2017-국회적-컴퓨터일반-가형-해설

대방고시 전산직/계리직, 하이클래스 군무원 곽후근(gobarian@gmail.com)

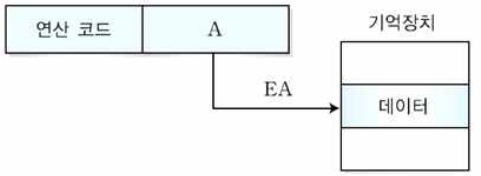
1. CPU의 주소지정 방식 중 간접 주소지정 방식(indirect addressing mode)을 이용하는 이유로 옳은 것은?

① 메모리에 직접 접근 하지 못하게 하여 보안을 강화한다.
② 메모리 접근 속도를 빠르게 한다.
③ 제작 비용을 절감 한다.
④ CPU의 실행 속도를 빠르게 한다.
⑤ 지정 가능한 주소 범위를 확대한다.

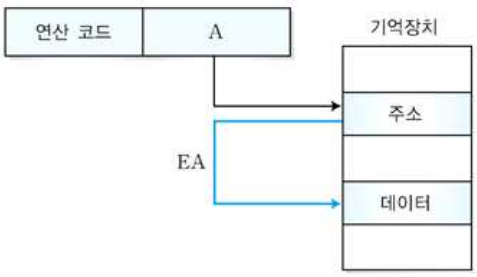
해설)

정답 체크 :

직접 주소 지정 방식은 메모리를 1번 접근하지만 연산 코드를 제외하고 남은 비트들이 주소 비트로 사용되기 때문에 지정할 수 있는 기억장소의 수가 제한된다. 예를 들어, 명령어의 길이(워드의 길이, 기억장치의 폭)가 16비트, 연산 코드(op-code, operation code)가 5비트라고 가정하면 A(오퍼랜드)가 가지는 기억장소의 수는 11비트(=16비트 - 5비트)이다. 그러므로 기억장소의 수는 2048개(2^{11}개), 즉 0번지부터 2047번지까지 접근할 수 있다. 결론은 직접 주소 방식으로는 메모리의 2048 이후의 번지를 접근할 수 없다.



간접 주소 지정 방식은 2번의 기억장치 액세스가 필요하다. 첫 번째는 유효 주소(effective address, 실제 오퍼랜드를 가리키는 주소)가 저장된 곳에 액세스하는 것이고, 두 번째는 유효 주소에 액세스하여 실질적인 데이터를 얻는 것이다. 실행 사이클 동안 두 번의 기억장치 액세스가 필요하지만 최대 기억장치 용량은 CPU가 한 번에 액세스할 수 있는 단어의 길이에 의하여 결정된다(지정 가능한 주소 범위가 확대된다). 예를 들어, 명령어의 길이(워드의 길이, 기억장치의 폭)가 16비트, 연산 코드(op-code, operation code)가 5비트라고 가정하면 A(오퍼랜드를 가리키는 유효 주소가 아님)가 가지는 기억장소의 수는 11비트(=16비트 - 5비트)이다. A를 통해 실제 오퍼랜드(데이터)가 있는 주소(유효 주소)를 가지고 오는데 해당 주소는 워드의 길이(기억장치의 폭)이므로 16비트가 된다. 16비트가 가지는 기억장치의 주소 수는 65536개(2^{16}개), 즉 0번지부터 65535번지까지 접근할 수 있다.



