◆ 11년 4월 고3 48~50번

[48 ~ 50] 다음 글을 읽고 물음에 답하시오.

프로토콜이란 통신 상황에서 송신자의 정보를 정확하고 안정적으로 수신자에게 전송하기 위해 국제적으로 표준화한 약속을 의미한다. 이러한 프로토콜이 실제 통신 상황에서 적용되기 위해서는 표준화된 논리적 구조가 필요한데, 그 중 가장 대표적인 것이 통신 기능을 일곱 단계로 분할한 OSI 7계층 모델이다. 이때 송신자가 보내고자 하는 정보는 송신의 7단계부터 1단계까지의 과정을 거치며 발송되고 수신의 1단계부터 7단계까지의 과정을 통해 수신자에게 전달된다.

이 모델의 7단계에서 5단계까지는 정보가 소프트웨어적인 측면에서 다루어지고 있다는 공통점이 있는데, 이를 상위 계층이라고 한다. 최상위 단계인 7단계를 응용 계층이라고 하는데, 송신자나 소프트웨어가 네트워크에 접근하는 단계이다. 예를 들면 송신자가 사이트에 접속하여 로그-인하는 것 등이 이 단계에 해당된다. 6단계인 표현 계층은 보내고자 하는 정보를 다른 컴퓨터와의 호환이 가능하도록 문자열, 숫자 등의 컴퓨터 표준 형식으로 변환하는 단계이다. 또한 송신 과정에서 정보를 압축하고 수신 과정에서 압축을 푸는 단계이기도 한다. 5단계인 세션 계층은 보내려는 정보에 검사점을 추가하여 오류 발생 시 재전송하기 위한 일종의 기준점을 제시하는 단계이다.

이와는 달리 4단계에서 1단계까지를 하위 계층이라고 하는데, 정보가 전달될 수 있도록 물리적인 측면에서 기능한다는 공통점이 있다. 이 중 4단계인 전송 계층은 정보를 '프레임(frame)'이라는 단위로 분할한 후 각 프레임에 수신자의 인터넷 주소를 입력하여 정보가 정확하게 전달되도록 하는 기능을 한다. 또한 3단계인 네트워크 링크 계층에서는 보내고자 하는 정보를 수신자에게 가장 빠르고 안전하게 전송하기 위한 최적의 경로를 설정해주는 기능을 한다. 경로가 설정되면 2단계인 데이터 링크 계층에서는 프레임 단위로 변환된 정보를 물리적으로 전송이 가능한 2진수 0과 1로 표시되는 비트(bit)로 변환하여 이를 물리 계층으로 보낸다. 마지막으로 물리 계층은 정보를 보내기 위한 케이블의 종류나 전기 신호 등의 기계적 조건을 점검하여 이상이 없으면 케이블을 통해 정보를 발송한다.

이러한 정보의 송·수신 과정에서 케이블로 연결되어 있는 물리계층을 제외하면 송신의 n계층과 수신의 n계층이 직접 연결되어 있지 않다. 하지만 정보를 정확히 전달하기 위해 송신과 수신의 n계층끼리는 해당 계층의 프로토콜을 중심으로 기능상 상호 작용해야 하므로 송신의 n계층마다 해당 계층의 프로토콜을 정보에 덧붙여 보내야 한다. 이처럼 각 계층에서 추가된 프로토콜들은 수신자의 해당 계층에서 해석된 후, 즉시 삭제되고 수신자의 7단계에서는 받고자 하는 정보만이 남게 된다.

48. 위 글의 표제와 부제로 적절한 것은? [1점]

① OSI 7계층 모델의 필요성
— 정보의 압축 과정을 중심으로

② OSI 7계층 모델의 장점과 단점
— 프로토콜의 변화 양상을 중심으로

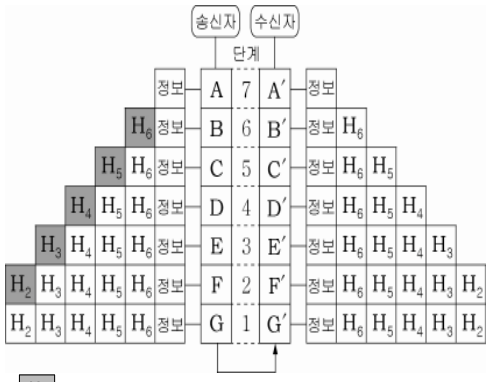
③ 프로토콜을 이용한 정보 전달
— OSI 7계층 모델의 단계를 중심으로

④ 프레임의 분할과 결합 과정
— OSI 7계층 모델의 발전 과정을 중심으로

⑤ 통신의 과정에서 프로토콜의 전달 방식
— 전송 과정상 오류의 해결방법을 중심으로

※ <보기>는 OSI 7계층 모델을 간략히 도식화한 것이다. 위 글과 <보기>를 바탕으로 49번과 50번의 두 물음에 답하시오.

