

1과목 : 데이터 베이스

1. 그래프의 특수한 형태로 노드(Node)와 선분(Branch)으로 되어 있고, 정점 사이에 사이클(Cycle)이 형성되어 있지 않으며, 자료 사이의 관계성이 계층 형식으로 나타나는 비선형 구조는?

- ① tree ② network
③ stack ④ distributed

<문제 해설>

비선형 구조 - 트리, 그래프

[해설작성자 : 민동v]

사이클형성 x 트리

사이클형성 o 네트워크(망),그래프

[해설작성자 : 낙옌음]

2. 다음 BETWEEN 연산의 의미와 동일한 것은?

```
SELECT *
FROM 성적
WHERE (점수 BETWEEN 90 AND 95)
      AND 학과 = "컴퓨터공학과"
```

- ① 점수 ≥ 90 AND 점수 ≤ 95
- ② 점수 > 90 AND 점수 < 95
- ③ 점수 > 90 AND 점수 ≤ 95
- ④ 점수 ≥ 90 AND 점수 < 95

<문제 해설>

between a and b 는 a 이상 b 이하

[해설작성자 : 흥인흥]

3. 다음 자료에 대하여 삽입(insertion) 정렬 기법을 사용하여 오름차순으로 정렬하고자 한다. 1회전 후의 결과는?(오류 신고가 접수된 문제입니다. 반드시 정답과 해설을 확인하시기 바랍니다.)

5, 4, 3, 2, 1

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

<문제 해설>

1회전 : 4 5 3 2 1 (4,5 비교)

2회전 : 3 4 5 2 1

(3.4 비교했을 시 3이 작으므로 앞으로 감 **만약 3이 아니라 6이었으면 3,6 비교 6이 크니까 그자리 그대로, 4,6 비교 6이 크니까 그 자리 그대로)

똑같은 방법으로

3회전 : 2 3 4 5 1

4회전 : 1 2 3 4 5

[해설작성자 : 모리귀여웁]

4. SQL View(뷰)에 대한 설명으로 틀린 것은?

- ① 뷰(View)를 제거하고자 할 때는 DROP 문을 이용한다.
- ② 뷰(View)의 정의를 변경하고자 할 때는 ALTER 문을 이용한다.
- ③ 뷰(View)를 생성하고자 할 때는 CREATE 문을 이용한다.
- ④ 뷰(View)의 내용을 검색하고자 할 때는 SELECT 문을 이용한다.

<문제 해설>

뷰는 변경이 안되므로 삭제후 (DROP->CREATE)로 다시 재 생성해야 합니다.

[해설작성자 : 사켈]

뷰는

1. 가상테이블(물리적으로 구현되는 테이블이 아니다)
2. 시스템 검색은 일반 테이블과 동일
3. 논리적 데이터 독립성을 제공
4. 하나의 뷰를 삭제하면 그 뷰를 기초로 정의된 다른 뷰도 자동으로 삭제
5. 접근 제한을 통한 보안 제공
6. 실제 데이터가 저장되지 않는다.
7. 독자적인 인덱스를 가질 수 없다.
8. 뷰를 이용한 또 다른 뷰의 생성이 가능하다.
- 9.필요한 데이터만 뷰로 정의해서 처리하여 사용자의 관리가 용이
10. 뷰는 변경이 안되어 지우고 다시 만들어야 함. <-
[해설작성자 : 내일 시험인데 불합격...]

5. 다음 설명에 해당하는 스키마는?

물리적 저장 장치의 입장에서 본 데이터베이스 구조로서 실제로 데이터베이스에 저장될 레코드의 형식을 정의하고 저장 데이터 항목의 표현방법, 내부 레코드의 물리적 순서 등을 나타낸다.

- ① conceptual schema ② internal schema
③ external schema ④ definition schema

<문제 해설>

외부 스키마(External Schema) : 프로그래머나 사용자의 입장에서 데이터베이스의 모습으로 조직의 일부분을 정의한 것
개념 스키마(Conceptual Schema) : 모든 응용 시스템과 사용자들이 필요로하는 데이터를 통합한 조직 전체의 데이터베이스 구조를 논리적으로 정의한 것
내부 스키마(Internal Schema) : 전체 데이터베이스의 물리적 저장 형태를 기술하는 것

[추가 해설]

외부 external 스키마

- 공용의 의미보다는 어느 개인이나 특정 응용에 한정된 논리적 데이터 구조
- 데이터베이스의 개별 사용자나 응용프로그래머가 접근하는 데이터베이스를 정의
- 사용자의 관점을 기술

개념 Conceptual 스키마

- the overall logical structure 전체적으로 통합된 논리적 데이터 구조로서, 접근 권한, 제약조건, 보안 정책, 무결성 규칙을 명세
 - 단순히 스키마라고도 하며, 조직이나 기관 전체에서 정의
 - 모든 응용 시스템과 사용자가 필요로 하는 데이터를 통합한 조직 전체의 데이터베이스로 하나만 존재
- Only one it can exist per database
- 범 기관적 입장에서 본 데이터베이스를 정의

내부 internal 스키마

- 데이터베이스 시스템 구조에서 데이터가 실제로 저장되는 것과 관계
- 물리적 저장장치의 입장에서 본 데이터베이스 구조 <-
- 실제로 데이터베이스에 저장될 레코드의 형식을 저장하고 저장 데이터 항목의 표현 방법, 내부 레코드의 물리적 순서등을 나타냄
- It defines how the data are physically arranged on a storage device 데이터가 기억장치에 배치되는 방법을 정의

- as seen by a system programmer or system designer
 시스템 프로그래머나 시스템 디자이너 관점의 스키마
 [해설작성자 : 내일 시험인데 불합격...]

6. 데이터베이스 내에서 데이터들이 불필요하게 중복되어 릴레이션 조작시 예기치 못한 곤란한 현상을 무엇이라고 하는가?

- ① Normalization ② Bug
 ③ Anomaly ④ Error

<문제 해설>

이상(anomaly) 현상

데이터의 중복으로 인해 릴레이션을 처리할때 발생하는 곤란한 현상

데이터 베이스 사용자의 의도와는 다르게 다른데이터가 삽입 삭제 갱신되는 현상

[해설작성자 : 으이그으]

이상 Anomaly

- 데이터의 중복으로 인하여 관계 연산을 처리할 때 곤란한 현상이 발생하는 것 <-

여러 종속 관계를 하나의 릴레이션에 표현하기 때문에 발생

- 삽입 이상 : 불필요하고 원하지 않는 데이터도 함께 삽입해야 되거나 삽입이 되지 않는 경우

- 삭제 이상 : 한 튜플을 삭제함으로써 연쇄 삭제 현상으로 인한 정보의 손실

- 갱신 이상 : 튜플 중에서 일부 속성만을 갱신함으로써 정보의 모순성이 발생하는 현상

[해설작성자 : 내일 시험인데 불합격...]

7. 다음 전위식(prefix)을 후위식(postfix)으로 옳게 표현한 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 -

<문제 해설>

전위식 -/*A+BCDE 를

중위식으로 먼저 변경하면 A*(B+C)/D-E 가 되고

다시 중위식을 후위식으로 변경하면 ABC+*D/E- 가 됩니다.

[해설작성자 : 민동v]

전위식 : -/*A+BCDE

중위식 : A*(B+C)/D-E

후위식 : ABC+*D/E-

[해설작성자 : HackedSniper]

쉬운 변환방법 설명

1. 연산자에 따라 묶는다..(이거 못하시면 이론 다시 ㄱㄱ)
 (- (/ (* A (+ B C)) D) E) -

2. 연산자를 각 괄호 뒤로 뺀다..(중위식으로 할때는 연산자를 문자 사이로 하시면 됩니다.)
 (((A (B C) +) * D) / E) -

3. 괄호를 제거한다.

A B C + * D / E -

[해설작성자 : 뽀뽀]

▶ 다른 거 한 문제 더 맞추고 말자. 시간 낭비 오지고, 문

제의 더러움이 지린다.

▶ 괄호 묶는 것 나오는 이론 어디서 어떻게 찾아? 그래서 여기에 올림.

전위식(prefix) : 연산자가 앞에 있다..반대로 후위식(postfix)는 뒤에 연산자가 있다.

사례) 전위식 : - / * a + b c d, 후위식 : a b c + * d / e -

전위식(prefix)을 후위식(postfix)으로 변환 공식 : 가운데서부터 (연산자, 변수, 변수) 형태로 괄호를 묶고, 연산자를 변수와 변수 뒤에 놓는다..(후위식에서 전위식으로 바뀔 때는 과정은 모두 같고 연산자 위치를 앞으로 당기면 된다.)

※ 주의 : 괄호로 묶은 것은 변수로 취급한다.

- / * a + b c d →

1단계 : (- (/ (* a (+ b c)) d) e) → (연산자, 변수, 변수) 형태로 괄호로 묶는다.

2단계 : (- (/ (* a (b c +)) d) e) → (b + c)에서 b와 c 뒤에 +를 넣는다.

3단계 : (- (/ (a (b c +) *) d) e) → a와 (b c +) 뒤에 *를 넣는다.

4단계 : (- ((a (b c +) *) d /) e) → (a (b c +) *)와 d 뒤에 /를 넣는다.

5단계 : (((a (b c +) *) d /) e -) → ((a (b c +) *) d /)와 e 뒤에 -를 넣는다.

5단계 : a b c + * d / e - → 괄호를 풀어 준다.

전위식(prefix)을 중위식(infix)으로 변환 공식 : 가운데서부터 (연산자, 변수, 변수) 형태로 괄호를 묶고, 연산자를 변수와 변수 가운데 집어넣는다.

※ 주의 : 괄호로 묶은 것은 변수로 취급한다.

- / * a + b c d →

1단계 : (- (/ (* a (+ b c)) d) e) → (연산자, 변수, 변수) 형태로 괄호로 묶는다.

2단계 : (- (/ (* a (b + c)) d) e) → (+ b c)에서 b와 c 사이에 +를 넣는다.

3단계 : (- (/ (a * (b + c)) d) e) → a와 (b + c) 사이에 *를 넣는다.

4단계 : (- ((a * (b + c)) / d) e) → (a * (b + c)와 d 사이에 /를 넣는다.

5단계 : (((a * (b + c)) / d) - e) → (a * (b + c) / d)와 e 사이에 -를 넣는다.

6단계 : a * b + c / d - e → 괄호를 제거한다.

※ 주의 : 미리 괄호를 제거 해 버리면 괄호로 묶어 변수처럼 취급해야 할 항이 헷갈려 틀릴 수가 있으니 괄호를 맨 마지막에 제거한다..괄호로 묶은 항을 변수처럼 취급해야 하는 것에 주의해야 한다.

[해설작성자 : 수아아빠]

8. 트랜잭션의 특성 중 아래 내용에 해당하는 것은?

시스템이 가지고 있는 고정요소는 트랜잭션 수행 전과 트랜잭션 수행 완료 후에 같아야 한다.

- ① 원자성(atomicity) ② 일관성(consistency)
 ③ 격리성(isolation) ④ 영속성(durability)

<문제 해설>

원자성: 트랜잭션 연산은 데이터베이스에 모두 반영되든지 아니면 전혀 반영되지 않아야 한다

일관성: 트랜잭션이 그 실행을 성공적으로 완료하면 언제나 일관성 있는 데이터베이스 상태로 변환한다..

격리성 : 둘이상의 트랜잭션이 동시에 병행 실행되는 경우 어느 하나의 트랜잭션 실행중에 다른 트랜잭션의 연산이 끼여들 수 없다

영속성: 성공적으로 완료된 트랜잭션의 결과는 시스템이 고장나더라도 영구적으로 반영되어야 한다.

[해설작성자 : 취업 가즈아!]

9. 관계데이터 모델의 무결성 제약 중 기본키 값의 속성 값이 널(null)값이 아닌 원자 값을 갖는 성질은?

- ① 개체 무결성 ② 참조 무결성
- ③ 도메인 무결성 ④ 튜플의 유일성

<문제 해설>

개체 무결성 : 릴레이션에서 기본키를 구성하는 속성은 널(NULL)값이나 중복값을 가질 수 없음을 의미합니다.
참조 무결성 : 외래키 값은 NULL이거나 참조 릴레이션의 기본키 값과 동일해야한다는 의미입니다, 즉 릴레이션은 참조할 수 없는 외래키 값을 가질 수 없습니다.
도메인 무결성 : 특정 속성의 값이 그 속성이 정의된 도메인에 속한 값이어야 한다는 규정입니다.
튜플의 유일성 : 무결성 제약 조건에 해당하지 않는 보기입니다.

[해설작성자 : 푸딩맛좀비]

10. 양 방향에서 입·출력이 가능한 선형 자료구조로 2개의 포인터를 이용하여 리스트의 양쪽 끝 모두에서 삽입·삭제가 가능한 것은?