2. 자신이 수행한 행동(action) 에 따른 보상(reward)을 통해 스스로 문제 해결방법을 찾아내도록 하는 기계학습(machine learning) 기법으로 옳은 것은?
① 베이시안 네트워크(bayesian network)
② 지도학습(supervised learning)
③ 군집화(clustering)
④ 경사 하강법(gradient descent method)
⑤ 강화학습(reinforcement learning)
해설)
정답 체크 :
(5) 강화학습 : 어떤 환경 안에서 정의된 에이전트가 현재의 상태를 인식하여, 선택 가능한 행동들 중 보상을 최대화하는 행동 혹은 행동 순서를 선택하는 방법이다. 강화 학습은 또한 입출력 쌍으로 이루어진 훈련 집합이 제시되지 않으며, 잘못된 행동에 대해서도 명시적으로 정정이 일어나지 않는다는 점에서 일반적인 지도 학습과 다르다.
오답 체크 :
(1) 베이시안 네트워크 : 랜덤 변수의 집합과 방향성 비순환 그래프를 통하여 그 집합을 조건부 독립으로 표현하는 확률의 그래픽 모델이다. 예를 들어, 베이지안 네트워크는 질환과 증상 사이의 확률관계를 나타낼 수 있다. 증상이 주어지면, 네트워크는 다양한 질병의 존재 확률을 계산할 수 있다.
(2) 지도학습 : 훈련 데이터(Training Data)로부터 하나의 함수를 유추해내기 위한 기계 학습(Machine Learning)의 한 방법이다. 훈련 데이터는 일반적으로 입력 객체에 대한 속성을 벡터 형태로 포함하고 있으며 각각의 벡터에 대해 원하는 결과가 무엇인지 표시되어 있다.
(3) 군집화 : 주어진 데이터들의 특성을 고려해 데이터 집단(클러스터)을 정의하고 데이터 집단의 대표할 수 있는 대표점을 찾는 것으로 데이터 마이닝의 한 방법이다. 예를 들면, 자율주행차가 주행 도로의 색을 군집화하여 인식하는 것을 들 수 있다.
(4) 경사 하강법 : 1차 근삿값 발견용 최적화 알고리즘이다. 기본 아이디어는 함수의 기울기(경사)를 구하여 기울기가 낮은 쪽으로 계속 이동시켜서 극값에 이를 때까지 반복시키는 것이다. 예를 들어, 경사 하강법은 회사 직원들의 근무 만족도를 1에서 100점 점수로 평가한 데이터가 있다고 가정했을 때. 급여에 따른 직원의 만족도를 예측할 수 있는 AI를 학습하는데 사용한다. 처음에는 만족도 예측에 오차가 발생하다가 경사 하강법으로 극값에 이르게 되면 정확한 만족도 예측을 할 수 있다.

3. 가상메모리(virtual memory) 관리 기법에 대한 설명으로 옳지 않은 것은?
① 프로그램의 크기가 실제 메모리의 크기보다 커도 실행이 가능하다.

② 페이지 방식은 페이지 사상 작업으로 인하여 비용이 증가하고 수행 속도가 감소한다.
③ 페이지 방식은 페이지를 작게 설계하여 내부 단편화를 줄 일 수 있다.
④ 세그먼트 방식은 동적 메모리 할당 방법으로 외부 단편화가 발생하지 않는다.
⑤ 세그먼트 방식은 세그먼트 번호와 변위(offset)로 구성하는 가상 주소를 만든다.
해설)
정답 체크 :
(4) 세그먼트 방식은 동적 메모리 할당 방법으로 외부 단편화가 발생한다.
오답 체크 :
(1) 프로그램의 크기가 실제 메모리의 크기보다 커도 가상 메모리를 사용하면 실행이 가능하다.
(2) 페이지 방식은 페이지 테이블을 이용한 페이지 사상 작업으로 인하여 비용이 증가하고 수행속도가 감소한다. 이를 보완하기 위해 페이지 테이블을 주기억장치에 두지 않고 캐시를 사용한다.
(3) 페이지 방식은 페이지를 작게 설계하면 할당을 하고 남는 공간이 적어지므로 내부 단편화를 줄일 수 있지만 단편화가 많이 발생하여 관리의 부담이 커진다.
(5) 세그먼트 방식은 세그먼트 번호와 변위를 가지고, 페이징 방식은 페이지 번호와 변위를 가진다.

4. UNIX 파일 시스템에서 i-node에 존재하는 정보로 옳지 않은 것은?
① 파일 이름
② 파일 저장 위치
③ 파일 생성 일시
④ 파일 소유자
⑤ 파일 접근 권한 비트
해설)
정답 체크 :
(1) 파일 이름은 디렉터리가 가진다. 디렉터리는 이외에도 i-node를 위한 포인터를 가진다.
오답 체크 :
i-node에 존재하는 정보는 다음과 같다.