〈 보 기 〉



* Hn 부분은 n 단계에서 새롭게 추가된 프로토콜을 의미함.
(단, 7단계 및 1단계에서는 프로토콜을 추가하지 않음.)

49. <보기>의 단계에서 이루어지는 작업으로 적절하지 <u>않은</u> 것은?

① B : 정보를 호환 가능한 형식으로 변환한다.
② D : 정보를 프레임 단위로 분할한다.
③ E : 정보를 전송하기 위한 최적의 경로를 설정한다.
④ F : 보내고자 하는 정보를 비트 단위로 변환한다.
⑤ G : 정보에 검사점을 추가하여 오류에 대비한다.

50. 위 글을 읽고 난 후 <보기>에 대해 보일 반응으로 적절하지 <u>않은</u> 것은?

① B에서 정보를 압축했다면 B′에서는 압축이 풀리겠군.
② C의 정보 속에는 수신자의 인터넷 주소가 포함되어 있겠군.
③ D에서 추가된 H_4는 D′에서 해석된 후 삭제되겠군.
④ E와 E′는 동일한 프로토콜로 연계되어 상호 작용하겠군.
⑤ D′~G′까지는 물리적 측면에서 기능하는 단계로군.

- 출전: 베루즈 A. 포루잔 저. 《TCP/IP 프로토콜》
- 정답: 48.③ 49.⑤ 50.②

◆ 15년 6월 고1 24~26번

[24~26] 다음 글을 읽고 물음에 답하시오.

일상에서 편지를 보낼 때는 편지 한 통이 통째로 전달된다. 그러나 네트워크상에서의 이메일(e-mail)은 그 내용이 조각조각으로 나뉘어 전송된다. 이렇게 나뉜 조각이 수신자에게 전송된 후 재결합되어 수신자는 한 통의 이메일을 받아볼 수 있다. 이러한 정보 전달 방식을 패킷 교환 방식이라 한다.

'패킷'이란 네트워크상에서 정보를 보낼 때 전송하기 쉽도록 데이터를 작은 단위로 나누어 놓은 것을 말한다. 패킷은 크게 헤더부와 데이터 영역으로 구분된다. 헤더부에는 메시지가 최종적으로 전달될 주소와 패킷의 일련번호 등의 정보가 들어있고, 데이터 영역에는 메시지 자체의 내용이 들어있다.

패킷 교환은 다음과 같은 순서로 진행된다. 먼저 긴 메시지는 여러 개의 패킷으로 나뉘고 각 패킷에는 헤더가 부착된다. 각각의 패킷은 버퍼와 여러 개의 노드로 이루어진 '패킷 교환망'을 지나게 된다. 패킷이 한꺼번에 많이 나가면 경로가 막힐 수도 있기 때문에 패킷들은 우선 '버퍼'라는 기억 장치에 잠시 저장된다. 버퍼는 패킷이 원활하게 전송될 수 있도록 먼저 도착한 패킷을 보내고 나머지 패킷들을 잠시 저장해 둔다. 이후 각각의 패킷들은 '노드'라고 불리는 여러 개의 통신 지점을 지나간다. 노드 하나에도 여러 개의 경로가 연결되어 있어서 패킷들은 서로 흩어져 여러 개의 노드와 경로를 통해 이동하게 된다. 패킷 교환망을 지나온 각 패킷들은 수신지에 일련번호의 순서와 상관없이 개별적으로 도착한다. 수신지에 모두 도착하면 패킷들은 일련번호의 순서에 맞게 원래의 메시지로 재결합된다. 만약 수신지에서 일련번호 순서대로 재결합이 되지 못했거나 패킷이 모두 전송되지 못했을 경우 '발신 후 수신 불능'이나 '수신 후 에러 메시지'를 받을 수도 있다.