- ① 데크(Deque) ② 스택(Stack)
- ③ 큐(Queue) ④ 트리(Tree)

<문제 해설>

1. 데크(Double Ended Queue) : 삽입과 삭제가 리스트의 양쪽 끝에서 모두 발생할 수 있는 자료구조이며 스택과 큐의 장점만 따서 구성한 것입니다..입력이 한쪽에서만 발생하고 출력은 양쪽에서 일어날 수 있는 입력 제한과 입력은 양쪽에서 일어나고 출력은 한쪽에서만 이루어지는 출력 제한이 있습니다.
2. 스택 : 리스트의 한쪽 끝으로만 자료의 삽입, 삭제 작업이 이루어지는 자료 구조입니다..가장 나중에 삽입된 자료가 가장 먼저 삭제되는 후입선출(LIFO)방식으로 자료를 처리합니다.
3. 큐 : 한쪽에서는 삽입 작업이 이루어지고 다른 한쪽에서는 삭제 작업이 이루어지도록 구성된 자료구조입니다..시작과 끝을 표시하는 두 개의 포인터가 있습니다.
4. 트리 : 트리는 비선형 자료구조입니다.

[해설작성자 : 푸딩맛좀비]

11. 병행제어 기법 중 로킹에 대한 설명으로 옳지 않은 것은?

- ① 로킹의 대상이 되는 객체의 크기를 로킹 단위라고 한다.
- ② 파일은 로킹 단위가 될 수 있지만 레코드는 로킹 단위가 될 수 없다.
- ③ 로킹의 단위가 작아지면 로킹 오버헤드가 증가한다.
- ④ 로킹의 단위가 커지면 데이터 베이스 공유도가 저하한다.

<문제 해설>

로킹(Locking) : 주요 데이터의 액세스를 상호 배타적으로 하는 것·데이터베이스, 파일, 레코드, 필드 등은 로킹 단위가 될 수 있다
· 로킹 단위가 크면 로크 수가 작아 관리하기 쉽지만 병행성 수준이 낮아지고,로킹 단위가 작으면 로크수가 많아 관리하기 복잡해 오버헤드가 증가하지만 병행성 수준이 높아짐

[해설작성자 : 썸]

로킹 단위

- 로킹 단위가 작으면 로크 수가 많아져 오버헤드가 커지고 관리하기는 복잡하지만 공유도 즉 병행성 수준이 높아진다
데이터베이스, 파일, 레코드, 등 로킹단위가 될 수 있다

<-

[해설작성자 : 내일 시험인데 불합격...]

로크단위가작으면 로크수증가>오버헤드증가

12. NoSQL의 설명으로 틀린 것은?

- ① Not Only SQL의 약자이다.
- ② 비정형 데이터의 저장을 위해 유연한 데이터 모델을 지원한다.
- ③ 전통적인 관계형 데이터베이스관리시스템과는 다른 비관계형(non-relational) DBMS이다.
- ④ 정규화를 전제로 하고 있어 갱신 시에 저장 공간이 적게 든다.

<문제 해설>

일관성 모델을 이용하는 데이터의 저장 및 검색을 위한 매커니즘을 제공하며, 정규화를 전제로 하는것이 아닌 단순 검색 및 추가 작업을 위한것으로 성능 이익을 내는것이 목적인 언어입니다.

[해설작성자 : 푸딩맛좀비]

13. 트랜잭션의 실행이 실패하였음을 알리는 연산자로 트랜잭션이 수행한 결과를 원래의 상태로 원상 복구 시키는 연산은?

- ① COMMIT 연산 ② BACKUP 연산
- ③ LOG 연산 ④ ROLLBACK 연산

<문제 해설>

COMMIT 연산:

하나의 트랜잭션에 대한 작업이 성공적으로 끝나고 수행결과를 관리자에게 알려주는 연산

ROLLBACK 연산:

하나의 트랜잭션 처리가 비정상적 종료되어 해당 트랜잭션을 재시작 OR 폐기

성공-커미트~(COME HERE) 실패-롤백(가라)

[해설작성자 : 붉은해성]

14. 관계 대수에 대한 설명으로 옳지 않은 것은?

- ① 릴레이션을 처리하기 위한 연산의 집합으로 피연산자가 릴레이션이고 결과도 릴레이션이다.
- ② 원하는 정보와 그 정보를 어떻게 유도하는가를 기술하는 절차적 특징을 가지고 있다.
- ③ 일반 집합 연산과 순수 관계 연산이 있다.
- ④ 수학의 Predicate Calculus 에 기반을 두고 있다.

<문제 해설>

관계대수 : 절차적 언어 / 순수 관계 연산(select, project, join, division)과 일반 집합연산(union, intersection, difference, cartesian product)

관계해석 : 비절차적 언어 / Predicate Calculus에 기반을 둠 / 튜플 및 도메인 관계해석

[해설작성자 : sdk]

관계대수(릴레이션 조작을 위한 연산의집합.기술적인특성)의 순수관계 연산자

- 릴레이션을 처리하기 위한 연산의 집합으로 피연산자가 릴레이션이고 결과도 릴레이션이다.

- Select : 수평적 부분집합. 시그마를 사용

- Project : 수직적 부분집합. 파이를 이용

- Join

- Division : 두 릴레이션 A,B에 대해 릴레이션의 모든 조건을 만족하는 튜플들을 릴레이션 A에서 분리해 내어 프로젝션함.

보기 4의 Predicate Calculus에 기반을 두고있다는 관계해석에 대한 설명이다..<-

[해설작성자 : 내일 시험인데 불합격...]

15. 데이터베이스 로그(log)를 필요로 하는 회복 기법은?

- ① 즉각 갱신 기법 ② 대수적 코딩 방법
 ③ 타임 스탬프 기법 ④ 폴딩 기법

<문제 해설>

데이터베이스 로그(log)를 필요로 하는 회복 기법의 종류

1. 연기 갱신 기법(Deferred Update)
2. 즉각 갱신 기법(Immediate Update)
3. 그림자 페이지 대체 기법(Shadow Paging)
4. 검사점 기법(Check Point)

[해설작성자 : 푸딩맛쵸비]

회복(recovery)

- 트랜잭션들을 수행하는 도중 장애로 인해 손상된 데이터베이스를 손상되기 이전의 정상적인 상태로 복구시키는 작업.

- 회복 기법의 종류로는

1. Deferred Modification(연기 갱신 기법)
2. Immediate Update(즉각 갱신 기법)
3. Shadow Paging(그림자 페이지 대체 기법)
4. Check Point(검사점 기법)

[해설작성자 : 내일 시험인데 불합격...]

16. What is the quantity of tuples in consist of the relation?

- ① Degree ② Instance
 ③ Domain ④ Cardinality

<문제 해설>

튜플 : 릴레이션을 구성하는 요소로서 행을 의미한다..같은 의미의 용어로는 레코드, 로우가 있으며 튜플은 릴레이션에서 유일해라 하므로 서로다른 키값을 가진다.

* 튜플의 수 를 의미하는 단어는 카디널리티 이다.

[해설작성자 : sdk]

17. 이진 검색 알고리즘에 대한 설명으로 틀린 것은?

- ① 탐색 효율이 좋고 탐색 시간이 적게 소요된다.
 ② 검색할 데이터가 정렬되어 있어야 한다.
 ③ 피보나치 수열에 따라 다음에 비교할 대상을 선정하여 검색한다.
 ④ 비교횟수를 거듭할 때마다 검색 대상이 되는 데이터의 수가 절반으로 줄어든다.

<문제 해설>

3번은 피보나치 검색(Fibonacci Search)에 관한 설명 이진 검색은 찾고자 하는 Key 값을 파일의 중간 레코드 Key 값과 비교하면서 검색하는 방식이다.

[해설작성자 : 꼬마자바]

이분 검색(이진 검색)

- 특정 레코드를 찾기 위해 순서화된 연결 리스트를 찾는 빠른 방법은 그 레코드가 알려져 있는 리스트의

마지막 반쪽 부분을 연속적으로 살피는 것(피보나치 검색 X)<-

[해설작성자 : 내일 시험인데 불합격...]

18. 정규화의 필요성으로 거리가 먼 것은?

- ① 데이터 구조의 안정성 최대화
 ② 중복 데이터의 활성화
 ③ 수정, 삭제시 이상현상의 최소화
 ④ 테이블 불일치 위험의 최소화

<문제 해설>

정규화 : 함수적 종속성 등의 종속성 이론을 이용하여 잘못 설계된 관계형 스키마를 더 작은 속성의 세트로 쪼개어 바람

직한 스키마로 만들어가는 과정이다.

정규화 필요성(목적)

- 데이터 구조의 안정성을 최대화한다.
- 어떤 릴레이션이라도 데이터베이스 내에서 표현 가능하게 만든다.
- 효과적인 검색 알고리즘을 생성 가능
- 중복을 배제하여 삽입, 삭제, 갱신 이상의 발생을 방지한다.
- 데이터 삽입 시 릴레이션을 재구성할 필요성을 줄인다.

2번은 중복데이터를 활성화가 아닌 중복 데이터를 배제한다.

[해설작성자 : 희구리]

19. 순서가 A, B, C, D 로 정해진 입력 자료를 스택에 입력 하였다가 출력할 때, 가능한 출력 순서의 결과가 아닌 것은?

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

<문제 해설>

stack : LIFO

queue : FIFO

[해설작성자 : sdk]

d가 먼저나오려면 abc가 먼저 쌓여있는 상태라서 dcba밖에 안됨

[추가 해설]

ABCD : a 넣고 빼고, b 넣고 빼고, c 넣고 빼고, d 넣고 빼고

CBAD : a 넣고 b 넣고 c넣고 c 빼고 b 빼고 a 빼고 d 넣고 d 빼고

[해설작성자 : 1004 silver gogo]

※ 주의 : a, b, c, d를 차례로 한 번에 다 넣고, 출력을 하는 것이 아니라오. 중간에 입출력이 이뤄지는 것에 주의를 하시오. 그리고 스택은 나중 들어 간 것이 먼저 나온다오.

1. D, A, B, C

▶ a, b, c, d가 모두 들어간 상태라야만 d가 먼저 출력이 되오. 모두 들어간 상태라면 a는 제일 바닥에 깔려있는 상태가 아니겠소? 그러니 d 다음엔 a가 절대로 나올 수가 없는 것 아니겠소. 문제는 늘 같은 형식이니 오답을 외워도 될 듯하오.

2. A, B, C, D

▶ a 넣고 출력, b 넣고 출력, c 넣고 출력, d 넣고 출력하면 출력이 abcd가 되오.

3. A, B, D, C

▶ a 넣고 출력, b 넣고 출력, c 넣고, d 넣고, d 출력, c 출력하면 출력이 abdc가 되오.

4. B, C, D, A

▶ a 넣고, b 넣고 출력, c 넣고 출력, d 넣고 출력, a 출력하면 출력이 bcda가 되오.

LIFO(last in first out) : 스택은 벽돌을 쌓은 것과 같아 맨 처음 바닥에 쌓은 것을 먼저 뺄 수가 없소. 위에서부터 빼야하오. 그러니 맨 마지막에 쌓은 벽돌이, 뺄 때는 맨 처음으로 빼지는 것이오. 이 원리를 생각하면서 입출력을 해 본다면 이해가 될 듯하오.

(1~2년 지나고 나니 모두 잊어버린 듯, 머리가 리셋이 된 듯, 이려고 틀리면 무슨 망신 $\pi\pi$)

[해설작성자 : 수아 아빠]

20. 개체-관계 모델의 E-R 다이어그램에서 사용되는 기호와 그 의미의 연결이 옳지 않은 것은?

- ① 사각형 - 개체 타입
- ② 삼각형 - 속성
- ③ 선 - 개체타입과 속성을 연결
- ④ 마름모 - 관계 타입

<문제 해설>

사각형 - 개체
 선 - 개체 타입과 속성을 연결
 마름모 - 관계
 타원 - 속성
 [해설작성자 : 썬스프링]

E-R 다이어그램

- 개체 간의 관계는 물론 시스템 내의 역할을 하는 모든 개체들, 즉 조직, 부서, 사용자, 프로그램, 데이터를 모두 표시
 - 사각형 - 개체
 - 타원 - 속성 <-
 - 마름모 - 관계
 [해설작성자 : 내일 시험인데 불합격...]

2과목 : 전자 계산기 구조

21. 다음과 같이 표현되는 바이트 머신의 데이터 형식의 명칭으로 가장 옳은 것은?