구분	주요 사항
소유자 식별자	파일을 소유하고 있는 소유자 계정정보
그룹소유자 식별자	파일을 소유하고 있는 소유자의 그룹정보
파일접근 허가권한	파일에 대한 읽기, 쓰기, 실행권한에 대한 정보
파일 내용 담긴 Disk실 주소	파일 내용이 담겨져 있는 정보에 대한 포인터
File Size (byte)	파일 크기
처음 만들어진 시기	파일 처음 생성 시기
마지막 사용된 시기 (접근시간)	파일 마지막 접근시기
마지막 수정된 시기 (수정시간)	파일 마지막 수정시기
File에 대한 링크수	동일 파일에 대한 다양한 이름을 지닌 링크수의 개수
파일종류	디렉토리, 파일인지 (특수파일, 일반파일) 구분

5. 스키마(schema)에 대한 설명으로 옳지 않은 것은?

① 내부 스키마는 범기관적 입장에서 데이터 베이스를 정의한 것이다.
② 개념 스키마는 모든 데이터 개체, 관계, 제약 조건, 접근 권한, 무결성 규칙 등을 명세한다.
③ 개념 스키마는 일반적으로 스키마를 의미한다.
④ 외부 스키마는 사용자나 응용 프로그래머가 접근 할 수 있는 데이터베이스를 정의한다.
⑤ 스키마는 데이터베이스의 논리적 정의, 데이터 구조와 제약 조건에 관한 명세를 기술한 것이다.
해설)
정답 체크 :
(1) 내부 스키마 : 해당 설명은 개념 스키마를 의미하고, 내부 스키마는 전체 데이터베이스가 저장 장치에 실제로 저장되는 방법을 정의한 것이다. 레코드 구조, 필드 크기, 레코드 접근 경로 등 물리적 저장 구조를 정의한다.
오답 체크 :
(2) 개념 스키마 : 전체 데이터베이스에 어떤 데이터가 저장되는지, 데이터들 간에는 어떤 관계가 존재하고 어떤 제약조건이 존재하는지에 대한 정의뿐만 아니라, 데이터에 대한 보안 정책이나 접근 권한에 대한 정의도 포함한다.
(3) 개념 스키마 : 개념 단계에서 데이터베이스 전체의 논리적 구조를 정의한 것이다. 조직 전체의 관점에서 생각하는 데이터베이스의 모습이다.
(4) 외부 스키마 : 외부 단계에서 사용자에게 필요한 데이터베이스를 정의한 것이다. 각 사용자가 생각하는 데이터베이스의 모습, 즉 논리적 구조로 사용자마다 다르다. 서브 스키마(sub schema)라고도 한다.
(5) 스키마 : 데이터베이스에 저장되는 데이터 구조와 제약조건을 정의한 것이다.

6. 다음 SQL 문장 중 구문이 옳은 것은?
① DELETE FROM STUDENT , ENROL WHERE SNO = 100;
② SELECT COUNT (DISTINCT CNO) FROM ENROL WHERE SNO = 100;
③ SELECT SNO , SNAME FROM STUDENT WHERE DEPT = NULL;
④ INSERT STUDENT INTO VALUES (100, '홍길동' , 2, '전산과')
⑤ SELECT DNO , SUM (SNAME) FROM STUDENT GROUP BY DNO WHERE SUM (SNAME) < 100;
해설)
정답 체크 :
(2) 집계함수 COUNT도 정상이고, WHERE 조건도 정상이다.
오답 체크 :
(1) DELETE는 1개의 테이블에 대해서만 동작한다.
(3) DEPT = NULL이 아니라 DEPT is NULL이다.
(4) INSERT STUDENT INTO가 아니라 INSERT INTO STUDENT이다.
(5) GROUP BY DNO WHERE가 아니라 GROUP BY DNO HAVING이다.

7. 다음 정수 리스트를 퀵 정렬 알고리즘으로 오름차순 정렬할 때, 리스트를 처음 분할한 직후의 분할된 두 리스트의 상태로 옳은 것 은? (단, 제어키는 5로 한다.)

