테이블은 내용 면에서나 크기 면에서 변하게 된다.

- 릴레이션 스키마를 구성하는 속성들 간의 순서는 중요하지 않다.  
예) 학번, 이름 등의 속성을 나열하는 순서가 이름, 학번순으로 바뀌어도 데이터 처리에는 전혀 문제가 되지 않는다.
- 속성의 유일한 식별을 위해 속성의 명칭은 유일해야 하지만, 속성을 구성하는 값은 동일한 값이 있을 수 있다.  
예) 각 학생의 학년을 기술하는 속성인 '학년'은 다른 속성 명들과 구분되어 유일해야 하지만 '학년' 속성에는 2, 1, 2, 4 등이 입력된 것처럼 동일한 값이 있을 수 있다.
- 릴레이션을 구성하는 튜플을 유일하게 식별하기 위해 속성들의 부분집합을 키(Key)로 설정한다.  
예) <학생> 릴레이션에서는 '학번'이나 '성명'이 튜플들을 구분하는 유일한 값인 키가 될 수 있다.
- 속성은 더 이상 쪼갤 수 없는 원자값만을 저장한다.  
예) '학년'에 저장된 1, 2, 4 등은 더 이상 세분화할 수 없다.

## 7 [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 개념입니다.

- 8 터미널 노드(단말 노드)란 자식이 하나도 없는 노드를 말합니다. 제시된 그림에서 자식이 없는 터미널 노드는 D, I, J, F, G, H로 총 6개입니다.

## [전문가의 조언]

트리의 차수(Degree)를 묻는 문제도 자주 출제됩니다. 함께 알아두세요.

## 트리의 차수(Degree)

가장 차수가 많은 노드의 차수를 트리의 차수라고 한다. 이 문제에서 가장 차수가 많은 노드는 B이고, B의 차수는 3이므로 트리의 차수는 3이다.

- 9 중위 표기식(Infix)을 전위 표기식(Prefix)으로 변경하는 과정과 결과는 다음과 같습니다.

- ① 연산 우선순위에 따라 괄호로 묶습니다.

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

- ② 연산자를 해당 괄호의 앞(왼쪽)으로 옮깁니다.

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



## 1회

- ③ 괄호를 제거합니다.  
- + \* A B C / D E

**[전문가의 조언]**

수식의 표기법에서는 이 문제와 같이 표기된 수식의 표기를 변경하라는 문제가 주로 출제됩니다. 중위식에서 후위식, 전위식 또는 반대의 관계로 표기할 수 있도록 연습하세요.

- 10 뷰를 제거할 때는 DROP문을 사용합니다.

**[전문가의 조언]**

중요해요! 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.

- 11 • 모든 도메인은 원자 값이므로 제1정규형을 만족합니다.  
• 키가 아닌 모든 속성들이 기본키에 대해 완전 함수 종속이므로 제2정규형을 만족합니다.  
• 이행적 함수 종속 관계가 제거 되었으므로 제3정규형을 만족합니다.

**[전문가의 조언]**

중요해요! 정규화와 관련해서는 특정 단계에서 다음 단계로 정규화하기 위해 필요한 작업을 묻는 문제가 자주 출제됩니다. 정규화 단계별로 수행해야 할 작업을 꼭 암기하세요.

**정규화 과정****비정규 릴레이션**

↓ 도메인이 원자값

**1NF**

↓ 부분적 함수 종속 제거

**2NF**

↓ 이행적 함수 종속 제거

**3NF**

↓ 결정자이면서 후보키가 아닌 것 제거

**BCNF**

↓ 다치 종속 제거

**4NF**

↓ 조인 종속성 이용

**5NF****정규화 단계 암기 요령**

두부를 좋아하는 정규화가 두부가게에 가서 가게에 있는 두부를 다 달라고 말하니 주인이 깜짝 놀라며 말했다.

**두부이걸다줘 ⇔ 도부이걸다조**

도메인이 원자값

부분적 함수 종속 제거

이행적 함수 종속 제거

결정자이면서 후보키가 아닌 것 제거

다치 종속 제거

조인 종속성 이용

- 12 해석 : 이것은 데이터베이스의 구조와 제약사항이 묘사된 메타 데이터의 집합이다. 이것은 데이터 개체, 속성, 관계와 데이터를 조작할 때의 제약사항들을 정의한다.

**[전문가의 조언]**

스키마의 정의와 특징은 매우 중요합니다. 꼭 정리하고 넘어가세요.

**스키마(Schema)**

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

- 13 저장 레코드 양식 설계는 물리적 설계 단계에서 수행하는 작업입니다.

**[전문가의 조언]**

중중 출제되는 내용입니다. 논리적 설계 단계에서 수행하는 작업들을 파악해 두세요.

**논리적 설계**

- 논리적 설계 단계에서는 현실 세계에서 발생하는 자료 형태를 컴퓨터가 처리할 수 있는 물리적 저장장치에 저장할 수 있도록 변환하기 위해 특정 DBMS가 지원하는 논리적 자료 구조로 변환시키는 과정이다.

- 개념 세계의 데이터를 필드로 기술된 데이터 타입과 이 데이터 타입들 간의 관계로 표현되는 논리적 구조의 데이터로 모델화한다.
- 개념적 설계가 개념 스키마를 설계하는 단계라면 논리적 설계에서는 개념 스키마를 평가하고 DBMS에 따라 서로 다른 논리적 스키마를 설계하는 단계이다.
- 트랜잭션의 인터페이스를 설계한다.
- 관계형 데이터베이스라면 테이블을 설계하는 것이다.

14 SQL에서 테이블 생성에 사용하는 명령어는 CREATE입니다.

**[전문가의 조언]**

데이터 정의어(DDL)의 3가지 유형을 숙지하세요.

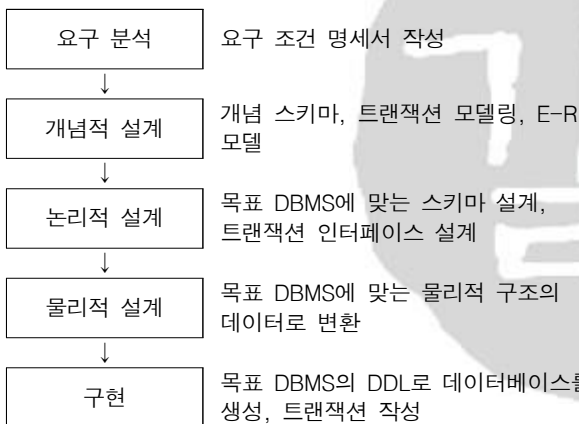
**데이터 정의어**

CREATE	Schema, Domain, Table, View, Index 정의 (생성)
ALTER	Table에 대한 정의 변경
DROP	Schema, Domain, Table, View, Index 삭제

15 **[전문가의 조언]**

중요해요! 데이터베이스의 설계 순서를 꼭 기억하고, 각각의 단계는 무엇을 하는 단계인지 개략적으로 알아두면 됩니다.

**데이터베이스 설계 순서**



16 로킹(Locking)은 하나의 트랜잭션이 데이터를 액세스 하는 동안 다른 트랜잭션이 그 데이터 항목을 액세스할 수 없도록 잠그는 것입니다. 로킹 단위가 크면 로크 수가 적어 관리하기 쉽지만 병행성 수준이 낮아지고, 로킹 단위가 작으면 로크 수가 많아 관리하기는 복잡하지만 병행성 수준이 높아집니다.

**[전문가의 조언]**

문제와 보기가 동일하게 다시 출제되었습니다. 나머지 보기를 통해 로킹의 특징을 정리하고 넘어가세요.

17 E-R 다이어그램에서 개체 타입은 사각형, 관계 타입은 마름모, 속성은 타원으로 표현합니다.

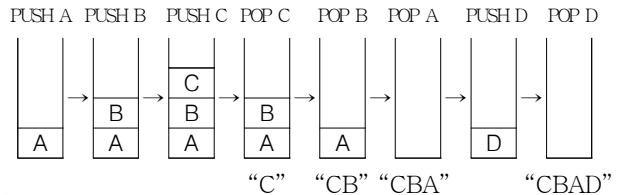
**[전문가의 조언]**

E-R 모델과 관련해서는 E-R 다이어그램에서 사용하는 도형이 가장 많이 출제됩니다. 문제에 제시된 보기를 통해 E-R 모델의 특징을 정리하고, E-R 도형에서 사용하는 기호들을 꼭 암기해 두세요.

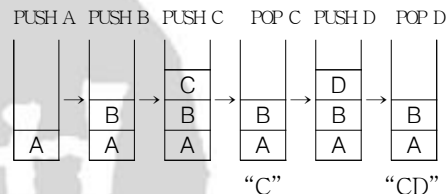
**E-R 도형**

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

18 A, B, C, D 순으로 입력되는 상태에서는 C, D, A, B 순으로 출력할 수 없습니다. 왜 그런지 스택에 직접 자료를 입·출력해보면서 알아보겠습니다. 먼저 ①번을 살펴볼게요. PUSH는 스택에 자료를 입력하는 명령이고, POP은 스택에서 자료를 출력하는 명령입니다.



