

# IT CookBook, 쉽게 배우는 데이터 통신과 컴퓨터 네트워크(개정) 4장 연습문제 해답

본 자료의 저작권은 박기현과 한빛아카데미에 있습니다.

이 자료는 강의 보조자료로 제공되는 것으로, 학생들에게 배포되어서는 안 됩니다.

1. ① 교환, ② 전송
2. ① 유니캐스팅, ② 브로드캐스팅, ③ 멀티캐스팅
3. ① 긍정 응답, ② 부정 응답
4. 타임아웃
5. 순서 번호
6. 흐름 제어
7. ① 프레임, ② 체크섬
8. ① 문자, ② 비트
9. ① 비트 스톱핑, ② 0
10. ① 패리티 비트, ② 블록 검사
11. 다항 코드

12. ①, ②, ③

(설명④) 연결된 모든 호스트에 데이터를 전송하는 방식을 브로드캐스팅이라 한다.

(설명⑤) 점대점 방식은 WAN 환경에서 주로 사용하고, 브로드캐스팅 방식은 LAN 환경에서 주로 사용한다.

13. ①, ③, ④

(설명②) 스타형 구조는 하나의 중개 호스트 주위로 여러 호스트를 1:1로 연결하는 형태로 주변 호스트가 데이터를 송수신하려면 반드시 중앙의 중개 호스트를 거쳐야 한다.

(설명⑤) 완전형은 네트워크에 존재하는 모든 호스트가 다른 모든 호스트와 1:1로 직접 연결되는 방식이며, 데이터 송수신시에 교환 기능이 필요 없다.

14. ③, ⑤

(설명③) 링형 구조에서는 전송 데이터가 링 주위를 순환하면서 전송되며, 송신자

가 해당 데이터를 회수하기 때문에 한 바퀴를 순환하여 브로드캐스팅된다.

(설명⑤) 충돌과 관련하여 대표적인 공유 버스 방식인 이더넷에서는 충돌 허용 방식을 사용한다.

15. ①, ②, ③

(설명④) 브로드캐스팅 방식에서는 호스트 수가 많을수록 네트워크 트래픽이 급격히 증가하는 단점이 있다.

(설명⑤) 멀티포인트 유니캐스팅에서는 수신 호스트 수만큼 데이터를 반복 전송해야 하지만, 멀티캐스팅에서는 송신 호스트의 전송 요구 한 번으로 모든 수신 호스트에 데이터를 전달할 수 있다.

16. ①, ③, ④, ⑤

(설명②) 수신 호스트에 데이터가 도착하지 못하는 프레임 분실 오류가 발생하면 타이아웃 기능으로 오류 복구를 시작해야 한다.

17. ②

(설명②) 흐름 제어는 송신 호스트가 수신 호스트보다 아주 빨리 데이터를 전송하는 경우에 필요하다.

18. ①, ③

(설명②) 문자 프레임에서는 각 프레임의 시작 위치에 2 문자(DLE, STX)를 추가하고, 끝나는 위치에 2 문자(DLE, ETX)를 추가해 프레임의 다른 정보와 구분한다.

(설명④) 비트 프레임 방식은 프레임의 시작과 끝 위치에 플래그라는 특수하게 정의된 비트 패턴 (01111110)을 사용해 프레임 단위를 구분한다.

(설명⑤) 수신 호스트가 수신한 데이터 프레임의 내용에서 플래그 패턴 외에는 어떤 경우에도 1이 5개보다 많이 연속하지 않는다.

19. ③, ④

(설명③) 다수의 비트에서 오류가 발생할 때 오류를 검출하는 방법으로는 패리티 방식을 개선한 블록 검사가 있다.

(설명④) 송신 호스트와 수신 호스트는 짝수 패리티나 홀수 패리티 중 동일한 한

가지 방식을 사용해야 한다.

20. ①, ③, ⑤

(설명②) 다항 코드 방식은 버스트 에러 형태의 오류를 검출하는 확률이 높다.

(설명④) 수신 호스트는 전송 오류가 발생했는지를 판단하기 위해 수신한 데이터 전체를 생성 다항식으로 나누는 연산을 하고, 나머지가 0이면 전송 오류가 없는 것으로 판단한다.

21.

컴퓨터 네트워크에는 전송 매체로 연결된 호스트들이 존재한다. 송신 호스트가 수신 호스트에 데이터를 전달(Transfer)하려면 전송과 교환 과정을 거쳐야 한다. **교환(Switching)**은 전달 경로가 둘 이상일 때 라우터에서 데이터를 어느 방향으로 전달할지를 선택하는 기능으로, 다양한 기준에 따라 데이터를 올바른 경로로 전달할 수 있도록 해준다. **전송(Transmission)**은 특정한 물리 매체에 의하여 1:1로 직접 연결된 두 시스템간의 신뢰성 있는 데이터 전송을 보장하기 위한 것이다. 전송에는 라우팅 개념이 포함되지 않는다.