(5 2 6 4 7 3 8 1)

① (5 2 6 4), (7 3 8 1)

② (2 4 3 1), (6 7 8)

③ (3 1 2 4), (7 6 8)

④ (3 2 1 4), (7 8 6)

⑤ (1 2 3 4), (6 7 8)

해설)

정답 체크 :

퀵 정렬에서 분할 알고리즘은 다음과 같다.

피벗(pivot)을 가장 왼쪽 숫자라고 가정한다.
두개의 변수 low(피벗 다음의 숫자를 가리킴)와 high(맨 오른쪽을 가리킴)를 사용한다.
low를 오른쪽으로 이동하면서 피벗보다 작으면 통과, 크면 정지한다.
high를 왼쪽으로 이동하면서 피벗보다 크면 통과, 작으면 정지한다.
정지된 위치의 숫자를 교환한다.
low와 high가 교차하면 종료한다.
피벗과 high가 가리키는 숫자를 교환한다.

주어진 조건을 이용해서 문제를 풀면 다음과 같다.

피벗(제어키)을 5라고 가정한다.

low는 6에서 정지하고, high는 1에서 정지한다.

이 둘을 교환한다. : (5 2 1 4 7 3 8 6)

low는 7에서 정지하고, high는 3에서 정지한다.

이 둘을 교환한다. : (5 2 1 4 3 7 8 6)

low(7)와 high(3)가 교차하므로 종료한다.

피벗(5)와 high(3)을 교환한다. : (3 2 1 4 5 7 8 6)

피벗을 기준으로 두 리스트로 분할하면 다음과 같다. : (3 2 1 4) (7 8 6)

8. 다음 C 프로그램에서 main() 함수를 실행할 때 fib() 함수가 호출 되는 횟수로 옳은 것은?

```
#include <stdio.h>
int fib(int n) {
   if (n == 0) return 0;
   if (n == 1) return 1;
   return (fib(n-1) + fib(n-2));
}
int main ()
{
   fib(5);
}
```

① 3번

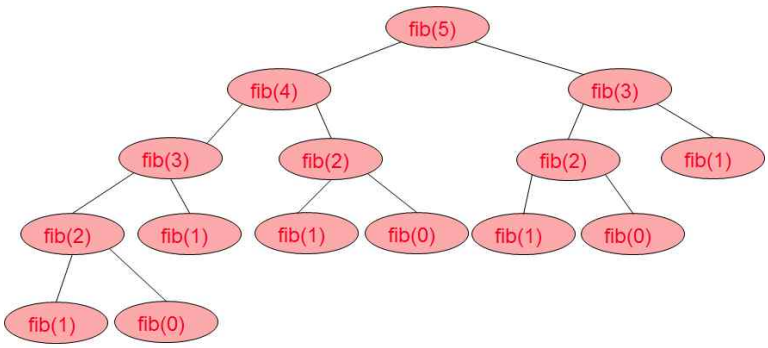
② 5 번

③ 10 번

④ 12 번

⑤ 1 5 번

해설)
정답 체크 :
호출을 그림으로 표현하면 다음과 같다. 그림을 통해 알 수 있듯이 15번의 fib() 함수 호출이 발생함을 알 수 있다.



Tip! : 프로그램의 패턴을 파악하면 기계적으로 함수 호출 횟수를 계산할 수 있다. 부가 설명을 하자면 해당 코드는 피보나치 수열을 순환 호출로 계산했을 때의 단점을 보여준다. 왜냐하면 함수들이 중복해서 호출되기 때문이다. 피보나치 수열의 경우에는 순환 호출보다 반복을 통해서 푸는 것이 좋다.