패킷 교환 방식은 작은 단위로 나눠진 패킷들이 여러 개의 노드를 통해서 서로 다른 경로로 전송된 후 나중에 합쳐지기 때문에 기존의 정보 전송 방식에 비해 많은 양의 데이터를 빠르게 전송할 수 있다. 패킷들이 각기 다른 경로로 전송되기 때문에 데이터 전송 시 하나의 경로에 과부하가 발생하여 전송이 지연되더라도 다른 경로를 통해 패킷을 전송할 수 있다는 장점이 있다. 이 방식을 활용하면 패킷들을 기기의 처리 속도에 맞추어 전송할 수 있어서 처리 속도가 다른 기기들 간에도 정보 전송이 가능하다. 또한 보내야 할 데이터가 큰 경우에도 패킷으로 나뉘어 전송되므로 정보를 원활하게 전송할 수 있다.

24. 윗글의 표제와 부제로 가장 적절한 것은?

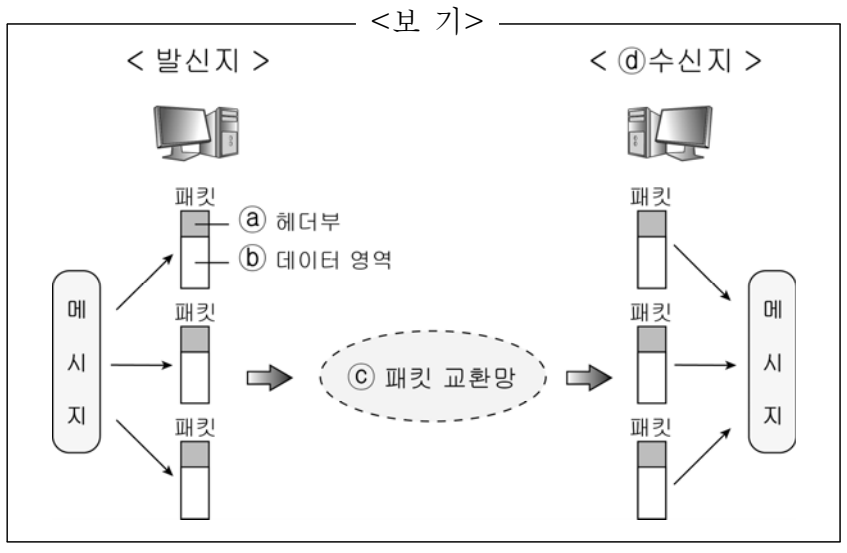
① **이메일 전송의 원리**
– 이메일과 일반 우편 전송 방식의 차이점을 중심으로
② **패킷의 구조와 생성 원리**
– 헤더부와 데이터 영역의 역할과 특징을 중심으로
③ **네트워크상에서의 정보 생성 방법**
– 패킷 교환 방식의 장점과 단점을 중심으로
④ **네트워크상에서의 정보 전송 원리**
– 패킷 교환 방식에서의 데이터 전송 원리를 중심으로
⑤ **정보 전달의 속도를 높여주는 패킷 교환 방식**
– 정보 전송의 역사적 발전 양상을 중심으로

25. 윗글의 내용과 일치하지 <u>않는</u> 것은?

① 네트워크상에서의 이메일은 그 내용이 여러 개의 조각으로 나뉘어 전송된다.
② 패킷은 네트워크상에서 전송하기 쉽도록 데이터를 작은 단위로 나누어 놓은 것을 의미한다.
③ 패킷 교환 방식은 정보 처리 속도가 다른 기기 사이에 정보 전송이 불가능하다는 단점이 있다.
④ 패킷 교환 방식에서는 하나의 경로에 과부하가 발생하더라도 다른 경로를 통해 패킷을 전송할 수 있다.
⑤ 패킷 교환 방식은 기존의 정보 전송 방식에 비해 많은 양의 데이터를 빠르게 전송할 수 있다는 장점이 있다.