22.

#### ■ 점대점 방식

점대점 방식에서는 호스트 간의 데이터 전달 과정에서 송신 호스트가 중개 호스트와 1:1로 연결되므로 다른 호스트에는 데이터가 전달되지 않는다. 따라서 데이터를 최종 목적지 호스트까지 올바르게 전달하려고 인접 호스트에 전송하는 과정을 단계적으로 반복한다. 이때 각 호스트는 데이터를 송수신하는 호스트일 수도 있고, 중간에 있는 교환 호스트일 수도 있다. 원거리에 있는 시스템 사이의 통신 방식으로, WAN 환경에서 주로 사용한다.

#### ■ 브로드캐스팅 방식

브로드캐스팅 방식에서는 공유 전송 매체 하나에 여러 호스트를 연결하기 때문에 네트워크에 연결된 모든 호스트에 데이터가 전송된다. 데이터 전달 과정에는 별도의 교환 기능이 필요 없다. 대신, 자신을 목적지로 하지 않은 데이터를 받은 호스트는 받은 데이터를 버려 결국 하나의 목적지 호스트만 데이터를 수신하도록 설계해야 한다. LAN처럼 지리적으로 가까운 호스트 사이의 통신에서 주로 사용한다.

23.

송신 호스트를 기준으로, 수신 호스트 하나와 연결되면 유니포인트(Unipoint)고, 다수의 수신 호스트와 연결되면 멀티포인트(Multipoint)가 된다. 송신 호스트가 한 번의 전송으로 수신 호스트 하나에만 데이터를 전송할 수 있으면 유니캐스팅(Unicasting)이고, 다수의 수신호스트에 전송할 수 있으면 멀티캐스팅(Multicasting)이다.

24.

#### ■ 응답 프레임

송신 호스트가 전송한 데이터 프레임의 일부가 깨지는 프레임 변형 오류를 확인한 수신 호스트는 송신 호스트에게 응답 프레임을 전송해 원래의 데이터 프레임을 재전송하도록 요구할 수 있다. 수신 호스트가 전송하는 응답 프레임의 종류에는 데이터 프레임이 정상적으로 도착했을 때 회신하는 긍정 응답 프레임과 데이터 프레임이 깨졌을 때 회신하는 부정 응답 프레임이 있다. 송신 호스트의 재전송 기능은 수신 호스트의 부정 응답 프레임 회신에 의해 이루어진다.

#### ■ 타이아웃

송신 호스트가 전송한 데이터 프레임이 수신 호스트에게 도착하지 못하는 프레임 분실 오류가 발생하면 수신 호스트는 이 사실을 인지할 수 없다. 따라서 오류의 복구 과정이 송신 호스트의 주도로 이루어져야 한다. 송신 호스트는 데이터 프레임을 전송한 후에 일정 시간 이내에 수신 호스트로부터 긍정 응답 프레임 회신이 없으면 타임아웃(Timeout)기능을 동작시켜 데이터 프레임을 재전송한다.

#### ■ 순서 번호

수신 호스트의 긍정 응답 프레임을 분실하면 데이터 프레임이 제대로 도착했음에도 불구하고 송신 호스트가 이를 인지할 수 없다. 따라서 송신 호스트가 타임아웃 기능에 의해 원래 데이터를 재전송함으로써 수신 호스트가 데이터 프레임을 중복 수신하는 결과를 초래한다. 이럴 때 수신 호스트가 중복 데이터 프레임을 가려내려면 각 프레임 내부에 순서 번호(Sequence Number)를 기록해야 한다.

25.

데이터 프레임이 전송되었을 때 발생 가능한 오류의 유형에는 크게 데이터 변형과 데이터 분실로 나눌 수 있다.

#### ■ 프레임 변형

프레임 변형은 데이터 프레임이 수신 호스트에 도착했으나, 전송 과정에서 프레임의 내용이 변형되는 오류가 발생한 경우다. 프레임 변형 오류를 인지한 수신 호스트는 송신 호스트에게 부정 응답 프레임을 전송함으로써, 원래의 데이터 프레임을 재전송하는 오류 복구 과정이 진행된다. 앞서 언급한 것처럼 부정 응답 프레임을 사용하지 않는 프로토콜에서는 송신 호스트의 타임아웃 기능에 따라 복구 과정을 시작한다.