②번의 경우 “CD” 출력 후에 A를 출력해야 하는데 “B”를 출력하지 않고는 “A”를 출력할 수 없으므로 불가능합니다.



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

19 그래프는 비선형 구조에 해당하고, 나머지는 선형 구조에 해당합니다.

**[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)

20 트리(Tree)의 차수(Degree)는 가장 차수가 많은 노드의 차수입니다. 문제에 주어진 트리(Tree)에서 각 노드의 차수는 A = 2, B = 3, C = 2, E = 2입니다. 곧 B 노드의 차수가 3으로 가장 크므로 트리의 차수는 3입니다.

**[전문가의 조언]**

트리의 차수를 묻는 문제가 자주 출제된다고 했죠. 8번 문제의 해설을 참고하여 노드의 차수가 0인 터미널 노드(단말 노드)에 대해서도 다시 한 번 정리하고 넘어가세요.

21 보조기억장치는 전원 공급이 중단되더라도 기억된 내용이 모두 유지됩니다.

**[전문가의 조언]**

자주 출제되는 문제는 아닙니다. 나머지 보기로 제시된 보조기억장치의 특징만 다시 한 번 읽어보고 넘어가세요.

22 **[전문가의 조언]**

문제와 보기가 동일하게 출제된 적이 있습니다. 버스(Bus)는 Multiplexer를 사용해 구성한다는 것만 기억하고 넘어가세요.

- 23 ALU는 산술 논리 연산장치를 의미하는 것으로 CPU에 있습니다.

**[전문가의 조언]**

종종 출제되는 내용입니다. ALU의 기능을 정리하세요.

**연산장치(ALU, Arithmetic & Logic Unit)**

- 제어장치의 명령에 따라 실제로 연산을 수행하는 장치이다.
- 연산장치가 수행하는 연산에는 산술연산, 논리연산, 관계연산, 이동(Shift) 등이 있다.
- 가산기, 누산기(AC: Accumulator), 보수기, 데이터 레지스터, 오버플로우 검출기, Shift Register 등으로 구성되어 있다.

- 24 **[전문가의 조언]**

인터럽트의 개념보다는 인터럽트의 종류 및 발생 원인을 묻는 문제가 자주 출제됩니다. 인터럽트의 개념을 기억하고 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 명령이나 데이터를 사용할 때 발생하며, 트랩(Trap)이라고도 부른다.
- **명령어 잘못에 의한 인터럽트** : 프로그램에서 명령어를 잘못 사용한 경우
- **프로그램 검사 인터럽트(Program Check Interrupt)** : 0으로 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

- 25     7 5 1  
      + 1 5 4  
      -----

1 1 2 5

**[전문가의 조언]**

8진수와 관련해서는 주로 진법 변환 문제가 출제되는데, 가끔 이 문제처럼 덧셈 문제가 출제되기도 합니다. 8진수 덧셈에서는 더한 결과가 8이 되면 자리올림이 발생한다는 것만 알면 쉽게 풀 수 있습니다. 꼭 이해하고 넘어가세요.

- 26 Op-Code(연산자부)의 크기(비트 수)는 표현할 수 있는 명령의 종류를 나타내는 것으로, OP 코드가 n Bit일 때 최대  $2^n$ 개

의 명령어를 사용할 수 있습니다. 이 문제에서는 OP 코드가 8비트이므로  $2^8=256$ 개의 명령어를 생성할 수 있습니다.

**[전문가의 조언]**

종종 출제되는 내용입니다. 명령어의 구성 형식을 정리하세요.

**명령어의 구성**

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

연산자(Operation Code)부	모드 비트	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).

**Operand부(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.

- 27 입출력 데이터 저장은 채널이 아니라 기억장치의 기능입니다.

**[전문가의 조언]**

채널의 의미, 특징, 종류 모두 중요합니다. 확실히 정리하세요.

**채널(Channel)**

- 채널은 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

- 28 사칙 연산은 수치적인 산술 연산에 속합니다.

**[전문가의 조언]**

자주 출제되는 문제는 아니지만 컴퓨터 학습 시 기본적으로 알아야 할 지식이니 수치적인 산술 연산과 비수치적인 논리 연산의 종류를 파악해 두세요.

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

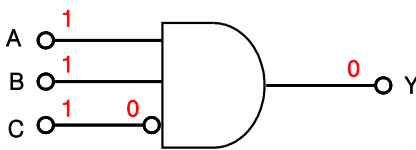
29 트랩이란 잘못된 명령이나 데이터를 사용할 때 발생하는 내부 인터럽트를 의미합니다. 나머지 보기들이 의미하는 인터럽트의 유형이 무엇인지 확인하세요.

- ② 외부 신호 인터럽트  
③, ④ 입·출력 인터럽트

#### [전문가의 조언]

인터럽트의 종류 및 발생 원인을 묻는 문제가 자주 출제된다고 했죠? 이 문제를 틀렸다면, 24번 문제의 [전문가의 조언]을 참조하여 인터럽트 종류 및 발생 원인을 확실히 정리하고 넘어가세요.

30 각각의 입력선에 값을 대입해 보면 쉽게 알 수 있습니다.



#### [전문가의 조언]

기본적인 논리 게이트의 출력 결과만 알고 있으면 맞힐 수 있는 문제가 자주 출제됩니다. 각각의 논리 게이트를 구분할 수 있도록 정리하세요.

#### 논리 게이트

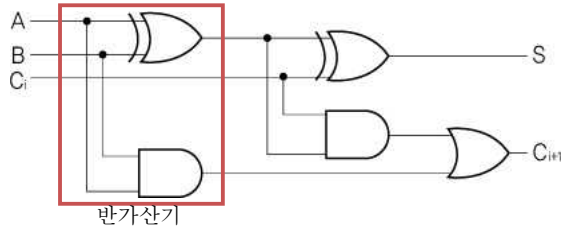
게이트	기호	의미	논리식
AND		입력 신호가 모두 1일 때 1 출력	$Y = A \cdot B$ $Y = AB$
OR		입력 신호 중 1개만 1이어도 1 출력	$Y = A + B$
NOT		입력된 정보를 반대로 변환하여 출력	$Y = A'$ $Y = \bar{A}$
BUFFER		입력된 정보를 그대로 출력	$Y = A$
NAND		NOT + AND, 즉 AND의 부정	$Y = \overline{A \cdot B}$ $Y = \overline{AB}$
NOR		NOT + OR, 즉 OR의 부정	$Y = \overline{A + B}$
XOR		입력되는 것이 모두 같으면 0, 한 개라도 다르면 1 출력	$Y = A \oplus B$ $Y = \bar{A}B + A\bar{B}$ $Y = (A+B)(\bar{A}\bar{B})$ $Y = (A+B)(AB)$
XNOR		NOT + XOR, 즉 XOR의 부정	$Y = A \odot B$ $Y = \bar{A} \oplus \bar{B}$ $Y = AB + \bar{A}\bar{B}$ $Y = (\bar{A}+B)(A+\bar{B})$ $Y = (AB)(\bar{A}\bar{B})$

31 반가산기(Half Adder) 2개를 이용해 만든 전가산기(Full Adder)는 뒷자리에서 올라온 자리올림수( $C_i$ )를 포함하여 1Bit 크기의 2진수 3자리를 더하여 합(S)과 자리올림수( $C_{i+1}$ )를 구하는 회로입니다.

#### [전문가의 조언]

전가산기의 회로 그림과 합, 캐리에 대한 논리식을 알아두세요.

#### 전가산기



- 합 :  $S = (A \oplus B) \oplus C_i$
- 캐리 :  $C_{i+1} = (A \oplus B)C_i + AB$

32 다른 보기가 틀린 이유를 확인하세요.

- ② 동기 고정식은 제어가 간단하지만 CPU의 시간 낭비가 심합니다.  
③ 동기 가변식은 CPU 시간의 낭비를 줄일 수 있지만 제어회로로 조금 복잡합니다.  
④ 한 개의 마이크로 오퍼레이션이 수행되는 과정을 마이크로 사이클이라고 하며, 한 개의 마이크로 오퍼레이션이 실행하는 데 걸리는 시간을 마이크로 사이클 타임이라고 부르니 마이크로 사이클과 마이크로 오퍼레이션은 밀접한 관계가 있습니다.

#### [전문가의 조언]

종종 출제되는 내용입니다. Micro Cycle Time 부여 방식의 종류별 특징을 정리하고 넘어가세요.

#### 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.

33 [전문가의 조언]

종종 출제되는 내용입니다. 0주소 명령어의 특징을 정리해 두세요.