26. <보기>는 패킷 교환 방식을 그림으로 표현한 것이다. ⓐ~ⓓ에 대한 설명으로 적절하지 <u>않은</u> 것은? [3점]

<보 기>



① ⓐ: 패킷이 최종적으로 전달되어야 할 주소와 패킷의 일련번호에 대한 정보가 포함되어 있다.
② ⓑ: 전달하고자 하는 메시지의 내용이 포함되어 있다.
③ ⓒ: 패킷이 원활하게 전송될 수 있도록 패킷을 잠시 저장해 두는 장치가 있다.
④ ⓒ: 패킷들이 이곳을 통과할 때는 여러 개의 노드와 경로를 거쳐 이동한다.
⑤ ⓓ: 패킷들이 이곳에 일련번호의 순서대로 도착하지 않았을 경우 '발신 후 수신 불능' 메시지를 받을 수 있다.

- 출전: 진혜진 저. 《네트워크 개론》
- 정답: 24.④ 25.③ 26.⑤

[30～34] 다음 글을 읽고 물음에 답하시오.

DNS(도메인 네임 시스템) 스푸핑은 인터넷 사용자가 어떤 사이트에 접속하려 할 때 사용자를 위조 사이트로 접속시키는 행위를 말한다. 이는 도메인 네임을 IP 주소로 변환해 주는 과정에서 이루어진다.

인터넷에 연결된 컴퓨터들이 서로를 식별하고 통신하기 위해서 각 컴퓨터들은 IP(인터넷 프로토콜)에 따라 ㉠ 만들어지는 고유 IP 주소를 가져야 한다. 프로토콜은 컴퓨터들이 연결되어 서로 데이터를 주고받기 위해 사용하는 통신 규약으로 소프트웨어나 하드웨어로 구현된다. 현재 주로 사용하는 IP 주소는 '***.126.63.1'처럼 점으로 구분된 4개의 필드에 숫자를 사용하여 ㉡ 나타낸다. 이 주소를 중복 지정하거나 임의로 지정해서는 안 되고 공인 IP 주소를 부여받아야 한다.

공인 IP 주소에는 동일한 번호를 지속적으로 사용하는 고정 IP 주소와 번호가 변경되기도 하는 유동 IP 주소가 있다. 유동 IP 주소는 DHCP라는 프로토콜에 의해 부여된다. DHCP는 IP 주소가 필요한 컴퓨터의 요청을 받아 주소를 할당해 주고, 컴퓨터가 IP 주소를 사용하지 않으면 주소를 반환받아 다른 컴퓨터가 그 주소를 사용할 수 있도록 해 준다. 한편, 인터넷에 직접 접속은 안 되고 내부 네트워크에서만 서로를 식별할 수 있는 사설 IP 주소도 있다.

인터넷은 공인 IP 주소를 기반으로 동작하지만 우리가 인터넷을 사용할 때는 IP 주소 대신 사용하기 쉽게 'www.***.***' 등과 같이 문자로 ㉢ 이루어진 도메인 네임을 이용한다. 따라서 도메인 네임을 IP 주소로 변환해 주는 DNS가 필요하며 DNS를 운영하는 장치를 네임서버라고 한다. 컴퓨터에는 네임서버의 IP 주소가 기록되어 있어야 하는데, 유동 IP 주소를 할당받는 컴퓨터에는 IP 주소를 받을 때 네임서버의 IP 주소가 자동으로 기록되지만, 고정 IP 주소를 사용하는 컴퓨터에는 사용자가 네임서버의 IP 주소를 직접 기록해 놓아야 한다. 인터넷 통신사는 가입자들이 공동으로 사용할 수 있는 네임서버를 운영하고 있다.

㉮ 사용자가 어떤 사이트에 정상적으로 접속하는 과정을 살펴보자. 웹 사이트에 접속하려고 하는 컴퓨터를 클라이언트라 한다. 사용자가 방문하고자 하는 사이트의 도메인 네임을 주소창에 직접 입력하거나 포털 사이트에서 그 사이트를 검색해 클릭하면 클라이언트는 기록되어 있는 네임서버에 도메인 네임에 해당하는 IP 주소를 물어보는 질의 패킷을 보낸다. 네임서버는 해당 IP 주소가 자신의 목록에 있으면 클라이언트에 이 IP 주소를 알려 주는 응답 패킷을 보낸다. 응답 패킷에는 어느 질의 패킷에 대한 응답인지가 적혀 있다. 만일 해당 IP 주소가 목록에 없으면 네임서버는 다른 네임서버의 IP 주소를 알려 주는 응답 패킷을 보내고, 클라이언트는 다시 그 네임서버에 질의 패킷을 보내는 단계로 돌아가 같은 과정을 반복한다. 클라이언트는 이렇게 ㉣ 알아낸 IP 주소로 사이트를 찾아간다. 네임서버와 클라이언트는 UDP라는 프로토콜에 ㉤ 맞추어 패킷을 주고받는다. UDP는 패킷의 빠른 전송 속도를 확보하기 위해 상대에게 패킷을 보내기만 할 뿐 도착 여부는 확인하지 않으며, 특정 질의 패킷에 대해 처음 도착한 응답 패킷을 신뢰하고 다음에 도착한 패킷은 확인하지 않고 버린다. DNS 스푸핑은 UDP의 이런 허점들을 이용한다.