부호(sign), 지수(exponent), 가수(mantissa)

- ① 고정소수점 데이터(fixed point date)
- ② 가변장 논리 데이터(variable length logical data)
- ③ 부동소수점 데이터(floating point data)
- ④ 팩(pack) 형식의 10진수(decimal number)

<문제 해설>

고정소수점 표현 방식 : 부호, 숫자로 구성, 부호비트는 0 양수 1 음수, 연산속도 고속.
 부동소수점 표현 방식 : 부호, 지수, 가수 로 구성, 2진 실수 연산작업에 사용, 연산속도 느림, 정밀도 높음.
 [해설작성자 : 콩콩이]

고정소수점 표현 방식

1. 10 진수의 존 형식(언팩) 표현
 10진수 한 자리를 표현하기 위해서 1바이트(8비트)를 사용하는 형식이다.
 존 영역은 상위 4비트이며, 항상 1111로 표현한다.
 수치 영역은 하위 4비트이며, 표현하고자 하는 10진수 한 자리 값에 대한 2진수 값을 표시
 마지막 자리의 존 영역에 부호를 표시 (양수(+): 1100 =>C, 음수(-): 1101 =>D)

2. 10진수의 팩 형식 표현

10진수 한 자리를 표현하기 위해서 존 영역 없이 4비트를 사용하는 형식이다.
 최하위 4비트에 부호를 표시 (양수(+): 1100 =>C, 음수(-): 1101 =>D)

3. 2진수의 정수 표현

최상위 1비트는 부호를 나타내고, 나머지 n-1비트는 2진수 값을 나타낸다.
 ① 부호 절대값 형식 표현
 최상위 비트에 부호를 표시. 부호가 양수인 경우에는 최상위 비트를 0으로 하고, 음수인 경우에는 최상위 비트를 1로한다..
 나머지 7비트는 2진수의 절대값을 표현

② 1의 보수 형식 표현

- 음수의 표현에서 부호 비트를 사용하는 대신 1의 보수를 사용
 - 음수의 표현에서 부호 비트를 사용하는 대신 2의 보수를 사용

부동소수점 표현 방식

고정소수점 표현 방식보다 표현 가능한 값의 범위가 큼
 실수 자료형은 4바이트나 8바이트로 표현하는데 4바이트의 부동소수점 표현 방식은 다음과 같이 표현.
 - 부호 (1비트)+지수부 (7비트)+소수부 (24비트)

출처: <http://itdexter.tistory.com/57> [IT_Dexter]

[해설작성자 : 석영호]

22. 다음 ADD 명령어의 마이크로 오퍼레이션에서 t2시간에 수행되어야 할 가장 적합한 동작(A)는? (단, MAR : Memory Address Register, MBR : Memory Buffer Register, M(addr) : Memory, AC : 누산기이다.)

t0 : MAR ← MBR(addr)
 t1 : MBR ← M(MAR)
 t2 : (A)

- ① AC ← MBR ② MBR ← AC
- ③ M(MBR) ← MBR ④ AC ← AC + MBR

<문제 해설>

*마이크로 오퍼레이션 : LOAD (AC ← M) , ADD(AC ← AC + M) , STORE(M ← AC)
 [해설작성자 : 좋은아빠]

마이크로 오퍼레이션 -> 마이크로 오퍼레이션

[해설작성자 : HackedSniper]

23. 모듈러스-14 카운터는 몇 가지의 상태를 가지며, 이 카운터를 구성하기 위한 최소의 플립플롭의 수는 몇 개인가?

- ① 상태 : 13가지, 플립플롭 : 3개
- ② 상태 : 14가지, 플립플롭 : 4개
- ③ 상태 : 15가지, 플립플롭 : 5개
- ④ 상태 : 16가지, 플립플롭 : 6개

<문제 해설>

Modulo-n 카운터에 필요한 플립플롭 수 -> $\log_2(\text{밀수})2^n$
 $\text{Modulo-14} = \log_2(\text{밀수})2^{14}$
 $14 = 2^x$
 $x = 3.807$ [반올림]
 $x = 4$ [최소 필요한 플립플롭 수]
 [해설작성자 : 지우개닷컴]

모듈러스(modulo)-N 카운터는 N개의 상태를 갖는 카운터를 말합니다..클럭펄스(clock pulse)가 생성될 때마다 미리 정해진 순서대로 상태가 반복되는 순차회로를 뜻하며 사건의 발생 횟수를 세거나 동작 순서를 제어하는 타이밍 신호를 만드는데 이 카운터(counter)를 사용합니다..

n bit의 2진 카운터가 가장 흔한 형태이며 n개의 플립플롭(기억장치)로 구성, 2^n 개의 상태를 갖습니다.

$14 = 2^x$

14개의 상태, 밑을 2로 갖는 14의 로그(=이진로그로서 $\log_2 14$)개의 플립플롭

[해설작성자 : 출처: 네이버 블로그]

24. 다음 중 SDRAM의 동작에 대한 설명으로 가장 옳지 않은 것은?

- ① 여러 개의 내부 뱅크들(Banks)에서 동시 액세스가 진행된다.
- ② 액세스가 진행되는 동안 CPU가 대기한다.
- ③ 버스 클럭에 동기화되어 정보가 전송된다.
- ④ 여러 개의 데이터들을 연속으로 전송하는 버스트 모드를 지원한다.

<문제 해설>

SDRAM(Synchronous Dynamic Random Access Memory) Synchronous동기 -> CPU가 대기(X) 동기(O)

[해설작성자 : 강 외우셈]

25. 전체 기억장치 액세스 횟수가 50 이고, 원하는 데이터가 캐시에 있는 횟수가 45 라고 할 때, 캐시의 미스율(miss ratio)은?

- ① 0.1 ② 0.2
- ③ 0.8 ④ 0.9

<문제 해설>

캐시의 미스율 = 1 - 캐시에 적중되는 횟수 / 전체기억장치 액세스 횟수

$$1 - (45/50) => 1 - 0.9 = 0.1$$

[해설작성자 : 원동민]

[해설 추가]

잘 읽어보시면 쉬운 문제입니다

전체 기억장치 액세스 중에 캐시가 있는것은 50개 중에 45개가 맞았다는 거겠죠? $45/50 = 90/100 = 0.9$ (적중률) 그럼 나머지 5개는 안맞다는거니깐 $5/50 = 10/100 = 0.1$ (미스율) 또는 $1 - \text{적중률}$

[해설작성자 : 야매]

26. 입출력장치의 인터럽트 우선순위를 하드웨어적으로 결정하는 방식은?

- ① Daisy Chain ② Handshake
- ③ Polling ④ Strobe

<문제 해설>

직렬 우선순위 부여방식은 데이지 체인입니다.

인터럽트가 발생하는 모든 장치를 1개의 회선으로 직렬로 연결하는데 우선순위가 높은 장치를 선두에, 나머지를 우선순위에 따라 차례로 연결합니다.

병렬 우선순위 부여방식은 인터럽트가 발생하는 각 장치를 개별적인 회선으로 연결합니다.

[해설작성자 : 넌 할 수 있어. 세계 제일의 정보왕, 정보 화이팅 정보왕!]

하드웨어적인 우선순위 결정 방식은 데이지체인, 소프트웨어적인 우선순위 결정방식은 폴링

[해설작성자 : 변정은]

인터럽트 우선순위 판별 방법

- 소프트웨어적인 방법 : 폴링(Polling)

인터럽트 요청 신호 플래그를 차례로 검사하여 인터럽트의 원인을 판별, 장점은 우선 순위 변경이 쉬움

- 하드웨어적인 방법 : 데이지체인(Daisy Chain) : 인터럽트가 발생하는 모든 장치를 직렬로 연결 장점은 빠름 <- [해설작성자 : 내일 시험인데 불합격...]

27. 다음 중 일반 응용프로그램이 직접 접근할 수 없는 레지스터는?

- ① 범용 레지스터 ② 플래그 레지스터
- ③ 인덱스 레지스터 ④ 세그먼트 레지스터

<문제 해설>

세그먼트 레지스터는 총 6개(CS, SS, DS, ES, FS, GS)로 구성되며 각 크기는16비트입니다..각 세그먼트 레지스터가 가리키는 세그먼트 디스크립터와 가상메모리가 조합되어 선형주소가 되며, 페이징 기법에 의해 선형 주소가 최종적으로 물리주소로 변환됩니다..만약 OS에서 페이징을 사용하지 않는다면 선형주소는 그대로 물리주소가 됩니다.

세그먼트 레지스터는 운영체제에서 관리하며 응용프로그램이 직접 접근할 수 없습니다.

[해설작성자 : 푸딩맛좀비]

28. 인스트럭션의 설계 과정에서 고려해야 할 사항이 아닌 것은?

- ① 데이터 구조 ② 연산자의 수와 종류
- ③ 인터럽트 종류 ④ 주소지정 방식

<문제 해설>

명령어 구성에서 중요한 것은 연산자, 주소, 워드의 크기 입니다.

인터럽트랑은 상관 없습니다.

[해설작성자 : 사랑합니다.]

29. DMA에 대한 설명으로 가장 옳은 것은?

- ① 인코더와 같은 기능을 수행한다.
- ② inDirect Memory Acknowledge의 약자이다.
- ③ CPU와 메모리 사이의 속도차이를 해결하기 위한 장치이다.
- ④ 메모리와 입출력 디바이스 사이에 데이터의 주고받음이 직접 행해지는 기법이다.

<문제 해설>

DMA는 Direct Memory Access의 약자입니다.

즉 입출력장치가 direct로 직접 주기억장치(Memory)에 접근(Access)하여 datablock을 입출력하는 방식으로 입출력 전송이 CPU의 레지스터를 경유하지 않고 수행됩니다.

[해설작성자 : 이 밤의 끝을 잡고~~♪ 정보기사공부]

30. 소형계산기(calculator)에서 BCD 코드 대신 excess-3 코드를 많이 사용하는 가장 큰 이유는?

- ① 그래픽 기호의 표현이 용이하다.
- ② 에러 검출이 쉽다.
- ③ 연속된 순간에 하나의 비트만 변화한다.
- ④ 자기 보수가 가능하다.

<문제 해설>

그래픽 기호의 표현이 용이한 것은 허프만 코드, 에러 검출이 쉬운 것은 해밍 코드, 연속된 순간에 하나의 비트만 변화하는 것은 그레이 코드의 특징을 의도한 것 같네요

자기 보수가 가능한 코드는 3초과 코드와 2421 코드가 있는데 3초과 코드는 비가중치코드, 2421 코드는 가중치코드라는 점으로 구분하면 좋을 것 같습니다

[해설작성자 : heuristic]

31. 인터럽트의 우선순위결정과 가장 관계없는 것은?

- ① 트랩 방식 ② 폴링 방식
- ③ 벡터 방식 ④ 데이지 체인 방식

<문제 해설>

Polling - 소프트웨어적 방법

Vectored Interrupt - 하드웨어적인 방법

-> Daisy Chain

-> Parallel - 병렬 우선순위 부여 방식

각각의 세부적인 내용은 원하시면 찾아보시는게 더 좋을 것 같아요^^

[해설작성자 : 사랑합니다.]

내부인터럽트(트랩): 불법적인 명령, 0 나누기 overflow, underflow가 발생할 경우, 보호 영역내의 기억장소참조 인터럽트의 우선순위결정과 관련이 없다.

[해설작성자 : masterPark]

32. 세그먼트에서 부연산을 수행하는데 20 ns가 걸리고, 파이프라인은 4 세그먼트로 구성되어 있으며 100개의 태스크를 순차적으로 수행하는 파이프라인 시스템은 비파이프라인 시스템에 비해 약 몇 배의 속도 향상을 얻을 수 있는가?

- ① 2.81 ② 3.25
 ③ 3.88 ④ 4.08

<문제 해설>

비파이프라인 : 총 수행시간 = 시간 * 세그먼트 갯수 * 태스크 갯수 = 20 * 4 * 100 = 8000

파이프라인 : 총 수행시간 = 시간 * (세그먼트 갯수 + 태스크 갯수 - 1) = 20 * (4+100-1) = 2060

8000/2060 = 약 3.88

[해설작성자 : coo]

33. N 가지의 정보를 2진수 코드로 부호화 하는데 필요한 비트수를 계산하는 방법으로 옳은 것은?

- ① $\frac{n}{\lceil \log_2 N \rceil}$ ② $\frac{n}{\lceil \log_{10} N \rceil}$
 ③ $\frac{\lceil \log_{10} N \rceil}{2}$ ④ $\lceil \log_2 N \rceil$

<문제 해설>

4가지 정보는 2bit로 표현할 수 있음. 00, 01, 10, 11

따라서 N가지 정보는 $\log_2 N$ 으로 표현가능

[해설작성자 : 뿌뿌]

34. 64K DRAM 기억소자를 이용하여 64K바이트 주기억장치를 구성하고자 한다. 이 때 64K DRAM을 몇 개 사용해야 하는가? (단, K=kilo이다.)

- ① 1 ② 2
 ③ 4 ④ 8

<문제 해설>

단위를 통일해야되요~ kilo는 kilobyte가 아닙니다.