#### 0주소 명령어

- 0주소 명령어는 Operand부 없이 OP-Code부만으로 구성되어 있다.
- 모든 연산은 Stack 메모리의 Stack Pointer가 가리키는 Operand를 이용하여 수행한다.
- 0주소 명령어의 모든 연산은 스택에 있는 자료를 이용하여 수행하기 때문에 스택 머신(Stack Machine)이라고도 한다.
- 스택에 기억된 데이터만을 이용하여 연산하므로 인스트럭



선의 수행시간이 짧다.

- 피연산자를 나타내지 않기 때문에 인스트럭션의 길이가 짧아서 기억공간의 이용이 효율적이다.
- 스택을 사용한 컴퓨터에서 수식을 계산하기 위해서는 우선 수식을 Postfix(역 polish) 형태로 변경하여야 한다.
- 0주소 명령어는 주소의 사용 없이 스택에 연산자와 피연산자를 넣었다 꺼내어 연산한 후 결과를 다시 스택에 넣으면서 연산하기 때문에 원래의 자료가 남지 않는다.

34 Fetch Cycle은 명령어를 주기억장치에서 중앙처리장치의 명령 레지스터로 가져와 해독하는 단계로, 이 단계에서 인스트럭션의 종류에 대한 판단이 이루어집니다.

#### [전문가의 조언]

종종 출제되는 내용입니다. 무슨 단계를 말하는지 구분할 있도록 메이저 스테이트 각 단계의 특징을 정리하세요.

#### 메이저 스테이트

##### • 인출 단계(Fetch Cycle)

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

##### • 간접 단계(Indirect Cycle)

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

##### • 실행 단계(Execute Cycle)

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

##### • 인터럽트 단계(Interrupt Cycle)

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

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

#### [전문가의 조언]

자주 출제되는 내용입니다. 프로그램 카운터를 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
- 시프트 레지스터(Shift Register) : 저장된 값을 왼쪽 또는 오른쪽으로 1Bit씩 자리를 이동시키는 레지스터, 2배 길이 레지스터라고도 함
- 메이저 스테이츠 레지스터(Major Status Register) : CPU의 메이저 상태를 저장하고 있는 레지스터

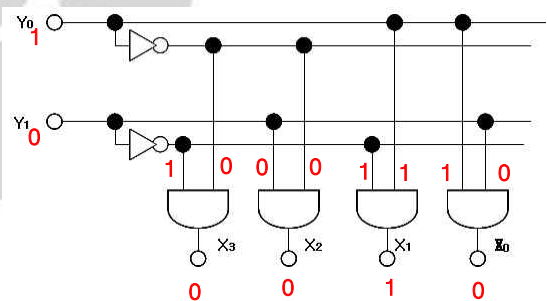
#### 36 [전문가의 조언]

캐시 메모리의 개념과 특징을 묻는 문제가 자주 출제됩니다. 확실히 숙지하세요.

#### 캐시 메모리(Cache Memory)

- 캐시 메모리는 CPU의 속도와 메모리의 속도 차이를 줄이기 위해 사용하는 고속 Buffer Memory이다.
- 캐시는 주기억장치와 CPU 사이에 위치한다.
- 캐시 메모리는 메모리 계층 구조에서 가장 빠른 소자이며, 처리속도가 거의 CPU의 속도와 비슷할 정도이다.
- 캐시를 사용하면 주기억장치를 접근(Access)하는 횟수가 줄어들므로 컴퓨터의 처리속도가 향상된다.
- 캐시 주소표는 검색 시간을 단축시키기 위해 주로 연관기억장치(CAM, Associative Memory)를 사용한다.
- 캐시의 크기는 보통 수십 KByte~수백 KByte이다.

37 이런 문제는 값을 직접 대입해서 풀면됩니다.



#### [전문가의 조언]

종종 출제되는 내용입니다. 꼭 이해하고 넘어가세요.

#### 38 [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)

#### 39 [전문가의 조언]

가끔 출제되는 문제입니다. 기억장치를 접근 속도 순으로 나열하면 다음과 같습니다. 기억해 두세요.

#### 기억장치별 접근 속도(빠름 → 느림)

레지스터(Register) → 캐시(Cache) → RAM(Main Memory) → ROM → 자기 코어(Magnetic Core) → 자기

디스크(Magnetic Disk) → 자기 테이프(Magnetic Tape)

#### 40 [전문가의 조언]

DMA에서는 특징, 동작 원리, 장점 등에 대한 다양한 문제가 출제됩니다. 이 문제에서는 DMA의 특징을 정리하고 넘어가세요.

##### DMA(Direct Memory Access)

- DMA는 입·출력장치가 직접 주기억장치를 접근(Access)하여 Data Block을 입·출력하는 방식으로, 입·출력 전송이 CPU의 레지스터를 경유하지 않고 수행된다.
- CPU는 I/O에 필요한 정보를 DMA 제어기에 알려서 I/O 동작을 개시시킨 후 I/O 동작에 더 이상 간섭하지 않고 다른 프로그램을 할당하여 수행한다.
- DMA 방식은 입·출력 자료 전송 시 CPU를 거치지 않기 때문에 CPU의 부담이 없어 보다 빠른 데이터의 전송이 가능하다.
- DMA는 인터럽트 신호를 발생시켜 CPU에게 입·출력 종료 를 알린다.
- DMA는 사이클 스틸(Cycle Steal) 방식을 이용하여 데이터를 전송한다.  
※ 사이클 스틸(Cycle Steal) : 데이터 채널(DMA 제어기)과 CPU가 주기억장치를 동시에 Access할 때 우선순위를 데이터 채널에게 주는 방식
- CPU와 DMA 제어기는 메모리와 버스를 공유한다.

#### 41 [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 있는 지를 검사하는 방법

#### 42 [전문가의 조언]

코드 설계와 관련해서는 특정 단계에서 수행하는 작업에 대한 구체적인 문제보다는 코드 설계 순서를 나열하는 문제가 자주 출제됩니다. 이 문제에서는 사용 범위 결정 과정에서 수행하는 작업이 무엇인지만 기억하고 다음에 제시된 코드 설계 순서는 기억하세요.

##### 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]

파일을 용도를 구분하는 문제가 자주 출제됩니다. 무슨 파일을 말하는지 구분할 수 있도록 각각의 특징을 정리해 두세요.

##### 데이터 파일의 종류

- **원시 파일(Source File)** : 입력 데이터를 알맞은 매체에

변환하여 만든 파일

- **트랜잭션 파일(Transaction File)** : 거래 내역이나 변동 내용 등 일시적인 성격을 지닌 정보를 기록하는 파일로, 마스터 파일을 갱신하거나 조회할 때 사용됨
- **마스터 파일(Master File)** : 전표 처리에서의 원장 또는 대장에 해당하는 파일로, 자료 관리의 중추적 역할을 담당하며 기본이 되는 파일임. 트랜잭션 파일에 의해 갱신됨
- **요약 파일(Summary File)** : 다른 파일의 중요 내용이나 합계를 요약해 놓은 파일로, 집계용으로 많이 사용됨
- **히스토리 파일(History File)** : 후일 통계 처리에 사용할 자료나 사고 발생시 마스터 파일 등을 원상 복구시키기 위한 자료를 보존한 파일
- **백업 파일(Backup File)** : 만일의 사고에 대비하여 마스터 파일을 백업해 놓은 파일
- **트레일러 파일(Trailer File)** : 마스터 파일을 목적에 따라 여러 개의 파일로 나누었을 때 가장 끝부분에 해당하는 파일

#### 44 [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
	대체 항목의 나열 : 또는(or)
**	자료의 설명 : 주석(Comment)

#### 45 [전문가의 조언]

주로 시스템 특성의 종류를 구분하는 문제가 출제되었는데, 시스템 특성에 대한 각각의 의미까지 파악해야 하는 문제가 출제 되었네요. 시스템 특성의 종류는 물론 의미까지 꼭 파악해 두세요.

##### 시스템의 특성

목적성	서로 다른 기능을 가지고 있는 시스템의 각 구성 요소들은 어떤 하나의 공통된 목적을 위해 존재함
자동성	어떤 조건이나 상황의 변화에 대응하여 자동으로 적절한 처리를 수행함
제어성	시스템이 오류 없이 어떤 기능을 수행하도록 제어함
종합성	항상 다른 관련 시스템과 상호 의존 관계를 유지함

#### 46 [전문가의 조언]

폭포수 모형과 프로토타이핑 모형을 구분해서 알아둬야 합니다. 문제를 통해 폭포수 모형의 특징을, 다음 내용을 통해 프로토타이핑 모형의 특징을 정리해 두세요.

##### 프로토타이핑 모형(Prototyping Model, 원형 모형)

- 사용자의 요구 사항을 정확히 파악하기 위해 실제 개발될 소프트웨어에 대한 건본(시제)품(Prototype)을 만들어 최 종 결과물을 예측하는 모형이다.
- 시스템의 일부 혹은 시스템의 모형을 만드는 과정으로서

요구된 소프트웨어를 구현하는데, 이는 추후 구현 단계에서 사용될 골격 코드가 된다.