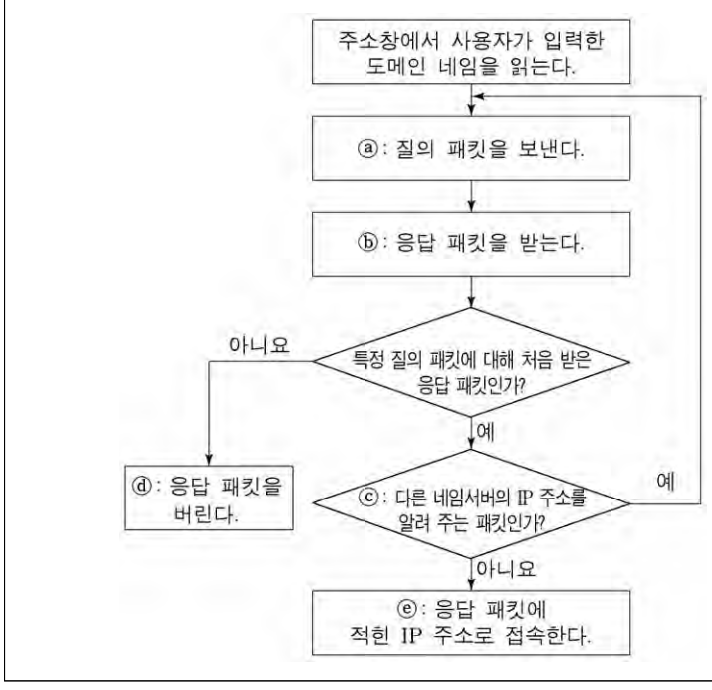
㉯ DNS 스푸핑이 이루어지는 과정을 알아보자. 악성 코드에 감염되어 DNS 스푸핑을 행하는 컴퓨터를 공격자라 한다. 클라이언트가 네임서버에 특정 IP 주소를 묻는 질의 패킷을 보낼 때, 공격자에도 패킷이 전달되고 공격자는 위조 사이트의 IP 주소가 적힌 응답 패킷을 클라이언트에 보낸다. 공격자가 보낸 응답 패킷이 네임서버가 보낸 응답 패킷보다 클라이언트에 먼저 도착하고 클라이언트는 공격자가 보낸 응답 패킷을 옳은 패킷으로 인식하여 위조 사이트로 연결된다.

30. 윗글의 '프로토콜'에 대한 설명으로 적절하지 않은 것은?

① 컴퓨터 사이의 통신을 위한 규약으로서 저마다 정해진 기능이 있다.
② IP에 따르면 현재 주로 사용하는 IP 주소는 4개의 필드에 적힌 숫자로 구성된다.
③ DHCP를 이용하는 컴퓨터는 IP 주소를 요청해야 IP 주소를 부여받을 수 있다.
④ DHCP를 이용하는 컴퓨터에는 네임서버의 IP 주소를 사용자가 기록해야 한다.
⑤ UDP는 패킷 전송 속도를 높이기 위해 패킷이 목적지에 제대로 도착했는지 확인하지 않는다.