64kb(주기억장치)는 비트단위로 변경(DRAM이 bit단위) = 64*8

주기억장치/DRAM = (64*8) / 64 = 8개!

[해설작성자 : bystander]

35. 병렬 가산기를 구성하는 각각의 전가산기 출력 캐리를 미리 예측 및 처리하여 리플캐리 지연을 제거한 가산기로 가장 옳은 것은?

- ① Ripple Carry Adder
 ② Carry Lookahead Adder
 ③ Serial-parallel Adder

④ Carry Save Adder

<문제 해설>

lookahead = 예측

[해설작성자 : 나니와 희구리]

1. RCA(Ripple Carry Adder)란?

▶ Full Adder를 일렬로 연결하여 구성한 덧셈회로이며, 다른 덧셈회로보다 회로구성이 간단하지만 많은 비트수를 계산할 때는 계산하는 시간이 오래 걸린다..

2. CLA(Carry Lookahead Adder)란?

▶ RCA의 느린 계산을 극복하기 위하여 각 비트 단에서 캐리를 미리 예측하여 좀 더 빠른 계산을 할 수 있도록, 설계된 덧셈회로이다..

(RCA회로보다 계산 시간은 빠르지만 사용되는 게이트 수가 많아서 대용량의 데이터용으로 설계하려면 엄청나게 많은 게이트가 필요하다..보통 4비트나 8비트 크기의 모듈로 만들어서 각 모듈을 RCA처럼 일렬로 연결하여 대용량의 데이터 계산에 사용한다.)

3. Serial-parallel Adder : 검색 불가능 아마도 지어낸 용어일 듯 (듣.보.잡)

4. CSA(Carry Save Adder) 란?

▶ 자리올림수 저장 가산기는 이진법에서 3개 그이상의 n 비트수 덧셈을 계산하기위한 컴퓨터 기본설계에 사용하는 가산기의 한 종류이다..입력과 같은 크기의 두개의 수를 출력하는 것이 다른 디지털 가산기와 다르다..하나는 부분적인 덧셈 비트의 반복이고, 다른 하나는 자리올림수 비트의 반복이다.

[해설작성자 : 수아 아빠]

36. 다음 마이크로명령어 형식에 관한 설명으로 가장 옳지 않은 것은?

- ① 조건 필드는 분기에 사용될 제어신호들을 발생시킨다.
 ② 연산 필드가 2개인 경우 2개의 마이크로 연산이 동시에 수행된다.
 ③ 주소 필드는 분기가 발생할 경우 목적지 마이크로명령어 주소로 사용된다.
 ④ 분기 필드는 분기의 종류와 다음에 실행할 마이크로명령어의 주소를 결정하는 방법을 명시한다.

<문제 해설>

1.조건 필드는 분기를 위한 플래그 정보를 나타낸다.

2. 연산 필드는 동시에 수행되는 연산 정보를 나타낸다.

3. 주소 필드는 분기가 발생할 경우 실행할 목적지 마이크로명령어 주소로 사용된다.

4. 분기 필드는 분기의 종류와 다음에 실행할 마이크로명령어의 주소를 결정하는 방법

[해설작성자 : heuristic]

37. 다음 중 1주소 명령어 형식을 따르는 마이크로명령어 MUL A를 가장 바르게 표현한 것은? (단, 보기의 M[A]는 기억장치 A번지의 내용을 의미한다.)

- ① $AC \leftarrow AC \times M[A]$ ② $R1 \leftarrow R2 \times M[A]$
 ③ $AC \leftarrow M[A]$ ④ $M[A] \leftarrow AC$

<문제 해설>

3. $AC \leftarrow M[A]$ = LDA(Load to AC)

4. $M[A] \leftarrow AC$ = STA(Store to AC)

[해설작성자 : 푸딩맛좀비]

38. 일반적으로 CPU가 DMA 제어기로 보내는 정보가 아닌 것은?

- ① I/O 장치의 주소

- ② 연산(쓰기 혹은 읽기)지정자
- ③ CPU 제조 고유 번호
- ④ 전송될 데이터 단어들의 수

<문제 해설>

CPU에서 DMA 제어기로 보내는 자료

- I/O 장치의 주소
- 데이터가 있는 주기억장치의 시작 주소
- DMA를 시작시키는 명령
- 입·출력 하고자 하는 자료의 양
- 입력 또는 출력을 결정하는 명령

[해설작성자 : 푸딩맛좀비]

39. AND 마이크로 동작과 가장 유사한 것은?

- ① insert 동작 ② mask 동작
- ③ OR 동작 ④ packing 동작

<문제 해설>

AND(Masking Operation) : 특정문자 또는 특정 Bit를 삭제시키는 명령으로 Masking 명령이라고도 합니다.

[해설작성자 : 푸딩맛좀비]

▶ Insert operation(삽입 동작)

레지스터 A : 0110 0000

레지스터 B : 0000 1100 (OR연산, 빵이나 우유 중 하나라도 있을 경우)

레지스터 A : 0110 1100 (수행 후)

▶ MASK operation(마스크 동작)

레지스터 A : 0110 1010

레지스터 B : 1111 0000 (AND연산, 빵과 우유가 둘 다 있어야 경우)

레지스터 A : 0110 0000 (수정 후)

▶ OR operation(오아 동작)

insert operation 참조

1. 마이크로 동작의 종류

(1)논리 마이크로 오퍼레이션, (2)시프트 마이크로 오퍼레이션, (3)레지스터 전송 마이크로 오퍼레이션, (4)산술 마이크로 오퍼레이션

▶ 논리 마이크로 오퍼레이션(동작)

선택적 비트 설정(Selective-set): 1비트단위 OR 연산

선택적 보수화(Selective-complement): 1비트단위 XOR 연산

선택적 지움(Selective-clear): 1비트단위 AND 연산 (A B A ∪ B')

마스크 동작(Mask): 해당 비트 그룹 단위 AND연산

삽입 동작(Insert): 해당 비트 그룹 단위 OR 연산

비교 동작(Compare): 레지스터 단위로 비교하는 XOR 연산

[해설작성자 : 수아 아빠]

40. 캐시메모리의 기록정책에서 쓰기(write) 동작이 이루어질 때마다 캐시메모리와 주기억장치의 내용을 동시에 갱신하는 방식으로 가장 옳은 것은?

- ① write-through ② write-back
- ③ write-none ④ write-all

<문제 해설>

1. write-Through : 캐시에 쓰기 동작이 이루어질 때마다 캐시 메모리와 주기억장치의 내용을 동시에 갱신하는 방식, 쓰기 동작에 걸리는 시간이 길다.

2. write-Back : 캐시에 쓰기 동작이 이루어지는 동안은 캐시의 내용만이 갱신되고, 캐시의 내용이 캐시로부터 제거될 때 주기억장치에 복사됩니다.

3. write-Once : 캐시에 쓰기 동작이 이루어질 때 한 번만 기

록하고 이후의 기록은 모두 무시합니다.

[해설작성자 : 푸딩맛좀비]

3과목 : 운영체제

41. 페이징 기법에서 페이지 크기가 작아질수록 발생하는 현상으로 거리가 먼 것은?

- ① 기억장소 이용 효율이 증가한다.
- ② 입·출력 시간이 늘어난다.
- ③ 내부 단편화가 감소한다.
- ④ 페이지 맵 테이블의 크기가 감소한다.

<문제 해설>

*페이지가 작아질 경우

- 페이지 단편화 감소 -한 개의 페이지를 주기억장치로 이동하는 시간이 줄어듦
- 프로세스 수행에 필요한 내용만 주 기억장치에 적재 가능 -기억장치 효율성 높아짐
- 페이지 맵 테이블의 크기가 커짐 -매핑 속도가 늦어짐
- 입 출력 시간 증가

*페이지 크기가 클 경우

- 페이지 맵 테이블의 크기가 작아짐 -매핑 속도가 빨라짐
- 입 출력 효율성이 증가
- 페이지의 단편화 증가
- 한 개의 페이지를 주기억장치로 이동하는 시간이 늘어남
- 불필요한 내용까지도 주 기억장치에 적재됨

[해설작성자 : 쿠갱]

42. Preemptive Scheduling 방식에 해당하는 것은?

- ① FIFO ② FCFS
- ③ HRN ④ RR

<문제 해설>

선점(Preemptive)스케줄링의 종류 : SRT, 선점 우선순위, RR, 다단계 큐, 다단계 피드백 큐 등

1,2,3번 보기는 모두 비선점(Non-Preemptive)스케줄링의 종류에 해당합니다.

[해설작성자 : 푸딩맛좀비]

43. 시스템소프트웨어의 구성에서 처리프로그램과 가장 관계가 없는 것은?

- ① Job Scheduler
- ② Language Translate Program
- ③ Service Program
- ④ Problem Program

<문제 해설>

제어 프로그램(Control Program): 시스템 전체의 작동 상태

감시, 작업의 순서 지정(스케줄링), 작업에 사용되는 데이터 관리, 인터럽트 처리 등의 역할을 수행하는 프로그램

- 감시(Supervisor) 프로그램: 각종 프로그램의 실행과 시스템 전체의 작동상태를 감시, 감독하는 프로그램

- 작업 제어(Job Control) 프로그램: 어떤 업무를 처리하고 다른 업무로의 이행을 다중으로 수행하기 위한 준비 및 그 처리에 대한 완료를 담당하는 프로그램

- 자료 관리(Data Management) 프로그램: 주기억장치와 보조 기억장치 사이의 데이터 전송과 보조기억장치의 자료 갱신 및 유지 보수 기능을 수행하는 프로그램

처리 프로그램: 제어 프로그램의 지시를 받아 사용자가 요구한 문제를 해결하기 위한 프로그램
 - 언어 번역(Language Translate) 프로그램: 원시 프로그램을 기계어 형태의 목적 프로그램으로 번역하는 프로그램(어셈블러, 컴파일러, 인터프리터)
 - 서비스(Service) 프로그램: 컴퓨터를 효율적으로 사용할 수 있는 사용 빈도가 높은 프로그램
 - 문제(Problem) 프로그램: 특정 업무 및 해결을 위해 사용자가 작성한 프로그램
 [해설작성자 : 썸]

44. 다음과 같은 Task List에서 SJF방식으로 Scheduling할 경우 Task 2의 종료 시간을 구하면? (단, 발생하는 Overhead는 무시한다.)

| Task | 도착시간 | 실행시간 |
|--------|------|------|
| Task 1 | 0 | 6 |
| Task 2 | 1 | 3 |
| Task 3 | 2 | 4 |

- ① 3 ② 6
 ③ 9 ④ 13

<문제 해설>

Task 1이 0시에 먼저 도착하여 실행을 6시간 한 후 Task 2는 1시에 먼저 도착하였지만 Task 1이 먼저 와서 6시간을 실행하였기 때문에 그 이후 3시간을 실행하여 9시간이 된다.
 [해설작성자 : 원동민]

sjf는 비선점 스케줄링 입니다.

비선점 스케줄링은 앞에 처리중이던 프로세스의 처리가 완료되어야지만 다음 프로세스를 실행할 수 있습니다.

45. UNIX에서 사용자에게 대한 파일의 접근을 제한하는데 사용되는 명령어는?

- ① chmod ② du
 ③ fork ④ cat

<문제 해설>

2. du : 해당 디렉토리의 정보 확인
 3. fork : 새로운 프로세스 생성
 4. cat : 해당 파일 내용을 화면에 표시
 [해설작성자 : 푸딩맛썸]

46. 프로세스들 간의 메모리 경쟁으로 인하여 지나치게 페이지 폴트가 발생하여 전체 시스템의 성능이 저하되는 현상은?

- ① Fragmentation ② Thrashing
 ③ Locality ④ Prepaging

<문제 해설>

스래싱(Thrashing)

:하나의 프로세스가 작업 수행 과정에서 수행하는 기억장치 접근에서 지나치게 페이지 폴트가 발생하여 프로세스 수행에 소요되는 시간보다 페이지 이동에 소요되는 시간이 더 커지는 현상..

오류율이 클수록 스래싱이 많이 발생한 것이고, 스래싱으로 인해 전체 시스템의 성능 및 처리율은 저하된다. 다중 프로그래밍의 어느 시점을 넘어서면 스래싱의 빈도가 높아진다..

[해설작성자 : 김은희]

[추가해설]

Thrashing 현상 방지 방법

- 다중 프로그래밍의 정도를 적정수준으로 유지
- 부족한 자원 증설
- 일부 프로세스 중단
- 페이지 부재 빈도 조절
- 워킹 셋 유지
- 적정프레임 수 제공

[해설작성자 : 네비자동실행 앱 : haewon.tistory.com]

1. 단편화(Fragmentation) : 분할된 주기억장치에 프로그램을 할당하고 반납하는 과정을 반복하면서 사용되지 않고 남는 기억장치의 빈 공간 조각

3. 구역성(Locality) : 프로세스가 실행되는 동안 주기억장치를 참조할 때 일부 페이지만 집중적으로 참조하는 성질이 있다는 이론

4. 프리페이징(Prepaging) : 처음의 과도한 페이지 부재를 방지하기 위해서 필요할 것 같은 모든 페이지를 한꺼번에 페이지 프레임에 적재하는 기법

[해설작성자 : 오픈채팅 베베]

47. 주기억장치의 사용자 영역을 일정 수의 고정된 크기로 분할하여 준비상태 큐에서 준비 중인 프로그램을 각 영역에 할당하여 수행하는 기법은?