재전송된 데이터 프레임은 올바르게 전송될 수도 있지만, 변형 오류가 발생할 수도 있다. 또한 긍정 응답 프레임이나 부정 응답 프레임도 전송 과정에서 변형이나 분실과 같은 오

류가 발생할 수 있다. 따라서 데이터 링크 계층 프로토콜에서 다루는 전송 오류 문제의 원리는 매우 단순하지만, 프로토콜 설계 시 세심한 주의가 필요하다.

#### ■ 프레임 분실

데이터 링크 계층의 주요 기능 중 하나는 프레임을 전송한 송신 호스트가 동작하는 타임아웃기능이다. 네트워크에서는 송신 호스트가 전송한 데이터 프레임이 전송 과정에서 사라지는 프레임 분실 오류가 발생할 수 있다. 수신 호스트는 송신 호스트로부터 어떠한 데이터 프레임도 전달받지 못했기 때문에 긍정 응답이나 부정 응답 프레임을 회신할 수 없다. 결과적으로 송신 호스트도 응답 프레임을 회신 받을 수 없어 응답 프레임을 무작정 기다려야 한다. 따라서 오류 복구는 송신 호스트 주도로 타임아웃(Timeout) 기능에 따라 처리된다. 즉, 송신 호스트는 데이터 프레임을 전송한 후에 특정 시간까지 수신 호스트의 긍정 응답 프레임이 도착하지 않으면 타임아웃 기능에 따라 원래의 프레임을 스스로 재전송한다.

26.

#### ■ 순서 번호가 있을 경우와 없을 경우

올바르게 수신한 데이터 프레임에 대한 긍정 응답 프레임이 사라지는 오류가 발생하면 송신 호스트의 타임아웃 기능에 따라 재전송 과정이 진행된다. 재전송된 데이터 프레임이 제대로 수신되면 수신 호스트 입장에서 동일한 프레임을 중복해 수신하는 결과를 초래한다. 송신 호스트 입장에서 보면 자신이 동일한 데이터 프레임을 두 번 전송했는지, 아니면 서로 다른 두 개의 데이터 프레임을 연속 전송했는지 구분할 수 있다. 그러나 수신 호스트는 두 경우를 구분할 방법이 없다.

즉, 순서 번호가 있을 경우 수신 호스트는 순서 번호에 근거하여 동일한 데이터 프레임이 중복 도착했는지, 아니면 서로 다른 데이터 프레임이 도착했는지를 구분할 수 있다. 하지만 순서 번호가 없을 경우 수신 호스트는 동일한 데이터 프레임이 중복 수신하였는지 확인할 수 없다.

[추가 설명]

#### ■ 순서 번호

데이터 링크 계층의 오류 복구 기능이 수행되는 과정에서 동일한 데이터 프레임이 수신 호스트에 중복해 도착할 수 있다. 따라서 오류 없이 수신된 중복 데이터의 문제에 대비해야 한다. 중복 데이터 처리는 프레임 내부에 각 프레임의 고유 번호인 순서 번호(Sequence Number)를 기록하여 해결한다.

27.

문자 스템핑(Character Stuffing)은 문자 프레임의 전송 데이터 중 DLE 문자가 포함되면서

발생하는 혼란을 예방하는 방법이다. [그림]처럼 송신 호스트가 전송하는 데이터를 미리 변형함으로써 혼선의 여지를 없앨 수 있다. 즉, 전송 데이터가 DLE 문자를 포함하면 DLE 문자 다음에 DLE 문자를 강제로 추가한다.

수신 호스트는 프레임 내용에 DLE 문자가 연속해서 두 번 나타나면 두 번째 DLE는 송신 호스트가 임의로 추가한 문자라고 판단할 수 있다. 따라서 상위 계층인 네트워크 계층에 데이터를 전달하기 전에 둘 중 하나를 제거해야 한다. [그림 (b)]는 이를 설명하고 있으며, 송신 호스트가 최초에 전송한 프레임인 [그림 (b)]와 동일한 결과가 된다. 이와 같이 문자 프레임의 전송 과정에서 제어 문자를 추가하는 기능을 문자 스텀핑이라고 한다.

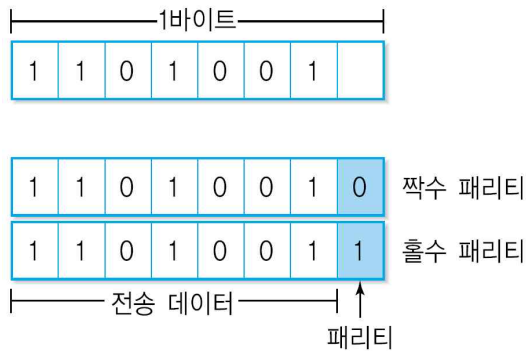


28.