- 소프트웨어의 개발이 완료된 시점에서 오류가 발견되는 폭포수 모형의 단점을 보완하기 위한 모형이다.
- 프로토타입은 요구 분석 단계에서 사용하게 되며, 프로토타입의 평가가 끝나고 개발이 승인되면 다른 모형을 이용하여 본격적인 개발이 이루어진다.

#### 47 [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) : 동일한 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

- 48 지문은 9 8 7 6 5 의 5 대신 4를 기록한 것, 즉 임의의 한 자리를 잘못 기록해서 발생한 오류이므로 필사 오류(Transcription Error)에 해당합니다.

##### [전문가의 조언]

오류의 형태를 구분하는 문제는 종종 출제됩니다. 특히 필사 오류(Transcription Error)와 전위 오류(Transposition Error)가 자주 출제되므로, 두 오류를 중심으로 각 내용을 정리해 두세요.

##### 코드의 오류

- **필사 오류(Transcription Error) = 오자 오류** : 입력시 임의의 한 자리를 잘못 기록한 경우 발생
- **전위 오류(Transposition Error)** : 입력시 좌우 자리를 바꾸어 기록한 경우 발생
- **이중 오류(Double Transposition Error)** : 전위 오류가 2개 이상 발생한 경우
- **생략 오류(Omission Error)** : 입력시 한 자리를 빼놓고 기록한 경우 발생
- **추가 오류(Addition Error)** : 입력시 한 자리를 더 추가하

여 기록한 경우 발생

- **임의 오류(Random Error)** : 오류가 두 가지 이상 결합하여 발생한 경우

#### 49 [전문가의 조언]

종종 출제되는 내용입니다. 어떤 방식을 의미하는지 구분할 수 있을 정도로만 알고 있으면 됩니다.

##### 데이터 입력 방식

터 언 어라운드 시스템	<ul style="list-style-type: none"> <li>• 입력된 자료가 처리되어 일단 출력된 후 이용자를 경유하여 다시 재입력되는 방식</li> <li>• 공과금, 보험료 징수 등의 지로용지를 처리하는 데 사용됨</li> </ul>
집중 매체화 시스템	발생한 데이터를 전표상에 기록하고, 일정 시간 단위로 일괄 수집하여 입력 매체에 수록하는 방식
분산 매체화 시스템	데이터를 발생한 장소에서 매체화하여 처리하는 방식
직접 입력 시스템	OMR 카드 등과 같이 사람이 직접 손으로 작성하여 입력하는 방식
타건 입력 시스템	키보드나 라이트 펜을 이용하여 입력하는 방식
음성 입력 시스템	음성 인식 장치의 패턴 인식 기능을 이용하여 소리로 입력하는 방식

#### 50 [전문가의 조언]

자료 흐름도에서 사용하는 구성 요소와 이를 나타내는 기호를 묻는 문제는 자주 출제됩니다. 자료 흐름도의 4가지 구성 요소와 구성 요소를 나타내는 기호를 꼭 알아두세요.

##### 자료 흐름도의 구성 요소

구성 요소	의미
처리 (Process)	<ul style="list-style-type: none"> <li>• 입력된 자료를 출력으로 변환하는 것으로 프로세스, 기능, 버블이라고도 함</li> <li>• 원 안에 처리 명칭을 기술함</li> </ul>
자료 흐름 (Data Flow)	<ul style="list-style-type: none"> <li>• 발생지, 종착지, 처리 및 저장소 사이에서 자료의 흐름을 나타냄</li> <li>• 화살표 위에 자료의 명칭을 기술함</li> </ul>
자료 저장소 (Data Store)	<ul style="list-style-type: none"> <li>• 시스템상의 자료 저장소를 나타냄</li> <li>• 평행선 안에 자료 저장소 명칭을 기술함</li> </ul>
단말 (Terminator)	<ul style="list-style-type: none"> <li>• 시스템에 필요한 자료가 입력되는 발생지(Source)와 시스템에서 처리된 자료가 출력되는 종착지(Sink)를 나타냄</li> <li>• 대상 시스템의 외부에 존재하는 사람이나 조직체를 나타냄</li> <li>• 사각형 안에 발생지/종착지 명칭을 기술함</li> </ul>

- 51 MTTR(Mean Time To Repair, 평균수리시간)은 시스템에 고장이 발생하여 가동하지 못한 시간들의 평균입니다. 상태에서 R은 가동중, S는 고장중이므로 계산식은 다음과 같습니다.

$$\begin{aligned}
 \text{MTTR} &= \frac{\text{고장중1} + \text{고장중2} + \text{고장중3} + \dots + \text{고장중n}}{N} \\
 &= \frac{100 + 60 + 80}{3} \\
 &= 80
 \end{aligned}$$

##### [전문가의 조언]

자주 출제되는 내용은 아닙니다. MTTR의 개념과 계산식만



알아두세요.

## 52 [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 정리하세요.

### 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
- **갱신(Update)** : 마스터 파일의 내용을 변동 파일에 의해 추가, 삭제, 수정 등의 작업을 하여 새로운 내용의 마스터 파일을 생성하는 것
- **분배(Distribution)** : 하나의 파일 안에서 조건에 맞는 것과 그렇지 않은 것을 분리하는 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
- **생성(Generate)** : 파일을 읽어 들어서 데이터를 변형하여 입력 파일과 다른 형식의 새로운 파일을 작성하는 기능

## 53 문서화는 시스템의 개발 요령과 순서 등 시스템 개발에 관련된 모든 행위를 문서로 만들어 두는 것으로, 업무에 실질적인 도움을 주고자 하는 것이지 요식적 행위로 하는 것은 아닙니다.

### [전문가의 조언]

시스템 문서화의 효과(목적)를 묻는 문제는 자주 출제되고 있습니다. 어려운 내용이 아니니 차분히 읽어보세요.

### 문서화의 목적 및 효과

- 시스템 개발팀에서 운용팀으로 인수인계가 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.
- 다른 업무 개발에 참고할 수 있다.

## 54 결합도(Coupling)는 두 모듈 간의 상호 의존도를 측정하는 것으로, 결합도와 관련하여 설계 품질이 좋다는 것은 결합도가 낮다는 것을 의미합니다. 즉 보기 중 설계 품질이 가장 좋은 결합도는 결합도가 가장 낮은 자료(Date) 결합도입니다.

### [전문가의 조언]

결합도와 관련해서는 각각의 결합도가 어떤 결합을 하고 있는지와 결합 정도에 따른 순서를 묻는 문제가 자주 출제됩니다. 확실히 정리해 두세요.

### 결합도의 종류

- **자료 결합도(Data Coupling)** : 모듈 간의 인터페이스가 자

료 요소로만 구성될 때의 결합도

- **스탬프(검인) 결합도(Stamp Coupling)** : 모듈 간의 인터페이스로 배열이나 레코드 등의 자료 구조가 전달될 때의 결합도
- **제어 결합도(Control Coupling)** : 한 모듈에서 다른 모듈로 논리적인 흐름을 제어하는 데 사용하는 제어 요소(Function code, Switch, Tag, Flag)가 전달될 때의 결합도
- **외부 결합도(External Coupling)** : 어떤 모듈에서 외부로 선언한 데이터(변수)를 다른 모듈에서 참조할 때의 결합도
- **공통(공유) 결합도(Common Coupling)** : 공유되는 공통 데이터 영역을 여러 모듈이 사용할 때의 결합도
- **내용 결합도(Content Coupling)** : 한 모듈이 다른 모듈의 내부 기능 및 그 내부 자료를 직접 참조하거나 수정할 때의 결합도

### 결합도의 순서(약함 → 강함)

자료 결합도 → 스탬프 결합도 → 제어 결합도 → 외부 결합도 → 공통 결합도 → 내용 결합도

## 55 [전문가의 조언]

동일한 문제가 출제된 적이 있는 문제입니다. 이 문제에서는 N-S 차트가 무엇인지 정도만 알아두세요.

## 56 [전문가의 조언]

중중 출제되는 내용입니다. 객체와 관련된 용어들의 의미를 정리하고 넘어가세요.