- ① 가변분할 기억장치 할당
 ② 고정분할 기억장치 할당
 ③ 교체 기법
 ④ 오버레이 기법

<문제 해설>

가변분할 할당 : 다중분할 할당의 한 방법으로, 프로그램을 주기억장치에 적재할 때 함수 등 필요한 크기로 영역을 분할하는 기법

고정분할 할당 : 다중분할 할당의 한 방법으로, 주기억장치의 사용자 영역을 일정 수의 고정크기로 분할하여 사용하는 기법
 오버레이 기법 : 프로그램의 크기가 주기억장치보다 클 때, 프로그램을 여러 조각으로 분할한 후 필요한 조각 순서대로 주기억장치에 적재한다..실행중 주기억장치의 메모리가 부족하면 이미 존재하는 조각에 새로운 조각을 중첩하여 적재한다.

[해설작성자 : ck]

48. 한정된 시간 내 자료를 분석하여 정해진 시간에 반드시 작업을 처리하여야 하는 시스템은?

- ① Batch Processing
 ② Online Processing
 ③ Real Time Processing
 ④ Time Sharing Processing

<문제 해설>

실시간처리 시스템(Real Time Processing)

- 데이터 발생 즉시, 또는 데이터 처리 요구가 있는 즉시 처리하여 결과를 산출하는 방식
- 은행의 온라인 업무, 좌석예약업무, 인공위성, 군함 등의 제어 업무 등
- 시간에 제한을 두고 수행되어야 하는 작업에 사용됨

[해설작성자 : 7인의 사무라이]

1. 일괄처리 시스템 - 데이터를 모아서 한꺼번에 처리한다.. 효율이 좋음

2. 다중 프로그래밍 시스템 - 하나의 cpu를 이용, 여러 개의 프로그램을 동시에 실행한다.

3. 시분할 시스템 - 여러 프로그램을 약간의 시간차를 두고 번갈아 실행하여, 동시에 실행되는 느낌을 줌.

4. 다중 처리 시스템 - 여러 개의 cpu를 이용, 여러 개의 프로그램을 동시에 실행한다.

5. 실시간 처리 시스템 - 데이터 처리의 요구가 발생한 즉시 처리한다.
 6. 다중 모드 처리 시스템 - 일괄, 시분할 처리 시스템 등의 여러 시스템들을 중복 사용하는 시스템
 7. 분산 처리 시스템 - 여러 개의 컴퓨터를 연결하여 하나의 작업을 처리. (게임으로 치면 보스 레이드)

49. 다음 디스크 스케줄링과 관련된 방법 중 그 성격이 다른 하나는?

- ① C-SCAN ② FCFS
 ③ SLTF ④ SSTF

<문제 해설>

C-SCAN - Circular SCAN, 한쪽 방향으로만 헤드를 이동하며 서비스할 끝에 도착하면 처음부터 진행
 FCFS - First Come First Serve, 작업의 요청 순서대로 서비스 받음
 SSTF - Shortest Seek Time First, 현재의 헤더 위치에서 탐색거리가 가장 짧은 요청이 먼저 처리됨
 SLTF - Shortest Latency Time First, 헤더의 이동이 거의 없는 고정 헤드 장치(드럼 등)에 쓰임

출처: <http://fingerdev.tistory.com/9> [FingerDev (핑거데브)]
 [해설작성자 : ㄴㅇㄹ]

50. 프로세스의 상태 전이에 속하지 않는 것은?

- ① Dispatch ② Spooling
 ③ Wake up ④ Workout

<문제 해설>

프로세서의 상태 전이

- ① dispatch (준비상태→실행상태)
 준비 상태의 프로세스들 중에서 우선 순위가 가장 높은 프로세스를 선정하여 CPU를 할당함으로써 실행상태로 전환한다.
 ② timerrunout (실행상태→준비상태)
 CPU의 지정된 할당 시간을 모두 사용한 프로세스는 다른 프로세스를 위해 다시 준비 상태로 되돌아간다.
 ③ block (실행상태→대기상태)
 실행중인 프로세스가 입출력 명령을 만나면 인터럽트가 발생하여 입출력 전용 프로세서에게 CPU를 양도하고 자신은 대기 상태로 전환한다.
 ④ wake-up (대기상태→준비상태)
 입출력 완료를 기다리다가 입출력 완료 신호가 들어오면 대기 중인 프로세스는 준비 상태로 전환한다.
 [해설작성자 : ㄴㅇㄹ]

1. Dispatch : 준비 -> 실행 상태
 2. Spooling : 제출된 작업들을 스푼 공간인 디스크에 수록하여 보류상태로 만들
 3. 대기 -> 준비 상태
 4. Workout : 오답

프로세스의 상태

1) 프로세스 상태 전이도
 프로세스 상태 전이도는 CPU를 사용할 때와 사용하기 전, 입출력을 행하는 상태로 실행 (Run), 준비 (Ready), 대기 (Block)로 구분할 수 있고 이러한 상태를 논리적으로 도식해 보면 다음과 같다.

- (1)보조 기억 장치에 존재하는 A.EXE이 사용자에게 의해 실행한다.
 (2) A.EXE는 스푼 (Spool)로 이동한다.
 (3) 스푼에 있던 실행 파일이 주기억 장치로 적재 (Loading)된다 (Job Scheduler에 의해서)
 (4) 주기억 장치에 있는 A.EXE는 준비 (Ready) 상태가 된다.

- (5) A.EXE가 실행 (Run) 상태가 된다 (Process Scheduler에 의해서)
 (6) A.EXE는 정해진 시간 할당량(Time Slice, Quantum)에 따라 다시 준비 상태가 된다..(Time Run Out).
 (7) A.EXE가 자체적인 입출력(I/O)행위가 요구되면 대기 (Block) 상태가 되고 채널에 의해 입출력이 행해진다.
 (8) 입출력 수행이 끝나면 다시 준비 상태가 된다..(Wake Up).
 (9) 보류 상태에서 준비가 안된 입출력이 요구되면 잠시 중단되어 중단된 봉쇄 상태 (Suspended)가 된다.
 (10) 보류 상태에서 실행 상태로 전이될 수 없으며 준비 상태에서 대기 상태로 전이될 수도 없다.

※ 상태전이는 이렇게 외웠는데, 이 난데 없는 문제는 원지

- ㅍㅍㅍ
 - 디스패치(dispatch) : 준비 -> 실행 상태
 - 할당시간 초과(time runout) : 실행 -> 준비상태
 - 대기(block) : 실행 -> 대기 상태
 - 깨움(wake up) : 대기 -> 준비 상태
 ※ 스푼링 : 디스크를 매우 큰 버퍼처럼 사용하는 것.
 [해설작성자 : 수아아빠]

51. 스레드의 특징으로 가장 옳지 않은 것은?

- ① 실행 환경을 공유시켜 기억장소의 낭비가 줄어든다.
 ② 프로세스 외부에 존재하는 스레드도 있다.
 ③ 하나의 프로세스를 여러 개의 스레드로 생성하여 병행성을 증진시킬 수 있다.
 ④ 프로세스들 간의 통신을 향상시킬 수 있다.

<문제 해설>

스레드는 프로세스 내부에만 존재.
 [해설작성자 : 엠뿐나]

52. 운영체제를 자원 관리자(Resource Manager)라는 관점으로 접근했을 때, 자원들을 관리하는 과정을 순서대로 가장 옳게 나열한 것은?

- ㉠ 프로세스에 배당한 자원을 회수하는 과정
 ㉡ 어떤 프로세스에게 언제, 어떤 자원을 할당할 것인가를 결정하는 분배 정책 수립 과정
 ㉢ 시스템 내 모든 자원들의 상태를 파악 하는 과정
 ㉣ 자원을 배당하고 운영함으로써 수립된 정책을 수행 하는 과정

- ① ㉡ → ㉣ → ㉢ → ㉠
 ② ㉡ → ㉢ → ㉠ → ㉣
 ③ ㉢ → ㉣ → ㉠ → ㉡
 ④ ㉢ → ㉠ → ㉣ → ㉡

<문제 해설>

자원은 메모리, 프로세스, 장치, 파일 등이 있고 운영체제를 통해 관리한다.

1. 자원을 누가, 얼마나, 어느부분을 사용하는지 파악한다..(다)
 2. 프로세스에게 언제, 어떤 자원을 할당시킬지 결정한다..(나)
 3. 자원을 할당하고 회수한다..(라, 가)
 [해설작성자 : ㄷㅈㄷㅎㄱ 전자정보통신 작은용]

53. 페이지 교체기법 중 LRU와 비슷한 알고리즘 이며, 최근에 사용하지 않은 페이지를 교체하는 기법으로 시간 오버헤드를 줄이기 위해 각 페이지마다 참조 비트와 변형 비

트를 두는 교체기법은?

- ① FIFO ② LFU
 ③ NUR ④ OPT

<문제 해설>

NUR (Not Used Recently)

- 최근 사용하지 않은 페이지 교체
- 최근 사용여부를 확인하기 위해(시간 오버헤드를 줄이기 위해), 각 페이지마다 참조/변형비트 사용
- 교체순서 : 00 - 01 - 10 - 11 (0 : 참조비트, 1 : 변형비트)

[해설작성자 : 비슷한문제 해설에서 따옴]

페이지 교체 기법 - 새 데이터를 주기억장치에 등록할 때 공간이 부족할 경우 먼저 있던 데이터를 빼고 넣는 행위를 말한다.

FIFO - 제일 오래된 데이터 제거

LFU - 가장 적게 사용 되었던 데이터 제거

LRU, NUR - '최근' 가장 적게 사용되었던 데이터 제거

OPT - 앞으로 가장 적게 사용될 것 같은 데이터를 예측하여 제거(신뢰성 떨어짐)

54. 분산 운영체제의 개념 중 강결합(TIGHTLY-COUPLED) 시스템의 설명으로 옳지 않은 것은?

- ① 프로세서간의 통신은 공유 메모리를 이용한다.
 ② 여러 처리기들 간에 하나의 저장장치를 공유한다.
 ③ 메모리에 대한 프로세서 간의 경쟁 최소화가 고려되어야 한다.
 ④ 각 사이트는 자신만의 독립된 운영체제와 주기억장치를 갖는다.

<문제 해설>

강결합 시스템은 동일 운영체제하에서 여러 개의 프로세서가 하나의 메모리를 공유하여 사용하는 시스템으로, 다중 처리 시스템이라고도 합니다.

[해설작성자 : 푸딩맛쵸비]

약결합 - 독립되어있다 (결합이 약함)

강결합 - 공유,융쳐있다 (결합이 강함)

55. 운영체제의 운용 기법 종류 중 다음 설명에 가장 부합하는 것은?

CPU의 시간을 각 사용자에게 균등하게 분할하여 사용하는 시스템으로 모든 컴퓨터 사용자에게 똑같은 서비스를 하는것을 목표로 하고 있다. CPU의 전에 사용 시간을 작은 작업 시간량(time slice)으로 나누어서 그 시간 동안만 번갈아 가면서 CPU를 할당하며 각 작업을 처리한다.

- ① Batch Processing System
 ② Multi Programming System
 ③ Time Sharing System
 ④ Real Time System

<문제 해설>

'시간을 각 사용자 에게 균등하게 '분할!!!!'
 분할==공유 라 볼수있다

영어 해석하면 Time(시간) Sharing(공유)
 3번이답임

국어,영어조금만 알아도 푸는문제

[해설작성자 : WannaCry]

[추가 해설]

다시 말해 시분할입니다.

[해설작성자 : 응 나는 합격]

56. 모니터에 대한 설명으로 옳지 않은 것은?

- ① 자원 요구 프로세스는 그 자원 관련 모니터 진입부를 반드시 호출한다.
 ② 한 순간에 하나의 프로세스만이 모니터에 진입할 수 있다.
 ③ 정보 은폐의 개념을 사용한다.
 ④ 모니터 외부의 프로세스는 모니터 내부 데이터를 직접 액세스 할 수 있다.

<문제 해설>

모니터는 외부의 프로시저를 직접 액세스할 수 없으며, 모니터의 경계에서 상호 배제가 시행됩니다.