9. 객체지향 언어에 대한 설명으로 옳지 않은 것은?
① 객체는 재사용이 가능하나 클래스는 불가능하다.
② 객체는 하나의 클래스 타입으로 선언된다.
③ 오버라이딩(overriding), 오버로딩(overloading) 모두 다형성을 지원한다.
④ 추상 메소드는 선언만 하고 그 내용은 기술하지 않은 메소드이다.
⑤ Smalltalk, Simula, Java 언어는 객체지향 개념을 잘 표현하고 있다.
해설)
정답 체크 :
(1) 클래스도 상속을 통해 재사용이 가능하다.
오답 체크 :
(2) 객체는 2개의 클래스 타입으로 선언될 수 없다.
(3) 다형성을 대표하는 것이 오버라이딩과 오버로딩이다.
(4) 추상메소드는 선언만 하고 그 내용을 기술하지 않고, 추상메소드를 가지는 클래스를 추상클래스라고 한다.
(5) Simula, Smalltalk, C++, C#, Java, Object Pascal, Delphi, Python, Perl, Ruby, ASP 등은 객체지향언어이다.

10. 인터넷에서 사용되는 TCP(Transmission Control Protocol)에 대한 설명으로 옳지 않은 것은?
① 데이터를 송수신하기 위해서는 송신측 과 수신측이 서로 미리 연결을 맺어야 한다.
② 수신측의 버퍼가 오버플로우되지 않도록 데이터 흐름 제어(flow control)를 수행한다.
③ 네트워크 내 패킷 수가 과도하게 증가하는 현상을 방지하기 위하여 혼잡 제어(congestion control)를 수행한다.
④ 서버와 클라이언트 간 파일을 전송하기 위한 FTP(File Transfer Protocol)에서 이용한다.

⑤ 송신측 에서는 수신측 이 데이터를 받았는지 확 인할 수 없다.
해설)
정답 체크 :
(5) ACK를 통해 송신측에서는 수신측이 데이터를 받았는지 확인할 수 있다.
오답 체크 :
(1) 연결 설정 과정(3-way handshake)를 수행한다.
(2) 슬라이딩 윈도우를 이용하여 데이터 흐름 제어(end-to-end)를 수행한다.
(3) slow start, fast retransmit, fast recovery 등을 이용해서 혼잡 제어(중간 라우터에서 발생하는 혼잡)를 수행한다.
(4) FTP는 TCP를 이용하고, TFTP는 UDP를 이용한다.

11. 직접 메모리 접근(Direct Memory Access, DMA)에 대한 설명으로 옳지 않은 것은?
① 입출력 장치와 메인 메모리 사이에 데이터를 전송하는 방식이다.
② 사이클 스틸링(cycle stealing)과 같은 의미이다.
③ 각 입출력 장치마다 DMA 제어기(DMA controller)가 한 개 씩 사용된다.
④ CPU가 메인 메모리에 접 하지 않는 동안에 데이터를 전송한다.
⑤ 데이터의 전송이 완료되면 DMA 제어기가 CPU로 인터럽트 신호를 전송한다.
해설)
정답 체크 :
(3) DMA 제어기는 여러 입출력 장치가 공유해서 사용한다.
오답 체크 :
(1) CPU를 대신해서 DMA 제어기가 입출력 장치와 메인메모리 사이에 데이터를 전송한다.
(2) 사이클 스틸링이란 CPU의 간섭 없이 버스 혹은 메인메모리에 접근하는 것을 의미하므로 DMA는 사이클 스틸링이랑 같은 의미이다.
(4) DMA 제어기는 CPU에게 버스 사용권을 얻어야 하므로 CPU가 메인메모리에 접근하지 않는 동안에 데이터를 전송할 수 있다.
(5) 데이터 전송이 완료되면 DMA 제어기가 CPU로 인터럽트를 보내 CPU가 자신에게 시킨 일이 다 끝났음을 알린다.

12. 4개의 512 × 4 비트 RAM을 직렬로 연결하여 구성한 메모리의 마지막 주소를 16진수로 표시한 것으로 옳은 것은?
① 1FF
② 200
③ 7FF
④ 800
⑤ FFF
해설)
정답 체크 :
메모리를 직렬로 연결하면 주소의 개수(워드의 개수)를 늘릴 수 있고, 병렬로 연결하면 워드의 길이를 늘릴 수 있다. 주소의 개수 x 워드의 길이 = (512 x 4) x 4 = 2048 x 4가 된다. 즉, 0번지부터 2047번지까지 사용할 수 있고 마지막 번지인 2047을 16진수로 바꾸면 7FF번지가

된다.