- **속성(Attribute)** : 한 클래스 내에 속한 객체들이 가지고 있는 데이터 값들을 단위별로 정의하는 것으로서, 성질, 분류, 식별, 수량 또는 현재 상태 등을 표현함
- **메서드(Method)** : 객체에 정의된 연산을 의미하며, 객체의 상태를 참조하거나 변경하는 수단이 됨
- **상속성(Inheritance)** : 이미 정의된 상위 클래스의 메서드를 비롯한 모든 속성을 하위 클래스가 물려받을 수 있는 것
- **메시지(Message)** : 외부로부터 하나의 객체에 전달되는 메소드의 요구
- **클래스(Class)** : 2개 이상의 유사한 객체들을 묶어서 하나의 공통된 특성을 표현하는 요소

## 57 [전문가의 조언]

코드의 종류에 대한 문제가 자주 출제됩니다. 무슨 코드를 말하는지 구분할 수 있도록 각각의 특징을 정리해 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
- **표의 숫자 코드(Significant Digit Code)** : 코드화 대상 항목의 성질, 즉 길이, 넓이, 부피, 지름, 높이 등의 물리적 수치를 그대로 코드에 적용시키는 방법
- **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

### 58 [전문가의 조언]

럼바우 객체 분석 기법의 3가지 모델링의 종류와 제시된 내용이 어떤 모델링인지에 대한 설명인지를 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 연관 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

### 59 [전문가의 조언]

색인 순차 파일의 개념과 구성 요소를 묻는 문제가 출제되고 있습니다. 잘 정리하고 넘어가세요.

**색인 순차 파일(Index Sequential File)**

- 순차 처리와 랜덤 처리가 모두 가능하도록 레코드들을 키값순으로 정렬(Sort)시켜 기록하고, 레코드의 키 항목만을 모은 색인을 구성하여 편성하는 방식이다.
- 색인을 이용한 순차적인 접근 방법을 제공하여 ISAM(Index Sequential Access Method)이라고 한다.
- 레코드를 참조할 때 색인을 탐색한 후 색인이 가리키는 포인터(주소)를 사용하여 직접 참조할 수 있다.
- 일반적으로 자기 디스크에 많이 사용되며, 자기 테이프에서는 사용할 수 없다.

**색인 순차 파일의 구조**

- **기본 구역(Prime Area)** : 실제 레코드들을 기록하는 부분으로, 각 레코드는 키값순으로 저장된다.
- **색인 구역(Index Area)** : 기본 구역에 있는 레코드들의 위치를 찾아가는 색인이 기록되는 부분으로 트랙 색인 구역, 실린더 색인 구역, 마스터 색인 구역으로 구분할 수 있다.
  - **트랙 색인 구역(Track Index Area)**
    - ▶ 기본 구역의 한 트랙상에 기록되어 있는 데이터 레코드 중 최대 키값과 주소가 기록되는 색인으로, 한 실린더당 하나씩 만들어진다.
    - ▶ 처리할 레코드가 실제로 어느 트랙에 기록되어 있는지를 판별할 수 있게 한다.
  - **실린더 색인 구역(Cylinder Index Area)** : 각 트랙 색인의 최대 키값과 해당 레코드가 기록된 실린더의 정보가 기록되는 색인으로, 한 파일당 하나씩 만들어짐
  - **마스터 색인 구역(Master Index Area)** : 실린더 색인 구역의 정보가 많을 경우 그것을 일정한 크기의 블록으로 구성하는 데, 이때 해당 레코드가 어느 실린더 색인

구역에 기록되어 있는지를 기록하는 색인임

- **오버플로 구역(Overflow Area)** : 기본 구역에 빈 공간이 없어서 새로운 레코드의 삽입이 불가능할 때를 대비하여 예비적으로 확보해 둔 부분이다.
  - **실린더 오버플로 구역(Cylinder Overflow Area)** : 각 실린더마다 만들어지는 오버플로 구역으로, 해당 실린더의 기본 구역에서 오버플로된 데이터를 기록함
  - **독립 오버플로 구역(Independent Overflow Area)** : 실린더 오버플로 구역에 더 이상 오버플로된 데이터를 기록할 수 없을 때 사용할 수 있는 예비 공간으로, 실린더 오버플로 구역과는 별도로 만들어짐

### 60 [전문가의 조언]

표준 처리 패턴은 자주 출제된다고 했죠? 52번 문제의 [전문가의 조언]을 통해 다시 한 번 공부합시다.

### 61 [전문가의 조언]

단순히 클라이언트(Client)의 개념을 묻는 문제가 처음 출제되었네요. 네트워크를 이용하여 서비스를 제공하는 측은 서버(Server), 서비스를 요구하는 측은 클라이언트(Client)라는 것만 기억하고 넘어가세요.

### 62 [전문가의 조언]

SRT를 풀어쓰면 Shortest Remaining Time입니다. 즉 남은 실행시간이 가장 짧은 프로세스에게 CPU를 할당한다는 것만 알아두면 쉽게 맞힐 수 있는 문제입니다. 나머지 보기로 제시된 스케줄링의 개념도 정리하고 넘어가세요.

**[전문가의 조언]**

- **기한부(Deadline)** : 프로세스에게 일정한 시간을 주어 그 시간 안에 프로세스를 완료하도록 하는 기법
- **HRN(Highest Response-ratio Next)** : 실행 시간이 긴 프로세스에 불리한 SJF 기법을 보완하기 위한 것으로, 대기 시간과 서비스(실행) 시간을 이용하는 기법
- **다단계 큐(MQ; Multi-level Queue)** : 프로세스를 특정 그룹으로 분류할 수 있을 경우 그룹에 따라 각기 다른 준비상태 큐를 사용하는 기법

### 63 [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 기법으로, 벨레이디의 모순(Belady's Anomaly) 현상이 발생함
- **LRU(Least Recently Used)** : 최근에 가장 오랫동안 사용하지 않은 페이지를 교체하는 기법
- **LFU(Least Frequently Used)** : 사용 빈도가 가장 적은 페이지를 교체하는 기법
- **SCR(Second Chance Replacement)** : 가장 오랫동안 주기억장치에 있던 페이지 중 자주 사용되는 페이지의 교체를 방지하기 위한 것으로, FIFO 기법의 단점을 보완하는 기법

### 64 [전문가의 조언]

종종 출제되는 내용입니다. 파일 보호 기법의 종류는 물론 각각의 보호 방법도 알고 있어야 합니다.

## 파일 보호 기법

<b>파일의 명명 (Naming)</b>	접근하고자 하는 파일 이름을 모르는 사용자를 접근 대상에서 제외시키는 기법
<b>비밀번호 (Password, 암호)</b>	각 파일에 단독 암호와 기록 암호를 부여하여 암호를 아는 사용자에게만 접근을 허용하는 기법
<b>접근 제어 (Access Control)</b>	사용자에 따라 공유 데이터에 접근할 수 있는 권한을 제한하는 방법, 즉 각 파일마다 접근 목록을 두어 접근 가능한 사용자와 동작을 기록한 후 이를 근거로 접근을 허용하는 기법

- 65 3개의 페이지 프레임에 갖는 주기억장치이므로 아래 그림과 같이 표현할 수 있습니다.

참조 페이지	1	2	1	0	4	1	3
페이지 프레임	1	1	1	1	4	4	4
		2	2	2	2	1	1
				0	0	0	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 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 6회이고, 최종적으로 남아 있는 페이지는 4, 1, 3입니다.

## [전문가의 조언]

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

## 66 [전문가의 조언]

문제와 보기가 동일하게 출제된 문제로 Aging이 ‘나이를 먹는다’라는 의미란 걸 알면 쉽게 맞출 수 있는 문제입니다. 이 문제에서는 Aging의 의미만 기억하고 넘어가세요.

- 67 RR(Round-Robin) 방식은 프로세스에게 이미 할당된 프로세서를 강제로 빼앗을 수 있는 선점형 스케줄링 기법입니다.

## [전문가의 조언]

라운드 로빈 스케줄링에 대한 문제는 자주 출제됩니다. 라운드 로빈 스케줄링의 특징을 숙지하고 넘어가세요.

## 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.

- 68 기억장치의 분할 방식에는 단일 분할 기법과 다중 분할 기법이 있으며, 단일 분할 기법에는 오버레이 기법과 스와핑 기법이, 다중 분할 기법에는 고정(정적) 분할 기법과 동적(가변) 분할 기법이 있습니다.

## [전문가의 조언]

문제와 보기가 동일하게 다시 출제되었습니다. 기억장치의 분할 기법에는 어떠한 것들이 있는지 기억해 두세요.

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

## [전문가의 조언]

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

## 교착상태 발생의 필요 충분 조건

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

## 70 [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

- 71 매크로 정의 인식, 정의 저장, 호출 인식 등을 처리하는 것은 매크로 프로세서입니다.

## [전문가의 조언]

자주 출제되는 내용입니다. 운영체제의 기능을 꼭 숙지하세요.

#### 운영체제의 기능

- 프로세서(처리기, Processor), 기억장치(주기억장치, 보조기억장치), 입·출력장치, 파일 및 정보 등의 자원을 관리한다.
- 자원을 효율적으로 관리하기 위해 자원의 스케줄링 기능을 제공한다.
- 사용자와 시스템 간의 편리한 인터페이스를 제공한다.
- 시스템의 각종 하드웨어와 네트워크를 관리·제어한다.
- 데이터를 관리하고, 데이터 및 자원의 공유 기능을 제공한다.
- 시스템의 오류를 검사하고 복구한다.
- 자원 보호 기능을 제공한다.
- 입·출력에 대한 보조 기능을 제공한다.
- 가상 계산기 기능을 제공한다.