[해설작성자 : 푸딩맛쵸비]

57. Dead Lock 발생의 필요충분조건이 아닌 것은?

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

<문제 해설>

교착상태의 발생조건

1. Circular Wait (환형대기) : 이미 자원을 가진 프로세스가 앞이나 뒤의 프로세스 자원을 요구한다.
2. Hold and Wait (점유와대기) : 이미 자원을 가진 프로세스가 다른 자원의 할당을 요구한다.
3. Mutual Exclusion (상호배제) : 한 번에 한 개의 프로세스만이 공유 자원을 사용할 수 있어야 한다.
4. Non-Preemption (비선점) : 프로세스에 할당된 자원은 사용이 끝날 때까지 강제로 빼앗을 수 없다.

[해설작성자 : strawsoojin]

58. FIFO 스케줄링에서 3개의 작업 도착시간과 CPU 사용시간(burst time)이 다음 표와 같다. 이 때 모든 작업들의 평균 반환시간(turn around time)은? (단, 소수점 이하는 반올림 처리한다.)

| 작업 | 도착시간 | CPU 사용시간 (burst time) |
|-------|------|--------------------------|
| JOB 1 | 0 | 13 |
| JOB 2 | 3 | 35 |
| JOB 3 | 8 | 10 |

- ① 12 ② 36
 ③ 58 ④ 69

<문제 해설>

평균반환시간은

JOB 1부터 순차대로

도착시간 0이니깐 1은 그대로 반환시간 13

작업2는 13+35(1작업시간,2작업시간 더함) = 48인데 2의 도착시간이 3이니깐 48-3을해줌 = 45가됨

그리고 작업3으로 넘어가서 작업1,2,3의 사용시간을 모두더해줌= 13+35+10=58

그후에 작업3의 도착시간을 뺀다 = 58-8 =50

이제 1,2,3 반환시간 전부 더한것에서 나누기3을한다

작업1 = 13(도착시간이 0이니깐)

작업2 = 35+13=48 --> 48-3(작업2의 도착시간)= 45

작업3 = 13+35+10=58 --> 58 - 8(작업3의도착시간) = 50

$13 + 45 + 50 = 108 \rightarrow 108/3(\text{작업이3개니깐}) = 36$
[해설작성자 : WannaCry]

반환시간 = 총 CPU 사용시간(CPU Burst) - 도착시간
JOB 1 = $13 - 0 = 13$
JOB 2 = $(13 + 35) - 3 = 45$
JOB 3 = $(13 + 35 + 10) - 8 = 50$

따라서, 평균은 각 반환시간을 모두 더해 작업 개수만큼 나누면 되므로
답은 $(13 + 45 + 50) / 3 = 36$

59. UNIX에서 현재 디렉토리 내의 파일 목록을 확인하는 명령어는?

- ① ls ② cat
③ fack ④ cp

<문제 해설>

2. cat : 해당 파일의 내용을 화면에 표시
3. fack : 파티션 스캔, 오류검출
4. cp : 파일복사
[해설작성자 : 푸딩맛썸비]

60. 다음 설명에 해당하는 디렉토리 구조는?

- UNIX 에서 사용하는 디렉토리 구조이다.
- 각 디렉토리는 서브디렉토리나 파일을 가질 수 있다.
- 디렉토리의 생성과 파괴가 비교적 용이하다.
- 디렉토리의 탐색은 포인터를 사용하며, 경로명은 절대와 상대 경로명을 사용한다.

- ① 1단계 디렉토리 구조
② 2단계 디렉토리 구조
③ 비순환 그래프 디렉토리 구조
④ 트리 디렉토리 구조

<문제 해설>

UNIX = 트리!!
[해설작성자 : 고마츠 나나]

4과목 : 소프트웨어 공학

61. 소프트웨어 비용 추정모형(estimation models)이 아닌 것은?

- ① COCOMO ② Putnam
③ Function-Point ④ PERT

<문제 해설>

* 소프트웨어 비용 산정 방법

1. LOC방법 - 프로그램의 라인 수를 평가하여 비용을 산정
2. COCOMO 방법 - 소프트웨어의 복잡도 또는 원시 프로그램의 규모에 따라 산정
3. Putnam 모형
4. 기능점수(Function-Point) 모형
[해설작성자 : 문민범]

PERT는 프로그램 평가와 검토기법으로 프로젝트일정계획
[해설작성자 : jc7979]

(가) 소프트웨어 비용 추정 모형

1) ▶ 1) COCOMO 모형 :
보험이 제한한 모형으로 프로그램 규모에 따라 비용을 산정함

니다.

Organic(5만 라인 이하), Semi-Detached(30만 라인 이하), Embedded(그 이상)
비용 산정 단계와 적용할 인수에 따라 Basic, Intermediate, Detailed형으로 구분합니다.
LOC(Lines of code)기법을 사용합니다.
예상 비용 = (낙관 정도 + 4*기대 정도 + 비관 정도)/6

2) ▶ 2) Putnam 모형 :

소프트웨어 개발 주기의 각 단계별로 요구할 인력의 분포를 가정하는 모형입니다.
Putnam 모형과 Rayleigh Noden 곡선을 기초로 개발한 자동화 추정 도구로 SLIM이 있습니다.

3) ▶ FP 모형 :

FP(Funtion Point) = 전체 기능 점수 * $[0.65 + (0.1 * \text{총 영향 정도})]$
요구 기능을 증가시키는 인자별로 가중치를 부여하여 기능의 점수를 계산하여 비용을 산정하는 방식

(나) 프로젝트 일정 계획

4) ▶ PERT

작업들의 선후 관계를 표현 작업 패키지들의 순서에 관한 정보로 사이클이 없는 방향 그래프

▶ WBS(Work Breakdown Structure)

작업 분할 구조 프로젝트 수행을 위해 개발 업무를 분할하여 계층 구조로 표현 최하위 수준의 작업을 작업 패키지(work package)라고 하며 정량적으로 측정 가능한 입력 물과 출력 물을 가진 프로젝트 계획과 관리를 위한 기초 자료

▶ CPM

임계 경로 방법 임계 경로는 시작에서 종료 작업까지의 경로 중 가장 긴 경로 임계 경로 상의 작업들은 프로젝트의 일정 준수를 위해 지연이 허용되지 않는 작업 임계 경로 상에 있지 않은 작업들은 여유 시간을 가짐

▶ 간트(Gantt) 차트

막대 모양으로 프로젝트 작업들의 순차 또는 병행 순서를 보여주는 차트 상단에 시간축을 표시, 작업별로 막대를 표시 막대는 작업 시간에 맞추어지며 길이는 소요 시간을 의미 인력 배정 등의 자원 활용 계획에도 사용됨
[해설작성자 : 수아 아빠]

62. LOC기법에 의하여 예측된 총 라인수가 36,000라인, 개발에 참여할 프로그래머가 6명, 프로그래머들의 평균 생산성이 월간 300라인일 때 개발에 소요되는 기간을 계산한 결과로 가장 옳은 것은?

- ① 5개월 ② 10개월
③ 15개월 ④ 20개월

<문제 해설>

LOC 기법의 비용 산정 공식

노력(인/월수 M/M)=(참여 인력/월)×개발 기간=추정 LOC/1인당 월 평균 생산 코드 라인 수
개발 비용=(M/M)×단위 비용(1인당 월 평균 인건비)
개발 기간=(M/M)/참여 인원
생산성=LOC/(M/M)

=> $(36000[\text{추정총라인수}]/300[\text{월평균 생산코드수}]) / 6(\text{참여인원}) = 20$

[해설작성자 : crow]

더욱 간단하게 풀이

1. 개발참여프로그래머 6명×프로그래머 평균 생산성 월간

300라인 = 1,800라인(6명의 월간 평균생산성)
2. 예측된 총라인수 36,000 / 1,800 라인 = 20개월
[해설작성자 : 그린컴퓨터학원 화이팅]

63. CORBA에서 인터페이스 정의 언어는?

- ① IDL ② ADL
③ CSL ④ UML

<문제 해설>

IDL : 인터페이스 정의 언어. CORBA의 큰 특징 중 하나는 여러 언어의 지원. // C,C++,자바 ,COBOL

ADL : 아키텍처 기술 언어. 소프트웨어 품질과 정확성을 증진하기 위해 소프트웨어 아키텍처를 모형화하고 분석하는 수단을 제공

CSL : 제어 시뮬레이션 언어

UML : 객체지향분석과 설계를 위한 모델링 언어. Booch, Rumbaugh, Jacobson이 주장하는 각각의 객체지향방법론 중에서 장점들을 통합하여 여러 가지 방법론들을 모두 표현할 수 있도록 만든 언어

[해설작성자 : 윤호상]

IDL은 Interface Description Language로, SW개발자다 메시지를 정의할 수 있게 해주는 선언적 언어이며, CORBA에서 정보를 송수신하는 데 필요한 언어이다.

[해설작성자 : 송민]

64. 소프트웨어 개발 영역을 결정하는 요소 중 다음 사항과 가장 관계있는 것은?

- 소프트웨어에 의해 간접적으로 제어되는 장치와 소프트웨어를 실행하는 하드웨어
- 기존의 소프트웨어와 새로운 소프트웨어를 연결하는 소프트웨어
- 순서적 연산에 의해 소프트웨어를 실행하는 절차

- ① 기능 ② 성능
③ 제약 조건 ④ 인터페이스

<문제 해설>

S/W 개발 영역을 결정하는 주요 요소 : 처리될 데이터, 소프트웨어 기능, 성능, 제약조건, 인터페이스, 신뢰도 등

<인터페이스>

• 소프트웨어에 의해 간접적으로 제어되는 장치와 소프트웨어를 실행하는 프로세서나 하드웨어

• 운영체제, 서브루틴 패키지과 같이 새로운 소프트웨어에 연결되어야 하는 소프트웨어

• 키보드나 기타 I/O 장치를 통해 소프트웨어를 사용하는 사람

• 순서적인 연산에 의해 소프트웨어를 실행하는 절차

[해설작성자 : HackedSniper]

65. 블랙박스 테스트 기법에 관한 다음 설명과 가장 부합하는 것은?

여러 버전의 프로그램에 동일한 검사 자료를 제공하며 동일한 결과가 출력되는지 검사하는 기법이다.

- ① Boundary Value Analysis
② Cause Effect Graphing Testing
③ Equivalence Partitioning Testing
④ Comparison Testing

<문제 해설>

* 블랙 박스 테스트

1. Equivalence Partitioning Testing - 동치 분할 테스트
2. Cause Effect Graphing Testing - 원인 효과 그래픽 테스트

3. Comparison Testing - 비교 테스트

[해설작성자 : 문민범]

경계값 분석(Boundary Value Analysis) - 입력 조건의 경계값을 검사 사례로 선정하여 검사

[추가 해설]

2. Cause Effect Graphing Testing : 입력 데이터 간의 관계와 출력에 영향을 미치는 상황을 체계적으로 분석하여 효용성 높은 검사 사례를 선정하여 검사

3. Equivalence Partitioning Testing: 입력 자료에 초점을 맞춰 검사사례 만들고 검사

[해설작성자 : 출처 <https://under-desk.tistory.com/10>]

66. 유지보수의 종류 중 소프트웨어 테스트 동안 밝혀지지 않은 모든 잠재적인 오류를 수정하기 위한 보수 형태로서 오류의 수정과 진단 과정이 포함되는 것은?

- ① Perfective maintenance
② Adaptive maintenance
③ Preventive maintenance
④ Corrective maintenance

<문제 해설>

1. 완전화(perfective)보수 : 소프트웨어의 본래 기능에 새로운 기능을 추가하거나 성능을 개선하기 위해 소프트웨어를 확장시키는 활동입니다.

2. 적응(adaptive) 보수 : 소프트웨어의 수명 기간 중에 운영체제나 컴파일러와 같은 프로그래밍 환경 변화와 주변장치 또는 다른 시스템 요소가 향상되거나 변경될때 기존의 소프트웨어에 반영하기 위해 수행하는 활동입니다.

3. 예방(preventive)보수 : 미래의 유지보수성 또는 신뢰성을 개선하거나 소프트웨어의 오류 발생에 대비하여 미리 예방수단을 강구해 두는 활동

4. 수정(Corrective)보수 : 시스템을 운영하면서 검사 단계에서 발견하지 못한 잠재적인 오류를 찾아 수정하는 활동

[해설작성자 : 푸딩맛쵸비]

67. 브룩스(Brooks) 법칙의 의미를 가장 옳게 설명한 것은?

- ① 프로젝트 개발에 참여하는 남성과 여성의 비율은 동일해야 한다.
② 새로운 개발 인력이 진행 중인 프로젝트에 투입될 경우 작업 적응 기간과 부작용으로 인해 빠른 시간 내에 프로젝트는 완료될 수 없다.
③ 프로젝트 수행 기간의 단축을 위해서는 많은 비용이 투입되어야 한다.
④ 프로젝트에 개발자가 많이 참여할수록 프로젝트의 완료 기간은 지연된다.