13. 인터넷에서 인접한 노드(node) 간의 프레임을 전송하며 목적지의 주소 지정, 흐름제어 및 오류 제어 기능을 제공 하는 계층으로 옳은 것은?
① 물리 계층(physical layer)
② 데이터 링크 계층(data link layer)
③ 네트워크 계층(network layer)
④ 트랜스포트 계층(transport layer)
⑤ 응용 계층(application layer)
해설)
정답 체크 :
(2) 데이터 링크 : 패킷 노드(Node-to-Node or Peer-to-Peer) 전달, 물리적인(MAC) 주소 기정, 접근 제어(MAC filtering), 흐름 제어(stop-and-wait, sliding window), 오류 처리(ARQ) 등을 수행한다.
오답 체크 :
(1) 물리 : 데이터 링크층으로 부터 한 단위의 데이터를 받아 통신 링크를 따라 전송될 수 있는 형태로 변환한다. 회선 구성, 데이터 전송 모드, 접속형태, 신호, 부호화, 인터페이스, 전송 매체 등을 고려한다.
(3) 네트워크 : 패킷 종단(End-to-End) 전달, 논리적인(IP) 주소 지정, 경로 지정(Routing), 주소 변환(ARP) 등을 수행한다.
(4) 트랜스포트 : 메시지 종단(End-to-End) 전달, 포트 주소 지정, 단편화와 재조립, 연결 제어(관리), 흐름 제어, 혼잡 제어 등을 수행한다.
(5) 응용 : 네트워크 상의 소프트웨어 사용자에게 사용자 인터페이스 제공한다. 전자우편(X.400), 원격파일 접근과 전송(FTAM), 공유 데이터베이스 관리 및 여러 종류의 분산 정보 서비스(X.500) 제공한다.

14. 리눅스와 같은 운영 체제에서 사용되는 프로세스(process)와 스레드(thread)에 대한 설명으로 옳지 않은 것은?
① 프로세스는 서로 다른 수의 스레드를 가질 수 있다.
② 다른 프로세스에 속한 스레드 간의 교착 상태는 발생하지 않는다.
③ 한 프로세스에 속한 스레드 들은 메모리를 공유한다.
④ 스레드 단위로 CPU 스케줄링이 일어난다.
⑤ 같은 프로세스에 속한 스레드로 문맥 교환(context switching)하는 것이 다른 프로세스에 속한 스레드로 문맥 교환하는 것 보다 빠르다.
해설)
정답 체크 :
(2) 다른 프로세스에 속한 스레드 간의 교착상태는 결국 프로세스간 교착상태와 동일하다.
오답 체크 :
(1) 이를 멀티 스레드라고 한다.
(3) 메모리를 공유하고 실행(제어)는 분리한다.
(4) 스레드가 CPU 스케줄링의 가장 작은 단위이다.

(5) 스레드는 메모리를 공유하므로 문맥교환으로 인한 부하가 프로세스간 문맥교환보다 심하지 않다.

15. 현재 모바일 기기에 탑재되는 임베디드 데이터베이스 엔진의 대표적인 예로서, 서버가 따로 필요하지 않아 널리 활용되는 것으로 옳은 것은?
① SQLite
② MySQL
③ mSQL
④ Oracle
⑤ SQL Server
해설)
정답 체크 :
(1) SQLite : 모바일 데이터베이스의 종류에는 SQL Anywhere, DB2 Everyplace, SQL Server Compact, SQL Server Express, Oracle Database Lite, SQLite, SQLBase 등이 존재한다.
오답 체크 :
(2) MySQL : 오픈 소스 RDBMS(관계형 데이터베이스 관리 시스템)로서 서버용 데이터베이스이다.
(3) mSQL : David Hughes가 개발한 SQL 기반의 DBMS(데이터베이스 관리 시스템)으로 서버용 데이터베이스이다.
(4) Oracle : 오라클에서 개발한 RDBMS로 서버용 데이터베이스이다.
(5) SQL Server : 마이크로소프트에서 개발한 RDBMS로 서버용 데이터베이스이다.