#### 72 [전문가의 조언]

프로세스 스케줄링 기법과 관련해서는 선점 스케줄링과 비선점 스케줄링을 비교하는 문제가 종종 출제됩니다. 각각의 종류와 특징을 정리해 두세요.

##### 비선점(Non-preemptive) 스케줄링

- 이미 할당된 CPU를 다른 프로세스가 강제로 빼앗아 사용할 수 없는 스케줄링 기법이다.
- 프로세스가 CPU를 할당받으면 해당 프로세스가 완료될 때까지 CPU를 사용한다.
- 모든 프로세스에 대한 요구를 공정하게 처리할 수 있다.
- 프로세스 응답 시간을 예측하기 쉽고, 일괄 처리 방식에 적합하다.
- 중요한 작업(짧은 작업)이 중요하지 않은 작업(긴 작업)을 기다리는 경우가 발생할 수 있다.

- 종류 : FCFS(FIFO), SJF, 우선순위, HRN, 기한부 등

##### 선점(Preemptive) 스케줄링

- 하나의 프로세스가 CPU를 할당받아 실행하고 있을 때 우선순위가 높은 다른 프로세스가 CPU를 강제로 빼앗아 사용할 수 있는 스케줄링 기법이다.
- 우선순위가 높은 프로세스를 빠르게 처리할 수 있다.
- 주로 빠른 응답 시간을 요구하는 대화식 시분할 시스템에 사용된다.
- 많은 오버헤드(Overhead)를 초래한다.
- 선점이 가능하도록 일정 시간 배당에 대한 인터럽트용 타이머 클럭(Clock)이 필요하다.
- 종류 : Round Robin, SRT, 선점 우선순위, 다단계 큐, 다단계 피드백 큐 등

- 73 파일 시스템은 파일의 저장, 액세스, 공유, 보호 등 보조기억장치에서의 파일을 총괄하는 파일 관리 기술입니다. 사용자는 파일 시스템을 사용하므로 특정장치의 물리적인 제어 방식을 알 필요가 없습니다.

#### [전문가의 조언]

파일 시스템의 전반적인 기능을 묻는 문제가 종종 출제됩니다. 파일 시스템의 기능 및 특징을 한번 읽고 넘어가세요.

##### 파일 시스템의 기능 및 특징

- 사용자와 보조기억장치 사이에서 인터페이스를 제공한다.
- 적절한 제어 방식을 통해 타인의 파일을 공동으로 사용할 수 있도록 한다.
- 파일 공유를 위해서 관독만 허용, 기록만 허용, 수행만 허용 또는 이들을 여러 형태로 조합한 것 등 여러 종류의 액세스 제어 방법을 제공한다.
- 사용자가 적합한 구조로 파일을 구성할 수 있도록 한다.
- 불의의 사태를 대비하여 파일의 예비(Backup)와 복구

(Recovery) 등의 기능을 제공한다.

- 사용자가 파일을 편리하게 사용할 수 있도록 파일의 논리적 상태(디렉터리)를 보여 주어야 한다.
- 정보를 암호화하고 복호화할 수 있는 기능을 제공한다.

- 74 프로세스 제어를 위해 사용되는 명령어에는 끝내기(End), 중지(Abort), 적재(Load), 실행(Execute)이 있습니다.

#### [전문가의 조언]

처음 출제된 내용입니다. 이 문제에서는 프로세스 제어를 위해 사용되는 명령어 4가지만 정확히 기억하고 넘어가세요.

- 75 성형 구조는 모든 사이트가 하나의 중앙 사이트에 직접 연결되어 있는 구조이므로 중앙 사이트 고장 시 전체 사이트가 단절됩니다.

#### [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

- 76 카운팅(Counting), 집계(Totaling)에 사용되는 변수는 시간 구역성(Temporal Locality)이 이루어지는 기억 장소입니다.

#### [전문가의 조언]

문제와 보기가 동일하게 다시 출제되었습니다. 구역성에 대한 문제는 시간 구역성과 공간 구역성의 특징을 구분할 수 있어야 답을 찾을 수 있는 문제가 주로 출제되고 있으니 정리해 두세요.

##### 시간 구역성(Temporal Locality)

- 시간 구역성은 프로세스가 실행되면서 하나의 페이지를 일정 시간 동안 집중적으로 액세스하는 현상이다.
- 한 번 참조한 페이지는 가까운 시간 내에 계속 참조할 가능성이 높음을 의미한다.
- 시간 구역성이 이루어지는 기억 장소 : Loop(반복, 순환), 스택(Stack), 부프로그램(Sub Routine), Counting(1씩 증감), 집계(Totaling)에 사용되는 변수(기억 장소)

##### 공간 구역성(Spatial Locality)

- 공간 구역성은 프로세스 실행 시 일정 위치의 페이지를 집중적으로 액세스하는 현상이다.
- 어느 하나의 페이지를 참조하면 그 근처의 페이지를 계속 참조할 가능성이 높음을 의미한다.
- 공간 구역성이 이루어지는 기억 장소 : 배열 순회(Array Traversal, 배열 순례), 순차적 코드의 실행, 프로그래머들이 관련된 변수(데이터를 저장할 기억 장소)들을 서로 근처에 선언하여 할당되는 기억 장소, 같은 영역에 있는 변수를 참조할 때 사용



## 77 [전문가의 조언]

교착상태 해결 방법들의 개별적인 의미를 파악하고 있어야 합니다. 나머지 보기로 제시된 기법의 의미를 파악해 두세요.

- **회복 기법(Recovery)** : 교착상태를 일으킨 프로세스를 종료하거나 교착상태의 프로세스에 할당된 자원을 선점하여 프로세스나 자원을 회복하는 것
- **발견 기법(Detection)** : 시스템에 교착상태가 발생했는지 점검하여 교착상태에 있는 프로세스와 자원을 발견하는 것
- **회피 기법(Avoidance)** : 교착상태가 발생할 가능성을 배제하지 않고 교착상태가 발생하면 적절히 피해나가는 방법으로, 주로 은행원 알고리즘(Banker's Algorithm)이 사용됨

## 78 [전문가의 조언]

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

주/중 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>

## 79 HRN 기법의 우선 순위 계산식은 '(대기 시간+서비스 시간)/서비스 시간'입니다. 여기에 각 작업을 대입하여 계산하면

- ㉠ 작업 :  $(20+5)/5 = 5$
- ㉡ 작업 :  $(5+5)/5 = 2$
- ㉢ 작업 :  $(3+12)/12 = 1.25$
- ㉣ 작업 :  $(5+3)/3 = 2.66$

계산된 숫자가 클수록 우선 순위가 높습니다.

[전문가의 조언]

HRN 스케줄링의 우선 순위 계산식이나 계산 결과를 묻는 문제가 자주 출제되므로 계산식을 꼭 기억하고 있어야 합니다.

## 80 [전문가의 조언]

스래싱에 대한 문제는 개념을 묻는 문제가 대부분입니다. 스래싱의 개념과 함께 페이징과 스와핑의 개념도 알아두세요.

- **페이징(Paging) 기법** : 가상기억장치에 보관되어 있는 프로그램과 주기억장치의 영역을 동일한 크기로 나눈 후 나뉜 프로그램(페이지)을 동일하게 나뉜 주기억장치의 영역(페이지 프레임)에 적재시켜 실행하는 기법
- **스와핑(Swapping) 기법** : 하나의 프로그램 전체를 주기억장치에 할당해서 사용하다 필요에 따라 다른 프로그램과 교체하는 기법

## 81 [전문가의 조언]

자주 출제되는 문제는 아닙니다. IP 주소 클래스를 간단히 정리하고 넘어가세요.

IP 주소 클래스

A Class	<ul style="list-style-type: none"> <li>• 국가나 대형 통신망에 사용</li> <li>• 주소 범위 : 0.0.0.0~127.255.255.255</li> <li>• 16,777,216개 호스트 사용</li> </ul>
B Class	<ul style="list-style-type: none"> <li>• 중대형 통신망에 사용</li> <li>• 주소 범위 : 128.0.0.0~191.255.255.255</li> <li>• 65,536개 호스트 사용</li> </ul>

C Class	<ul style="list-style-type: none"> <li>• 소규모 통신망에 사용</li> <li>• 주소 범위 : 192.0.0.0~223.255.255.255</li> <li>• 256개 호스트 사용</li> </ul>
D Class	<ul style="list-style-type: none"> <li>• 멀티캐스트용으로 사용</li> <li>• 주소 범위 : 224.0.0.0~239.255.255.255</li> </ul>
E Class	<ul style="list-style-type: none"> <li>• 실험용으로 사용</li> </ul>

- 82 • 데이터 신호 속도(Bps)는 '변조 속도(Baud) × 변조 시 상태 변화 비트 수'이고 4상 위상이란 한 신호에 2비트( $2^2=4$ )를 전송하므로 상태 변화 비트 수는 2가 됩니다.  
• 신호 속도(bps) =  $800 \times 2 = 1600$

[전문가의 조언]

자주 출제되는 문제입니다. 신호 속도(bps) 계산 공식과 더불어 변조 속도(baud) 계산 공식도 확실히 암기하세요.