비트 프레임 방식에서는 송신 호스트가 전송하고자 하는 데이터 내용 중에 값이 1인 패턴이 연속해서 5번 발생하면 강제로 0을 추가해 전송한다. 플래그에 1이 연속해서 6개로 원천적으로 데이터 내용에 플래그 패턴의 발생을 차단하기 위함이다. 수신 호스트가 수신한 데이터 프레임의 내용에서 플래그 패턴 외에는 어떤 경우에도 1이 5개보다 많이 연속하지 않는다. 다시 말해서 플래그 패턴과 동일한 형태의 패턴이 데이터 링크 계층의 전송 데이터에는 발생할 수 없다. 이와 같은 기능을 비트 스텀핑(Bit Stuffing)이라 하며, 수신 호스트는 송신 과정에서 추가된 0을 제거하여 원래의 데이터를 상위 계층에 전달한다.

29.

패리티 비트는 전송 과정에서 1비트 오류를 검출하기 위한 것으로, 패리티 비트를 포함해 1의 개수가 짝수나 홀수 개가 되도록 한다. 1바이트(8비트) 구조에서 패리티(Parity) 비트는 7비트 크기의 ASCII 코드를 제외한 나머지 1비트다.



### 패리티 비트

예를 들어, 그림과 같이 1101001이라는 데이터를 짝수 패리티(Even Parity) 방식을 사용해 전송하려면 11010010의 형태로 만들어 전송한다. 즉, 짝수 패리티 방식에서는 데이터 끝에 0이라는 패리티 비트를 추가해 전체 1의 개수를 짝수로 만들어 준다. 데이터 전송 과정에서 1비트 오류가 발생하면 1의 개수가 홀수 개로 바뀐다. 예를 들어, 위의 데이터 전송 과정에서 세 번째 비트가 0에서 1로 바뀌면 수신 호스트는 11110010을 받는다. 수신 호스트는 패리티 검사를 하여 1의 개수가 홀수로 변경된 사실을 알게 되어 데이터 전송 과정에서 1비트 오류가 발생했음을 확인할 수 있다.

홀수 패리티(Odd Parity) 방식은 짝수 패리티와 반대로, 1의 개수를 홀수로 만드는 것이다. 송신 호스트와 수신 호스트는 짝수 패리티나 홀수 패리티 중 동일한 한 가지 방식을 사용해야 한다.

30.

생성 다항식  $G(x) = x^6 + x^3 + x + 1$  이 되고, 전송 데이터가 101101001인 경우의 체크섬 계산 과정은 다음과 같다.

$$\begin{array}{r}
 \phantom{1\ 0\ 0\ 1\ 0\ 1\ 1\ }1\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 1 \\
 1\ 0\ 0\ 1\ 0\ 1\ 1\ \overline{) 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0} \\
 \underline{1\ 0\ 0\ 1\ 0\ 1\ 1\phantom{0}} \\
 0\ 1\ 0\ 0\ 0\ 1\ 0\phantom{0} \\
 \underline{0\ 0\ 0\ 0\ 0\ 0\ 0} \\
 1\ 0\ 0\ 0\ 1\ 0\ 1\phantom{0} \\
 \underline{1\ 0\ 0\ 1\ 0\ 1\ 1\phantom{0}} \\
 0\ 0\ 1\ 1\ 1\ 0\ 0\phantom{0} \\
 \underline{0\ 0\ 0\ 0\ 0\ 0\ 0} \\
 0\ 1\ 1\ 1\ 0\ 0\ 0\phantom{0} \\
 \underline{0\ 0\ 0\ 0\ 0\ 0\ 0} \\
 1\ 1\ 1\ 0\ 0\ 0\ 0\phantom{0} \\
 \underline{1\ 0\ 0\ 1\ 0\ 1\ 1\phantom{0}} \\
 1\ 1\ 1\ 0\ 1\ 1\ 0\phantom{0} \\
 \underline{1\ 0\ 0\ 1\ 0\ 1\ 1\phantom{0}} \\
 1\ 1\ 1\ 1\ 0\ 1\ 0\phantom{0} \\
 \underline{1\ 0\ 0\ 1\ 0\ 1\ 1\phantom{0}} \\
 1\ 1\ 0\ 0\ 0\ 1\ 0\phantom{0} \\
 \underline{1\ 0\ 0\ 1\ 0\ 1\ 1\phantom{0}} \\
 1\ 0\ 1\ 0\ 0\ 1\phantom{0}
 \end{array}$$

계산을 통해 얻은 나머지(체크섬)는 101001이므로, 송신데이터는 '101101001101001'이 된다. 수신호스트는 수신 데이터 '101101001101001'을 생성다항식 '1001011'로 나누기 연산을 수행한다. 이때 나머지가 0이면 전송 오류가 없고, 0이 아니면 오류가 발생한 것으로 판단할 수 있다.