16. 뷰 (view)에 대한 설명으로 옳지 않은 것은?
① 뷰는 독자적인 인덱스를 가질 수 없다.
② 뷰의 정의를 변경하기 위해 ALTER 문을 사 용 할 수 있다.
③ 뷰의 정의만 시스템 카탈로그에 저장하였다가 필요시 실행시간에 테이블을 구축한다.
④ 데이터에 대한 보안을 제공한다.
⑤ 뷰는 또 다른 뷰의 정의에 사용될 수 있다.
해설)
정답 체크 :
(2) 뷰의 정의를 변경하기 위해 ALTER VIEW 문을 사용할 수 있다.
오답 체크 :
(1) 인덱스는 물리적인 테이블이 가질 수 있고, 뷰는 물리적인 테이블로 만들어진 논리적인 테이블로 인덱스를 가질 수 없다.
(3) 시스템 카탈로그는 데이터베이스에 저장되는 데이터에 관한 정보를 가지는데, 이중에 뷰의 정의를 포함한다.
(4) 논리적인 테이블인 뷰는 물리적인 테이블의 일부만이 사용자에게 보이므로 이로 인한 보안을 제공한다.
(5) 논리적인 뷰를 이용해서 또 다른 논리적인 뷰를 만들 수 있다.

17. 중위 표기법(infix notation)으로 된 다음 식의 후위 표기법(postfix notation)으로 옳은 것은?

(8 - 2 * 3) / 2 + 6 * 3 / 2

① 8 2 3 * - 2 / 6 3 * 2 / +
② 2 3 * 8 - 2 / 6 3 * 2 / +
③ 8 2 - 3 * 2 / 6 + 3 * 2 /
④ 8 2 3 * - 2 / 6 3 * 2 + /
⑤ + / - 8 * 2 3 2 / * 6 3 2
해설)
정답 체크 :
중위표기식을 후위표기식으로 바꾸는 방법은 다음과 같다.

피연산자를 만나면 그대로 출력한다.
연산자를 만나면 스택에 저장했다가 스택보다 우선 순위가 같거나 낮은 연산자가 나오면 그때 출력한다.
왼쪽 괄호는 우선순위가 가장 낮은 연산자로 취급한다.
오른쪽 괄호가 나오면 스택에서 왼쪽 괄호위에 쌓여있는 모든 연산자를 출력한다.

해당 방법을 기반으로 중위를 후위로 바꾸면 다음과 같다.
(를 스택에 넣는다.
8을 출력한다. (출력 : 8)
(보다 우선순위가 높으므로 -를 스택에 넣는다.
2를 출력한다. (출력 : 8 2)
-보다 우선순위가 높으므로 *를 스택에 넣는다.
3을 출력한다. (출력 : 8 2 3)
)가 나왔으므로 스택에서 (위에 쌓여있는 모든 연산자를 출력한다. (출력 : 8 2 3 * -)
/를 스택에 넣는다.
2를 출력한다. (출력 : 8 2 3 * - 2)
+보다 /의 우선순위가 높으므로 /를 출력하고 +를 스택에 저장한다. (출력 : 8 2 3 * - 2 /)
6을 출력한다. (출력 : 8 2 3 * - 2 / 6)
+보다 우선순위가 높으므로 *를 스택에 넣는다.
3을 출력한다. (출력 : 8 2 3 * - 2 / 6 3)
/와 *의 우선순위가 동일하므로 *를 출력하고 /를 스택에 저장한다. (출력 : 8 2 3 * - 2 / 6 3 *)
2를 출력한다. (출력 : 8 2 3 * - 2 / 6 3 * 2)
수식의 끝이므로 스택에 남아 있는 연산자를 출력한다. (출력 : 8 2 3 * - 2 / 6 3 * 2 / +)

18. 다음 C 프로그램의 출력 결과로 옳은 것은?

```
# include <stdio.h>
int m ain ()
{
  int i = 3;
  int j = 4;
```

```
    if (( ++i > j-- ) && ( i++ < --j )) i = i-- + ++j;
    else j = i-- - --j;
    printf ("%d\n", i);
}
```

① 1
② 2
③ 3
④ 4
⑤ 5
해설)
정답 체크 :
int i = 3;
int j = 4;
++i > j--; // i가 먼저 증가되어 4가 되고, j는 비교 후에 감소한다. 즉, 4 > 4가 되므로 false가 된다. &&(논리 AND)의 특성상 뒤의 조건을 검사할 필요 없다.