변조 속도 계산 공식

변조 속도(baud) = 데이터 전송 속도(bps) / 변조 시 상태 변화 수

## 83 [전문가의 조언]

문제와 보기가 동일하게 출제되었던 문제입니다. 이 문제에서는 Li-Fi의 개념만 기억하고 넘어가세요.

- 84 ATM 셀은 헤더 5Byte, 페이로드(사용자 정보) 48Byte로 구성됩니다.

[전문가의 조언]

이 문제에서는 ATM 셀의 헤더와 페이로드의 크기만 확실히 기억해 두세요.

## 85 [전문가의 조언]

IPv6의 특징을 묻는 문제가 최근들어 종종 출제되고 있습니다. 세부적인 특징까지 잘 정리해 두세요.

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.
- IPv6 주소의 구성

유니캐스트 (Unicast)	단일 송신자와 단일 수신자 간의 통신 (1 대 1 통신에 사용)
멀티캐스트 (Multicast)	단일 송신자와 다중 수신자 간의 통신 (1 대 다 통신에 사용)
애니캐스트 (Anycast)	단일 송신자와 가장 가까이 있는 단일 수신자 간의 통신(1 대 1 통신에 사용)

• IPv6 주소의 표현 방법

- 01DA : FF01 : 0000 : 0000 : 0000 : 00FF : FA21 : 3C5A → 16진수로 표현하며 콜론으로 구분
- 1DA : FF01 : 0 : 0 : 0 : FF : FA21 : 3C5A → 앞에 오는 0은 생략할 수 있음
- 1DA : FF01 :: FF : FA21 : 3C5A → 0이 연속되는 경우 연속된 0은 생략하고 '::'으로 기입할 수 있음



86 ②, ③, ④번은 데이터그램 패킷 교환 방식에 대한 설명입니다.

**[전문가의 조언]**

패킷 교환 방식의 종류에는 가상 회선 방식과 데이터그램 방식이 있으며, 이와 관련된 문제는 항상 두 방식을 서로 구분해서 알고 있어야 풀 수 있는 문제가 출제됩니다. 패킷 교환 방식의 종류별 특징을 정리하고 넘어가세요.

**패킷 교환 방식의 종류**

**가상 회선 방식**

- 단말장치 상호 간에 논리적인 가상 통신 회선을 미리 설정하여 송신지와 수신지 사이의 연결을 확립한 후에 설정된 경로를 따라 패킷들을 순서적으로 운반하는 방식이다.
- 정보 전송 전에 제어 패킷에 의해 경로가 설정된다.
- 통신이 이루어지는 컴퓨터 사이의 데이터 전송의 안정, 신뢰성이 보장된다.
- 모든 패킷은 같은 경로로, 발생 순서대로 전송된다. 즉 패킷의 송·수신 순서가 같다.

**데이터그램 방식**

- 연결 경로를 설정하지 않고 인접한 노드들의 트래픽(전송량) 상황을 감안하여 각각의 패킷들을 순서에 상관없이 독립적으로 운반하는 방식이다.
- 패킷마다 전송 경로가 다르므로, 패킷은 목적지의 완전한 주소를 가져야 한다.
- 네트워크의 상황에 따라 적절한 경로로 패킷을 전송하기 때문에 융통성이 좋다.
- 순서에 상관없이 여러 경로를 통해 도착한 패킷들은 수신측에서 순서를 재정리한다.
- 소수의 패킷으로 구성된 짧은 데이터 전송에 적합하다.

87 광섬유 케이블의 원료인 유리는 절연성이 좋아 전자 유도의 영향을 받지 않으므로 전자기적 잡음에 강합니다.

**[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.
- 가볍고 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.

88 [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

89 소요 대역폭(KHz) =  $2 \times (\text{변조지수} + 1) \times \text{최고 주파수}$   
 $= 2 \times (10 + 1) \times 4\text{kHz}$   
 $= 88\text{kHz}$

**[전문가의 조언]**

정보 통신 기사 시험에 주로 출제되는 문제입니다. 풀이 과정만 다시 한 번 확인하고 넘어가세요.

90 [전문가의 조언]

스트립 소켓의 용도를 묻는 문제가 처음 출제되었네요. TCP와 관련된 문제는 주로 TCP와 IP 프로토콜을 구분하는 문제가 출제됩니다. 스트립 소켓의 용도를 기억하고 TCP와 IP를 서로 구분할 수 있도록 각각의 특징을 정리하고 넘어가세요.

**TCP(Transmission Control Protocol)**

- OSI 7계층의 전송 계층에 해당한다.
- 신뢰성 있는 연결형 서비스를 제공한다.
- 패킷의 다중화, 순서 제어, 오류 제어, 흐름 제어 기능을 제공한다.

**IP(Internet Protocol)**

- OSI 7계층의 네트워크 계층에 해당한다.
- 데이터그램을 기반으로 하는 비연결형 서비스를 제공한다.
- 패킷의 분해/조립, 주소 지정, 경로 선택 기능을 제공한다.

91 [전문가의 조언]

프로토콜의 개념이나 기본 3요소를 묻는 문제가 종종 출제됩니다. 문제를 통해 프로토콜의 개념을 기억하고 프로토콜의 3요소를 정리하고 넘어가세요.

**통신 프로토콜의 기본 요소**

구문(Syntax)	전송하고자 하는 데이터의 형식, 부호화, 신호 레벨 등을 규정함
의미(Semantics)	두 기기 간의 효율적이고 정확한 정보 전송을 위한 협조 사항과 오류 관리를 위한 제어 정보를 규정함
시간(Timing)	두 기기 간의 통신 속도, 메시지의 순서 제어 등을 규정함

92 [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>	정지-대기(Stop-and-Wait) ARQ는 송신 측에서 한 개의 블록을 전송한 후 수신 측으로부터 응답을 기다리는 방식 • 수신 측의 응답이 긍정 응답(ACK)이면 다음 블록을 전송하고, 부정 응답(NAK)이면 앞서 송신했던 블록을 재전송함 • 블록을 전송할 때마다 수신 측의 응답을 기다려야 하므로 전송 효율이 가장 낮음 • 오류가 발생한 경우 앞서 송신했던 블록만 재전송하면 되므로 구현 방법이 가장 단순함				
<b>연속 ARQ</b>	연속(Continuous) ARQ는 정지-대기 ARQ가 갖는 오버헤드를 줄이기 위해 연속적으로 데이터 블록을 보내는 방식으로, 수신 측에서는 부정 응답(NAK)만을 송신함 <table border="1"> <tr> <td><b>Go-Back-N ARQ</b></td><td>           • 여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내 오면 송신 측이 오류가 발생한 블록 이후의 모든 블록을 재전송함            • 전송 오류가 발생하지 않으면 쉬지 않고 연속적으로 송신이 가능함            • 오류가 발생한 부분부터 모두 재전송하므로 중복 전송의 단점이 있음         </td></tr> <tr> <td><b>선택적 재전송(Selective Repeat) ARQ</b></td><td>           • 여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내오면 송신 측이 오류가 발생한 블록만을 재전송함            • 수신 측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리 회로와 큰 용량의 버퍼가 필요함         </td></tr> </table>	<b>Go-Back-N ARQ</b>	• 여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내 오면 송신 측이 오류가 발생한 블록 이후의 모든 블록을 재전송함 • 전송 오류가 발생하지 않으면 쉬지 않고 연속적으로 송신이 가능함 • 오류가 발생한 부분부터 모두 재전송하므로 중복 전송의 단점이 있음	<b>선택적 재전송(Selective Repeat) ARQ</b>	• 여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내오면 송신 측이 오류가 발생한 블록만을 재전송함 • 수신 측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리 회로와 큰 용량의 버퍼가 필요함
<b>Go-Back-N ARQ</b>	• 여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내 오면 송신 측이 오류가 발생한 블록 이후의 모든 블록을 재전송함 • 전송 오류가 발생하지 않으면 쉬지 않고 연속적으로 송신이 가능함 • 오류가 발생한 부분부터 모두 재전송하므로 중복 전송의 단점이 있음				
<b>선택적 재전송(Selective Repeat) ARQ</b>	• 여러 블록을 연속적으로 전송하고, 수신 측에서 부정 응답(NAK)을 보내오면 송신 측이 오류가 발생한 블록만을 재전송함 • 수신 측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리 회로와 큰 용량의 버퍼가 필요함				
<b>적응적 ARQ</b>	적응적(Adaptive) ARQ는 전송 효율을 최대화하기 위해 데이터 블록의 길이를 채널의 상태에 따라 그때그때 동적으로 변경하는 방식 • 전송 효율이 제일 좋음 • 제어 회로가 매우 복잡하고 비용이 많이 들어 현재 거의 사용되지 않고 있음				