<문제 해설>

브룩스(Brooks)의 법칙 : 소프트웨어 개발 일정이 지연된다고 해서 말기에 새로운 인원을 투입하면 작업 적응 기간과 부작용으로 인해 일정은 더욱 지연된다는 법칙이다.

[해설작성자 : strawsoojin]

68. 럼바우(Rumbaugh)의 객체지향 분석에서 사용되는 분석 활동을 가장 옳게 나열한 것은?

- ① 객체 모델링, 동적 모델링, 정적 모델링
② 객체 모델링, 동적 모델링, 기능 모델링
③ 동적 모델링, 기능 모델링, 정적 모델링

④ 정적 모델링, 객체 모델링, 기능 모델링

<문제 해설>

'객동기' 로 외웁시다

[해설작성자 : WannaCry]

69. 위험 모니터링의 의미를 가장 잘 설명한 것은?

- ① 위험을 이해하는 것
- ② 위험요소들에 대하여 계획적으로 관리하는 것
- ③ 위험 요소 징후들에 대하여 계속적으로 인지하는 것
- ④ 첫 번째 조치로 위험을 피할 수 있도록 하는 것

<문제 해설>

모니터링=계속주시하고있다는=주시만하고있다는

그래서 3번임

2번은 유지보수..?

4번은 대처,조치 정도로 볼수있겠음

[해설작성자 : WannaCry]

2번은 유지보수보다는 위험 관리에 가까운 것 같습니다.

위험 관리:프로젝트 추진과정에서 예상되는 각종 돌발 상황 (위험)을 인식하고 이에 대한 적절한 대책을 수립(관리)하는 일련의 활동

위험 모니터링을 위험 관리로 잘못 봐서 틀렸네요..

[해설작성자 : 안다다씨]

70. 자료 흐름도(DFD)에서 "Process"의 표기 형태는?

- ① 원
- ② 화살표
- ③ 사각형
- ④ 직선(단선, 이중선)

<문제 해설>

원 = 프로세스

화살표 = 흐름

사각형 = 단말

직선(평행선, 이중선) = 저장소

[해설작성자 : 땀땀이꿀]

71. 소프트웨어 재공학이 소프트웨어의 재개발에 비해 갖는 장점으로 가장 거리가 먼 것은?

- ① 위험부담 감소
- ② 비용 절감
- ③ 시스템 명세의 오류억제
- ④ 개발시간의 증가

<문제 해설>

이미 만들어놨던걸 뜯어가지고 그내용을 좀 수정해서 재개발 하는건데

첨부터 다시 설계하고 소스짜고 하는것도아니고 있던거를 그대로 들고와서 조금수정해서 개발하는데

개발시간이 증가될리가없잖아유??

[해설작성자 : WannaCry]

72. 소프트웨어 시스템 명세서의 유지 보수에 대한 설명으로 가장 거리가 먼 것은?

- ① 명세서의 유지 보수란 명세서를 항상 최신의 상태로 만드는 것을 말한다.
- ② 소프트웨어는 계속 수정 보완되기 때문에 명세서도 따라서 보완되지 않으면 일관성을 유지하기 어렵다.
- ③ 최신의 명세서는 필요한 경우 즉시 사용자에게 배포해야 한다.
- ④ 시스템 개발자와 사용자는 동일한 명세서를 사용하기 때문에 시스템의 구조를 사용자도 잘 알고 있어야 한다.

<문제 해설>

사용자는 시스템의 구조를 잘 알고있을 필요가 없다.

[해설작성자 : (induk)원동민]

73. 한 모듈 내의 각 구성 요소들이 공통의 목적을 달성하기 위하여 서로 얼마나 관련이 있는지의 기능적 연관의 정도를 나타내는 것은?

- ① cohesion
- ② coupling
- ③ structure
- ④ unity

<문제 해설>

응집도(Cohesion): 모듈 내부의 처리 요소들 간의 기능적 연관도. 독립적인 기능으로 정의. 응집도가 강할 수록 좋음
 기능적 응집도, 순차적 응집도, 교환적 응집도, 절차적 응집도, 시간적 응집도, 논리적 응집도, 우연적 응집도가 있음
 [해설작성자 : 이정현]

모듈 "내" = cohesion

모듈 "외" = coupling

[해설작성자 : 고마츠 나나]

74. 객체지향에서 정보 은닉과 가장 밀접한 관계가 있는 것은?

- ① Encapsulation
- ② Class
- ③ Method
- ④ Instance

<문제 해설>

Encapsulation : 캡슐화인데요. 여러분들이 감기 드시거나 하셨을때 먹는 약중에 캡슐에 들어 있는 알약 보신적 있나요? 보통 그 캡슐안에 아주 작고 동글 동글한 가루약 같은 작은 알갱이 약이 그 캡슐안에 들어가 있습니다. 캡슐의 기능은 약과 바깥의 공기를 차단해주는 역할도 하고 위, 대장 등 적절한 곳까지 캡슐에 싼여 약성분이 이동할수 있는 기능도 제공합니다.

이와 마찬가지로 객체지향의 캡슐화는 약에 사용되는 캡슐처럼 안과 밖을 구분하여 정보(메소드)를 관리하므로 정보의 접근을 허용 또는 차단을 보다 규칙적이고 쉽게 할수 있습니다.
 [해설작성자 : 밀양금성컴퓨터학원 ☎055-354-3344]

75. 시스템 감사의 종류 중 통합 시스템의 맥락에서 소프트웨어의 실시간 성능을 검사하며, 모든 단계에서 수행되는 것은?

- ① 복구 검사
- ② 보안 검사
- ③ 성능 검사
- ④ 강도 검사

<문제 해설>

성능을 검사하니깐 성능검사!

[해설작성자 : WannaCry]

76. 다음의 자동화 예측 도구들 중 Rayleigh-Norden 곡선과 Putnam의 예측모델에 기반을 둔 것은?

- ① ESTIMACS
- ② SLIM
- ③ SPQR/20
- ④ WICOMO

<문제 해설>

SLIM : Rayleigh-Norden 곡선과 Putnam 예측 모델을 기초로 하여 개발된 자동화 추정 도구

ESTIMACS : FP모형을 기초로 하여 개발된 자동화 추정 도구
 [해설작성자 : 이변주 시험]

77. 결합도(Coupling) 단계를 약한 순서에서 강한 순서로 가장 옳게 표시한 것은?

- ① stamp → data → control → common → content
- ② control → data → stamp → common → content
- ③ content → stamp → control → common → data
- ④ data → stamp → control → common → content

<문제 해설>

데이터 -> 스템프 -> 제어 -> 공통 -> 컨텐츠 결합도 순서
 [해설작성자 : 원동민]

78. 다음 설명에 해당하는 생명주기 모형으로 가장 옳은 것은?

가장 오래된 모형으로 많은 적용 사례가 있지만 요구사항의 변경이 어려우며, 각 단계의 결과가 확인되어야지만 다음 단계로 넘어간다. 선형 순차적 모형으로 고전적 생명 주기 모형이라고도 한다.

- ① 프로토타입 모형(Prototype Model)
- ② 코코모 모형(Cocomo Model)
- ③ 폭포수 모형(Waterfall Model)
- ④ 점진적 모형(Spiral Model)

<문제 해설>

폭포수 모델(waterfall model)은 순차적인 소프트웨어 개발 프로세스(소프트웨어를 만들기 위한 프로세스)로, 개발의 흐름이 마치 폭포수처럼 지속적으로 아래로 향하는 것처럼 보이는 데서 이름이 붙여졌다..이 폭포수 모델의 흐름은 소프트웨어 요구사항 분석 단계에서 시작하여, 소프트웨어 설계, 소프트웨어 구현, 소프트웨어 시험, 소프트웨어 통합 단계 등을 거쳐, 소프트웨어 유지보수 단계에까지 이른다.
 [해설작성자 : 이번주 시험]

79. 유지보수의 활동 종류로 볼 수 없는 것은?

- ① Interfere Maintenance
- ② Adaptive Maintenance
- ③ Perfective Maintenance
- ④ Preventive Maintenance

<문제 해설>

Interfere = 간섭하다
 유지보수랑 관계없죠
 [해설작성자 : 원동민]

80. Software Project의 비용 결정 요소와 가장 관련이 적은 것은?

- ① 개발자의 능력 ② 요구되는 신뢰도
- ③ 하드웨어의 성능 ④ 개발제품의 복잡도

<문제 해설>

S/W 프로젝트 이기 때문에 H/W는 고려하지 않는다.
 [해설작성자 : 실기불고싶다]

5과목 : 데이터 통신

81. HDLC에서 사용되는 프레임의 유형이 아닌 것은?

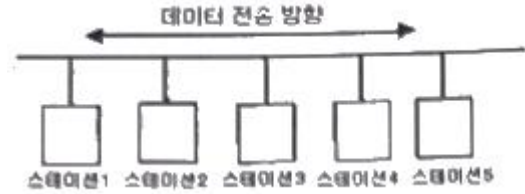
- ① Information Frame ② Supervisory Frame
- ③ Unnumbered Frame ④ Control Frame

<문제 해설>

ISU !! 여러분 모두 정처기 과정을 성공적으로 이수 하시다.
 [해설작성자 : 땡땡이꿀]

I프레임, S프레임, U프레임 정보프레임.감독프레임.비번호 프레임 이라고도한다.

82. 다음 LAN의 네트워크 토폴로지는 어떤 형인가?



- ① 버스형 ② 성형
- ③ 링형 ④ 그물형

<문제 해설>

<Bus형> 하나의 긴 케이블이 네트워크상의 모든 장치를 연결하는 중추 네트워크의 역할을 하는 형태
 <Star형> 각 호스트가 허브라고 불리는 중앙 전송 제어장치와 점대점(Point-to-Point) 링크에 의해 접속되어 있는 초창기 사용 형태.
 <Ring 형> 닫힌 루프 형태로 각 호스트가 자신의 양쪽 호스트와 전용으로 점 대 점으로 연결된 형태
 [해설작성자 : 이번주 시험]

83. 회선을 제어하기 위한 제어 문자 중 실제 전송한 데이터 그룹의 시작임을 의미하는 것은?

- ① SOH ② STX
- ③ SYN ④ DLE

<문제 해설>

<전송 제어 문자>

1. SOH(Start Of Heding) : 헤딩 시작
 2. STX(Start of Text) : 본문 시작, 전송할 데이터 그룹의 시작
 3. SYN(Synchronous idle) : 동기 문자
 4. DLE(Data Link Escape) : 데이터 투과성을 위해 삽입(전송 제어 문자와 전송 데이터 구분을 위해 사용)
- [해설작성자 : 실기도 화이팅!]

84. 8진 PSK의 오류 확률은 2진 PSK 오류 확률의 몇 배인가?

- ① 2배 ② 3배
- ③ 4배 ④ 5배

<문제 해설>

8은 2의 3승임

그니깐 3배

[해설작성자 : WannaCry]

85. 한 전송로의 데이터 전송 시간을 일정한 시간폭(time slot)으로 나누어 각 부 채널에 차례로 분배하는 방식의 다중화 방식은?