31. <보기>는 ㉮ 또는 ㉯에서 이루어지는 클라이언트의 동작을 나타낸 것이다. 이에 대한 이해로 적절한 것은? [3점]

<보 기>



① ㉮: ⓐ가 두 번 동작했다면, 두 질의 내용이 동일하고 패킷을 받는 수신 측도 동일하다.
② ㉮: ⓑ가 두 번 동작했다면, 두 응답 내용이 서로 다르고 패킷을 보낸 송신 측은 동일하다.
③ ㉮: ⓒ는 ⓐ에서 질의한 도메인 네임에 해당하는 IP 주소를 네임서버가 찾았는지 여부를 확인하는 절차이다.
④ ㉯: ⓓ의 응답 패킷에는 공격자가 보내 온 IP 주소가 포함되어 있다.
⑤ ㉯: ⓔ의 IP 주소는 ⓐ에서 질의한 도메인 네임에 해당하는 IP 주소이다.

32. 윗글을 바탕으로 알 수 있는 것은?

① DNS는 도메인 네임을 사설 IP 주소로 변환한다.
② 동일한 내부 네트워크에 연결된 컴퓨터들의 사설 IP 주소는 서로 달라야 한다.
③ 유동 IP 주소 방식의 컴퓨터들에는 동시에 동일한 공인 IP 주소를 할당할 수 있다.
④ 고정 IP 주소 방식의 컴퓨터들에는 동시에 동일한 공인 IP 주소를 부여할 수 있다.
⑤ IP 주소가 서로 다른 컴퓨터들은 각각에 기록되어 있는 네임서버의 IP 주소도 서로 달라야 한다.

33. 윗글과 <보기>를 참고할 때, DNS 스푸핑을 피하기 위한 방법으로 적절한 것은?

<보 기>

DNS가 고안되기 전에는 특정 컴퓨터의 사용자가 'hosts'라는 파일에 모든 도메인 네임과 그에 해당하는 IP 주소를 적어 놓았고, 클라이언트들은 이 파일을 복사하여 사용하였다. 네임서버를 사용하는 현재에도 여전히 클라이언트는 질의 패킷을 보내기 전에 hosts 파일의 내용을 확인한다. 클라이언트가 이 파일에서 원하는 도메인 네임의 IP 주소를 찾으면 그 주소로 바로 접속하고, IP 주소를 찾지 못했을 때 클라이언트는 네임서버에 질의 패킷을 보낸다.

① 클라이언트에서 사용자가 hosts 파일을 찾아 삭제하면 되겠군.
② 클라이언트의 IP 주소를 사용자가 클라이언트의 hosts 파일에 적어 놓으면 되겠군.
③ 클라이언트에 hosts 파일이 없더라도 사용자가 주소창에 도메인 네임만 입력하면 되겠군.
④ 네임서버의 도메인 네임과 IP 주소를 사용자가 클라이언트의 hosts 파일에 적어 놓으면 되겠군.
⑤ 접속하려는 사이트의 도메인 네임과 IP 주소를 사용자가 클라이언트의 hosts 파일에 적어 놓으면 되겠군.

34. 문맥상 ㉠~㉤과 바꿔 쓰기에 가장 적절한 것은?

① ㉠: 제조(製造)되는
② ㉡: 표시(標示)한다
③ ㉢: 발생(發生)된
④ ㉣: 인정(認定)한
⑤ ㉤: 비교(比較)해

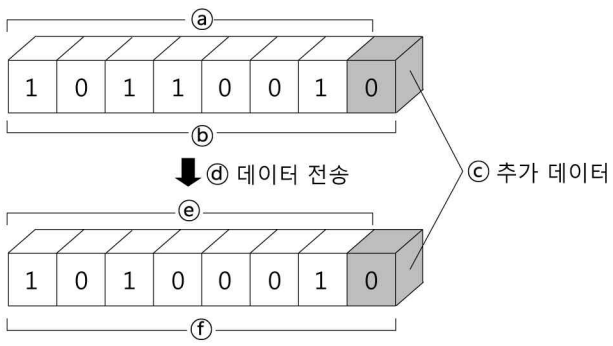
- 정답: 30.④ 31.③ 32.② 33.⑤ 34.②

◆ 14년 7월 고3 A형 28~30번

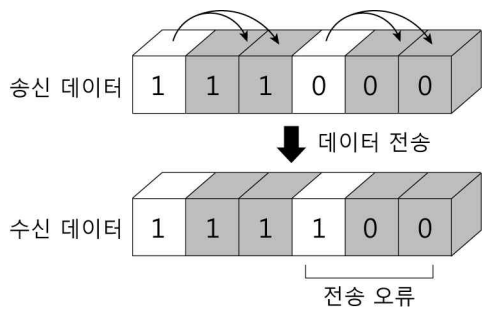
[28~30] 다음 글을 읽고 물음에 답하시오.