## 93 [전문가의 조언]

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

## OSI 7계층

<b>물리 계층 (Physical Layer)</b>	전송에 필요한 두 장치 간의 실제 접속과 절단 등 기계적, 전기적, 기능적, 절차적 특성에 대한 규칙을 정의함
<b>데이터 링크 계층 (Data Link Layer)</b>	• 두 개의 인접한 개방 시스템들 간에 신뢰성 있고 효율적인 정보 전송을 할 수 있도록 함 • 오류의 검출과 회복을 위한 오류 제어 기능을 함 • 프레임의 순서적 전송을 위한 순서 제어 기능을 함
<b>네트워크 계층 (Network Layer, 망)</b>	• 개방 시스템들 간의 네트워크 연결을 관리하는 기능과 데이터의 교환 및 중계 기능을 함 • 네트워크 연결을 설정, 유지, 해제하는

<b>계층)</b>	기능을 함 • 경로 설정(Routing), 데이터 교환 및 중계, 트래픽 제어, 패킷 정보 전송을 수행함
<b>전송 계층 (Transport Layer)</b>	• 논리적 안정과 균일한 데이터 전송 서비스를 제공함으로써 종단 시스템(End-to-End) 간에 투명한 데이터 전송을 가능하게 함 • 종단 시스템 간의 전송 연결 설정, 데이터 전송, 연결 해제 기능을 함 • 주소 설정, 다중화, 오류 제어, 흐름 제어를 수행함
<b>세션 계층 (Session Layer)</b>	• 송·수신측 간의 관련성을 유지하고 대화 제어를 담당하는 계층임 • 대화(회화) 구성 및 동기 제어, 데이터 교환 관리 기능을 함
<b>표현 계층 (Presentation Layer)</b>	코드 변환, 데이터 암호화, 데이터 압축, 구문 검색, 정보 형식(포맷) 변환, 문맥 관리 기능을 함
<b>응용 계층 (Application Layer)</b>	• 사용자(응용 프로그램)가 OSI 환경에 접근할 수 있도록 서비스를 제공함 • 응용 프로세스 간의 정보 교환, 전자 서함, 파일 전송 등의 서비스를 제공함

## 94 [전문가의 조언]

샤논의 정의와 관련해서는 다양한 문제가 출제되고 있습니다. 샤논의 표본화 이론, 정의, 통신 용량을 구하는 공식, 통신 용량을 늘리기 위한 방법을 알아두세요.

- **샤논(Shannon)의 표본화 이론** : 어떤 신호  $f(t)$ 가 의미를 지니는 최고 주파수보다 2배 이상의 주파수로 균일한 시간 간격 동안 채집된다면 이 채집된 데이터는 원래의 신호가 가진 모든 정보를 포함함
- **샤논(Shannon)의 정의** : 전송 회선의 대역폭과 신호, 잡음을 고려하여 정의한 통신 용량

$$C = W \log_2(1 + (S/N))$$

C : 통신 용량, W : 대역폭, S : 신호 전력, N : 잡음 전력

## • 전송로의 통신 용량을 늘리기 위한 방법

- 주파수(채널) 대역폭을 늘린다.
- 신호 세력을 높인다.
- 잡음 세력을 줄인다.

## 95 [전문가의 조언]

디지털 변조 방식과 관련해서는 각 변조 방식들의 변조 원리만 알고 있으면 맞힐 수 있는 문제가 많습니다. 이 문제에서 확실히 정리하고 넘어가세요.

## 디지털 변조 방식

- **진폭 편이 변조(ASK, Amplitude Shift Keying)** : 2진수 0과 1을 서로 다른 진폭의 신호로 변조하는 방식
- **주파수 편이 변조(FSK, Frequency Shift Keying)** : 2진수 0과 1을 서로 다른 주파수로 변조하는 방식
- **위상 편이 변조(PSK, Phase Shift Keying)** : 2진수 0과 1을 서로 다른 위상을 갖는 신호로 변조하는 방식
- **직교 진폭 변조(QAM, Quadrature Amplitude Modulation)** : 진폭과 위상을 상호 변환하여 신호를 얻는 변조 방식

## 96 [전문가의 조언]

매체 접근 제어 방식과 관련해서는 주로 CSMA/CD 방식의 개념을 묻는 문제가 출제되는데, 세부적인 특징까지 알아야

풀 수 있는 문제도 종종 출제됩니다. 이 문제에서 확실히 정리해 두세요.

#### CSMA/CD 방식

- CSMA/CD 방식은 CSMA 방식에서 충돌이 발생하는 문제점을 해소하기 위해 CSMA 방식에 충돌 검출 기능과 충돌 발생 시 재송신하는 기능을 부가한 방식이다.
- CSMA/CD 방식은 통신 회선이 사용 중이면 일정 시간 동안 대기하고, 통신 회선상에 데이터가 없을 때에만 데이터를 송신하며, 송신 중에도 전송로의 상태를 계속 감시한다.
- 송신 도중 충돌이 발생하면 송신을 중지하고, 모든 노드에 충돌을 알린 후 일정 시간이 지난 다음 데이터를 재송신한다.
- 버스형 LAN에 가장 일반적으로 이용된다.
- 전송량이 적을 때 매우 효율적이고 신뢰성이 높다.
- 알고리즘(처리 기법)이 간단하다.
- 노드 장애가 시스템 전체에 영향을 주지 않으며, 장애 처리가 간단하다.
- 일정 길이 이하의 데이터를 송신할 경우 충돌을 검출할 수 없다.
- 전송량이 많아지면 충돌이 잦아져서 채널의 이용률이 떨어지고 전송 지연 시간이 급격히 증가한다.

#### 97 [전문가의 조언]

HDLC와 관련해서는 특징, 프레임 구조, 동작 모드 등이 다양하게 자주 출제되고 있습니다. 이 문제에서는 먼저 프레임 구조를 순서대로 기억하고, 제어부에 속하는 프레임들의 개별적인 기능을 정리하세요.

#### HDLC 프레임의 구조

- **플래그(Flag)** : 프레임의 시작과 끝을 나타내는 고유한 비트 패턴(01111110)
- **주소부(Address Field)** : 송·수신국을 식별하기 위해 사용. 불특정 다수에게 전송하는 방송용(Broadcast)은 '11111111', 시스템에 의해 임의로 수신국이 지정되는 시험용(No Station)은 '00000000'을 사용
- **제어부(Control Field)**
  - 프레임의 종류를 식별하기 위해 사용. 제어부의 첫 번째, 두 번째 비트를 사용하여 구별함
  - 정보 프레임(Information Frame) : 제어부가 '0'으로 시작하는 프레임으로, 사용자 데이터를 전달하는 역할을 함
  - 감독 프레임(Supervisor Frame) : 제어부가 '10'으로 시작하는 프레임으로, 오류 제어와 흐름 제어를 위해 사용됨
  - 비번호 프레임(Unnumbered Frame) : 제어부가 '11'로 시작하는 프레임으로, 링크의 동작 모드 설정과 관리 및 오류 회복을 위해 사용됨
- **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 [전문가의 조언]

ARQ 종류는 물론 각각의 전송 원리도 파악하고 있어야 한다고 했죠? 이 문제를 틀렸다면, 92번 문제의 [전문가의 조언]을 참조하여 ARQ 종류는 물론 각각의 전송 원리도 확실히 파악하고 넘어가세요.

#### 99 [전문가의 조언]

펄스 코드 변조(PCM) 방식과 관련해서는 펄스 코드 변조(PCM) 순서를 묻는 문제가 자주 출제됩니다. 순서를 외올 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

**100** Link State 방식이란 라우팅 정보에 변화가 생길 경우, 변화된 정보만 네트워크 내의 모든 라우터에게 알리는 방식으로 OSPF가 대표적입니다.

#### [전문가의 조언]

자주 출제되지 않는 문제입니다. 주요 라우팅 프로토콜의 특징을 가볍게 읽어보고 넘어가세요.

- **RIP(Routing Information Protocol)** : 현재 가장 널리 사용되는 라우팅 프로토콜로, 소규모 동종의 네트워크(자율 시스템, AS) 내에서 효율적이며, 라우팅 정보를 30초마다 네트워크 내의 모든 라우터에 알림
- **EGP(Exterior Gateway Protocol)** : 자율 시스템(AS) 간의 라우팅, 즉 게이트웨이 간의 라우팅에 사용되는 프로토콜
- **BGP(Border Gateway Protocol)** : 자율 시스템(AS) 간의 라우팅 프로토콜로, 초기에 BGP 라우터들이 연결될 때에는 전체 경로 제어표(라우팅 테이블)를 교환하고, 이후에는 변화된 정보만을 교환함