19. 네트워크 장비에 대한 설명으로 옳지 않은 것은?
① 허브(hub)는 여러 곳으로부터 들어 온 데이터를 다른 여러 곳으로 보내는 역할을 하는 장비로, 더미 허브(dummy hub)와 스위칭 허브(switching hub)가 있다.
② 리피터(repeater)는 네트워크의 전송거리를 연장하기 위하여 사용하는 장비로, 장거리 전송으로 인해 약해진 신호를 재생하여 전송해 준다.
③ 브리지(bridge)는 동일한 기관의 두 개 이상의 LAN의 분할된 세그먼트를 서로 연결하여 하나의 네트워크로 만드는 장비로, 네트워크에 흐르는 프레임의 물리주소를 필터링 한다.
④ 게이트웨이(gateway)는 다른 네트워크로 들어가는 입구 역할을 하거나 나가는 출구 역할을 하는 장비로, 모뎀 (modem)이 있다.
⑤ 라우터(router)는 LAN, MAN, WAN과 같은 네트워크를 서로 연결해 주는 장비로, 네트워크에 흐르는 패킷의 논리 주소(IP 주소)에 따라 패킷을 라우팅 해준다.
해설)
정답 체크 :
(4) 게이트웨이 : 설명은 맞지만 모뎀은 게이트웨이가 아니다(게이트웨이는 모든 계층의 정보를 이용한다). 모뎀은 아날로그 신호를 디지털 신호로 바꿔주거나 아날로그 신호를 디지털 신호로 바꿔주는 역할을 수행한다.
오답 체크 :
(1) 허브 : 2계층 정보를 이용하는 L2 스위치이다. 더미 허브는 패킷을 받으면 다른 모든 포트들로 브로드캐스팅을 하고, 스위칭 허브는 학습된 포트(해당 MAC이 있는 포트)로 보낸다.
(2) 리피터 : 1계층 정보를 이용하는 L1 스위치이다.
(3) 브리지 : 2계층 정보를 이용하는 L2 스위치이다.
(5) 라우터 : 3계층 정보를 이용하는 L3 스위치이다.
Tip! : 4계층 정보를 이용하는 L4 스위치에는 부하 분산기, 방화벽 등이 있고, 7계층 정보를 이용하는 L7 스위치에는 어플리케이션 스위치, IDS, IPS, DPI 등이 있다.

20. 다음 <보기>의 CPU 스케줄링 기법 중 기아(starvation) 현상이 발생할 수 있는 스케줄링 기법으로 옳은 것을 모두 고르면?

< 보　　기 >
ㄱ . FIFO(First In First Out) 스케줄링
ㄴ . RR (Round Robin) 스케줄링
ㄷ . Priority 스케줄링
ㄹ . HRN(Highest Response ratio Next) 스케줄링
ㅁ . SJF(Shortest Job First) 스케줄링

① ㅁ
② ㄱ , ㄴ
③ ㄷ , ㅁ
④ ㄴ , ㄷ , ㄹ
⑤ ㄷ , ㄹ , ㅁ
해설)
정답 체크 :
(ㄷ) Priority : 우선 순위에 기반해서 처리하므로 우선 순위기 밀리는 프로세스는 기아가 될 수 있다.
(ㅁ) SJF : 실행 시간이 짧은 프로세스를 먼저 처리하므로 실행 시간이 긴 프로세스는 기아가 될 수 있다.
오답 체크 :
(ㄱ) FIFO : 들어온 순서대로 처리되므로 기아가 발생하지 않는다.
(ㄴ) RR : 프로세스가 자신에게 주어진 시간 할당량 만큼만 CPU를 점유하므로 기아가 발생하지 않는다.
(ㄹ) HRN : 우선순위가 대기 시간에 비례해서 증가하므로 기아가 발생하지 않는다.