- ① 시분할 다중화 ② 주파수분할 다중화
- ③ 위상분할 다중화 ④ 위치분할 다중화

<문제 해설>

일정한 시간폭으로 나눔 -> 시분할

[해설작성자 : WannaCry]

86. OSI 7계층에서 데이터 분할과 재조립, 흐름제어, 오류제어 등을 담당하는 계층은?

- ① 응용 계층 ② 표현 계층
- ③ 세션 계층 ④ 전송 계층

<문제 해설>

응용계층 - 사용자(응용프로그램)가 OSI 환경에 접근할 수 있도록 서비스를 제공함
 표현계층 - 응용계층 --> 세션계층, 세션계층 --> 응용계층

변환하는 기능(코드변환, 형식변환 등)
세션계층 - 송, 수신 측 간의 관련성유지 대화제어, 데이터 교환기능
[해설작성자 : 꼭되자]

보기에 정답이 없으시다는분.. 데이터 분할과 재조립, 흐름제어, 오류제어를 다 만족하는계층은 전송계층(트랜스포트 계층)입니다..
[해설작성자 : 보기에 정답이 있네]

87. 네트워크에 연결된 시스템은 논리주소를 가지고 있으며, 이 논리주소를 물리주소로 변환시켜 주는 프로토콜은?
- ① RARP ② NAR
③ PVC ④ ARP

<문제 해설>
논리주소 = IP 물리주소 = MAC

논리->물리 ARP
물리->논리 RARP
[해설작성자 : WannaCry]

88. X.25에서 오류 제어와 흐름 제어, 가상 회선의 설정과 해제, 다중화 기능, 망 고장 발생 시 회복 메커니즘을 규정하는 계층은?
- ① 링크 계층 ② 물리 계층
③ 패킷 계층 ④ 응용 계층

<문제 해설>
링크 계층(프레임 계층) : 패킷의 원활한 전송을 위해 "데이터 링크"의 제어를 수행하는 계층
물리 계층 : 단말장치(DTE)와 패킷 교환망(DCE)간의 물리적 접속에 관한 인터페이스를 "정의"하는 계층
패킷 계층 : 데이터 전송 제어기능을 수행하는 계층
[해설작성자 : 땀방울]

89. TCP/IP 프로토콜의 계층 구조 중 응용계층에 해당하는 프로토콜로 옳지 않은 것은?
- ① UDP ② Telnet
③ FTP ④ SMTP

<문제 해설>
UDP는 전송계층
[해설작성자 : <induk>원동민]

TCP/IP의 계층
응용 계층 : 응용 프로그램 간의 데이터 송.수신을 제공하며 TELNET,FTP,SMTP,SNMP,HTTP,DNS,WAP 등이 있다.
전송 계층 : 호스트들 간의 신뢰성 있는 통신을 제공하며 TCP,UDP,RTP,RTCP 등이 있다.
인터넷(네트워크)계층 : 데이터 전송을 위한 주소 지정, 경로 설정을 제공하며 IP,ICMP,IGMP,ARP,RARP 등이 있다.
네트워크 액세스 계층 : 실제 데이터(프레임)을 송.수신하는 역할을 하며 Ethernet, IEEE 802, HDLC, X25, RS-232C, PPP등이 있다.
[해설작성자 : 내이름은상그니]

90. 전진오류정정(FEC) 방식에 대한 설명으로 거리가 먼 것은?
- ① 재전송 요구 없이 수신측에서 스스로 오류검사 및 수정을 하는 방식이다.
② 역채널이 필요 없고, 연속적인 데이터 흐름이 가능하다.
③ 데이터 전송 과정에서 오류가 발생하면 송신 측에 재

전송을 요구하는 방식이다.

- ④ 블록 코드와 콘볼루션 코드도 FEC 코드의 종류이다.

<문제 해설>
데이터 전송 과정에서 오류가 발생하면 송신 측에 재전송을 요구하는 방식은 후진 오류 수정(BEC, Backward Error Correction)방식을 설명한 것이다.
[해설작성자 : (induk)원동민]

91. 라우팅 테이블 이 가지고 있는 경로 정보의 세가지 요소가 아닌 것은?
- ① 다음 홉 ② 메트릭
③ 수신지 네트워크 주소 ④ 디폴트 게이트웨이

<문제 해설>
라우팅은 L3장비에 해당하죠 네트워크 id가 C클래스까지 안에서 통신되는게 L2장비인데 게이트웨이는 L2장비가 L3장비 라우팅을 지원하는 장비로 따라가게끔 통신되게끔하게 하는 일종의 문이죠 즉 L3장비의 VLAN IP가 L2장비의 게이트 웨이가 되는데 같은 vlan안에 속해있어야지 통신이 됩니다.
L2 장비의 VLAN IP는 텔넷접속용 IP가 되구요 보통 ip routing을 할때 ip route (수신자 네트워크 주소) 서브네팅 벡스ٹ출입니다.. 예를 들어 디폴트라우팅 ip route 0.0.0.0 0 192.172.1.2라고 가정하면 내밀에 있는 모든 네트워크 정보를 저쪽아이피로 다 보내라 이런뜻입니다.
메트릭값은 라우팅의 최적경로값으로 메트릭값이 낮은놈이 우선순위가 되어서 통신이 되게됩니다.

따라서 문제는 L2장비에서 쓰이는 명령어가 디폴트 게이트웨이죠 만약에 저게 디폴트 라우팅이면 답이 맞는것입니다..이상 시청 전산실 유지보수팀DPN박진우입니다.
[해설작성자 : 박진우]

경로제어표(Routing Table) 구성 3요소
- 다음 홉 주소
- 메트릭(Metric)
- 목적지(수신지)주소
[해설작성자 : 유거스]

92. 192.168.1.0/24 네트워크를 FLSM 방식을 이용하여 3개의 subnet으로 나누고 ip subnet-zero를 적용했다. 이 때 서브네팅 된 네트워크 중 2번째 네트워크의 broadcast IP 주소는?
- ① 192.168.1.127 ② 192.168.245.128
③ 192.168.1.191 ④ 192.168.1.192

<문제 해설>
3개의 서브네팅으로 나누면 서브네팅비트는 2개이고, 2번째 네트워크는 비트로 했을때, 01 이다..나머지 다섯개비트는 0으로 처리하면, 호스트 주소는 192.168.1.64가 된다..세번째 네트워크의 호스트 주소는 192.168.1.128이므로, 그 이전 주소가 두번째 네트워크의 브로드캐스트 주소가 되므로 192.168.1.127 이 된다.
[해설작성자 : 윤호상]

192.168.1.0/24 에서 /24는 서브넳 마스크의 101 24개 있다는 뜻입니다..
즉 11111111 11111111 11111111 00000000 => 255.255.255.0 이며
3개의 서브넳으로 나누기 때문에 네트워크ID는 2비트($2^2 = 2$, $2^2 = 4$)이며,
남은 6비트로 $2^6 = 64$ 씩 호스트가 나뉘집니다.
따라서 첫번째 IP주소의 범위는 192.168.1.0 ~ 63
두번째 IP주소의 범위는 192.168.1.64 ~ 127
세번째 IP주소의 범위는 192.168.1.128 ~ 191 입니다.

문제에서는 두번째 네트워크의 브로드캐스트주소를 물어보는 문제인데,
브로드캐스트 주소는 해당 IP주소범위에서 가장 마지막 주소를 뜻하므로,
두번째 IP주소의 가장 마지막 주소인 192.168.1.127 이 정답이 됩니다.

[해설작성자 : 뽀뽀]

93. 다음 설명에 해당되는 ARQ 방식은?

데이터 프레임을 연속적으로 전송하는 과정에서 NAK를 수신하게 되면, 오류가 발생한 프레임 이후에 전송된 모든 데이터 프레임을 재전송하는 방식이다.

- ① Stop-and-Wait ARQ
- ② Selective-Repeat ARQ
- ③ Go-back-N ARQ
- ④ Sequence-Number ARQ

<문제 해설>

- 1. Stop-and-Wait ARQ : 송신 측에서 한 개의 블록을 전송한 후 수신 측으로부터 응답을 기다리는 방식
- 2. Selective-Repeat ARQ : 오류가 발생한 블록만을 재전송하는 방식

[해설작성자 : 푸딩맛좀비]

94. IEEE에서 규정한 무선 LAN 규격은?

- ① IEEE 802.3 ② IEEE 802.5
- ③ IEEE 802.11 ④ IEEE 801.99

<문제 해설>

- 802.1 = 전체의 구성
- 802.2 = 논리 링크 제어 (LLC) 계층
- 802.3 = CSMA/CD
- 802.4 = 토큰 버스
- 802.5 = 토큰 링
- 802.6 = 도시형 통신망 (MAN)
- 802.11 = 무선 LAN

[해설작성자 : 기출모의고사]

95. 라우팅 프로토콜이 아닌 것은?

- ① RIP ② OSPF
- ③ BGP ④ PPP

<문제 해설>

RIP: 소규모 동종의 네트워크 내에서 효율적인 방법(최대 홉 15로제한)
OSPF: 홉 수에 제한이 없어 대규모 네트워크에서 많이 사용되는 프로토콜로 라우팅 정보에 변화가 있을때 변화된 정보만 네트워크 내의

모든 라우터에 알림.

BGP: 자율 시스템 간의 라우팅 프로토콜

[해설작성자 : 고마츠 나나]

ppp : 주로 두개의 라우터를 접속 할때 사용되며, IETF의 표준 프로토콜.

오류 검출 기능만 사용되며 재전송을 통한 오류복구와 흐름제어기능은 제공되지 않음.

[해설작성자 : 꼭되자]

96. 내부라우팅 프로토콜의 일종으로 링크상태 알고리즘을 사

용하는 대규모 네트워크에 적합한 것은?

- ① RIP(Routing Information Protocol)
- ② BGP(Border Gateway Protocol)
- ③ OSPF(Open Shortest Path First)
- ④ IDRP(Inter Domain Routing Protocol)

<문제 해설>

- 1. RIP : 소규모 네트워크 내에서 효율적인 방법으로, 최대 홉 수를 15로 제한합니다.
- 2. BGP : 자율 시스템(AS) 간의 라우팅 프로토콜로 EGP의 단점을 보완하기 위해 만들어졌습니다.
- 3. OSPF : 홉 수에 제한이 없어 대규모 네트워크에 많이 사용되는 프로토콜입니다.

[해설작성자 : 푸딩맛좀비]

IDRP : IS-IS Interdomain Routing Protocol(IS-IS 도메인간 라우팅 프로토콜)의 약어. 라우터가 다른 도메인의 라우터와 통신하는 방식을 지정하는 OSI 프로토콜.

[해설작성자 : 구글참조]

OSPF (Open Short path first)의약자로 서로 라우터끼리 DR BDR을 정하는데 대장 라우터 오른쪽 라우터 그리고 왼쪽라우터 라우터끼리 계급을 맺는거죠 한마디로 대장라우터가 뒤질 경우 오른쪽라우터가 DR측 대장이 되고 그옆에 라우터가 BDR아되는 것입니다..클래스 리스로 네트워크를 축약하여서 서로 라우터 네이버를 맺은 관계끼리는 라우팅 테이블을 서로 업데이트를 하여 주고 받죠 가장 많이 쓰이는 라우팅프로토콜 이고 static으로 라우팅을 받고 OSPF로 돌리 경우에는 무조건 default information originate란 명령어가 들어가야지 스타틱에서 OSPF로 재분배를 합니다..이상입니다..저명령어가 없으면 안만 스타틱으로 뿌린거 OSPF로 돌릴려고 해도 안되죠

ospf 에서 static routing을 사용해야될 경우는 redistribute static을 설정해야됩니다

default information originate는 default routing에 대한 재분배를 설정하는 것입니다

[해설작성자 : 내일 시험 벼락치기 중]

97. 진폭과 위상을 변화시켜 정보를 전달하는 디지털 변조 방식은?

- ① FM ② QAM
- ③ PSK ④ ASK

<문제 해설>

FM = Frequency Modulation (주파수 변조)

QAM = Quadrature Amplitude Modulation (진폭과 위상 변조)

PSK = Phase Shift Keying (위상 편이 편조)

ASK = Amplitude Shift Keying (진폭 편이 변조)

[해설작성자 : (induk)원동민]

98. 메시지 교환 방식에 대한 설명으로 거리가 먼 것은?

- ① 송신데이터 순서와 수신 순서 불일치
- ② 고정적인 대역폭을 가진 전용 전송로 필요
- ③ 전송 도중 오류 발생 시 메모리에 축적되어 있는 복사본 재전송 가능
- ④ 각 메시지만다 수신 주소를 붙여서 전송

<문제 해설>

고정적인 대역폭을 가진 전송로를 사용하는 방식은 회선 교환 방식에 대한 설명이다.

[해설작성자 : (induk)원동민]

99. 불균형적인 멀티포인트 링크 구성 중 주 스테이션이 각 부 스테이션에게 데이터 전송을 요청하는 회선 제어 방식

은?

- ① Completion ② Polling
 ③ Select-Hold ④ Point to Point

<문제 해설>

2. Polling 방식은 주 PC(스테이션)이 단말기(부 스테이션)에 게 전송해야 할 데이터의 여부를 질의하고 있는 경우 허가하는 방식

[해설작성자 : HackedSniper]

100. HDLC의 데이터 전송 동작모드에 속하지 않는 것은?

- ① NRM ② ABM
 ③ ARM ④ WCM

<문제 해설>

HDLC 전송 동작모드 3가지

NRM (Normal Response Mode, 정규 응답 모드)

ABM (Asynchronous Balanced Mode, 비동기 균형 모드)

ARM (Asynchronous Response Mode, 비동기 응답 모드)

[해설작성자 : (induk)원동민]

본 해설집의 저작권은 www.comcbt.com에 있으며 카페, 블로그등 개인적 활용 이외에 문서의 수정 및 금전적 이익을 취하는 일체의 행위를 금지 합니다.

전자문제집 CBT PC 버전 : www.comcbt.com

전자문제집 CBT 모바일 버전 : m.comcbt.com

기출문제 및 해설집 다운로드 : www.comcbt.com/x

전자문제집 CBT란? 인터넷으로 종이 없이 문제를 풀고 자동 채점하는 프로그램으로 워드, 컴활, 기능사 등의 상설검정에서 사용하는 실제 프로그램 방식입니다.

해설을 제공하며 PC 버전 및 모바일 버전 완벽 연동 교사용/학생용 관리기능도 제공합니다.

오답 및 오탈자가 수정된 최신 자료와 해설은 전자문제집 CBT 에서 확인하세요.

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