무선 통신 시스템에서 전파로 전송되는 신호는 때에 따라 왜곡될 수도 있고, 안테나나 통신 장비에서 발생하는 다양한 열잡음*으로 원하지 않는 신호가 더해질 수도 있다. 이처럼 데이터 전송 과정에서 오류가 생기면 수신기는 잘못된 신호를 받아 정확한 정보 전달이 어려워진다. 이런 경우에 추가 데이터를 함께 보내서 오류가 발생한 데이터를 검출하거나 복구하는 방식을 사용하는데, 이는 크게 자동 재전송 요구 방식과 순방향 오류 정정 방식으로 나눌 수 있다.

먼저 자동 재전송 요구 방식은 송신기에서 데이터를 전송할 때 데이터 중 1의 개수가 홀수이면 1을, 짝수이면 0의 추가 데이터를 송신 정보 데이터와 함께 보낸다. 수신기에서 받은 수신 정보 데이터의 1의 개수와 추가 데이터의 값을 비교하여 두 값이 다르면 수신기는 전송된 데이터 속에 오류가 있음을 알게 된다.



예를 들어 위의 그림과 같이 7개의 데이터를 전송할 때 네 번째 데이터 비트가 1이 아닌 0으로 수신될 수 있다. 이때 수신 정보 데이터의 1의 개수가 홀수인데 추가 데이터가 0이므로 오류가 발생했다는 것을 알게 된다. 그러나 수신기는 오류가 발생한 데이터의 위치를 알 수 없고, 오류를 정정할 능력 또한 없다. 이 경우 수신기는 자동 재전송 요구를 통해 송신기에 데이터 재전송을 요청한 후 오류를 복구하게 된다.



다음으로 순방향 오류 정정 방식은 송신기에서 전송할 데이터를 위의 그림처럼 각각 두 번씩 복사한 추가 데이터를 송신 정보 데이터와 함께 전송하는 방식이다. 데이터 전송에 오류가 발생한 경우, 수신기는 수신 데이터에서 복사된 데이터들과 비교하여 다른 값으로 전송됐는지를 확인해 오류가 발생한 위치를 알 수 있다. 만약 수신 데이터가 1인데 복사된 데이터들의 값이 모두 0이라면 실제 전송된 데이터는 1이 아닌 0으로, 오류를 고칠 수 있다. 이처럼 순방향 오류 정정 방식은 복사된 추가 데이터를 이용해 수신기가 단독으로 오류를 정정할 수 있다.

* 열잡음 : 수신기나 전송 선로 또는 전파 매체에서 전자 운동이 열에너지에 의해 동요하여 발생하는 잡음.

28. 윗글에 대한 설명으로 가장 적절한 것은?

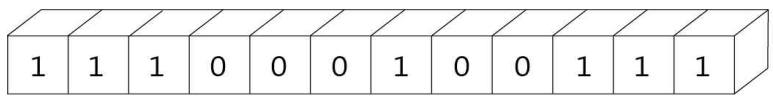
① 다른 대상과의 비교를 통해 가설을 입증하고 있다.
② 설명 대상을 구분하여 각각의 원리를 서술하고 있다.
③ 기술의 변화 양상을 시대별로 나누어 서술하고 있다.
④ 기술의 문제점을 분석하여 향후 개선 방향을 제시하고 있다.
⑤ 전문가의 견해를 인용하여 원리를 체계적으로 설명하고 있다.

29. ⓐ~ⓕ에 대한 설명으로 적절하지 <u>않은</u> 것은?

① ⓐ는 송신 정보 데이터이다.
② ⓑ는 전송 오류를 알기 위해 ⓐ에 ⓒ를 더한 데이터이다.
③ ⓒ는 ⓑ의 데이터 비트 전체의 개수에 따라 값이 결정된다.
④ ⓓ에서 열잡음으로 인해 전송 오류가 발생할 수 있다.
⑤ ⓕ에서 ⓒ의 값과 ⓔ의 분석 결과를 비교해 전송 오류를 확인할 수 있다.

30. <보기>는 '순방향 오류 정정 방식'을 통해 전송된 수신 데이터이다. 윗글을 바탕으로 <보기>에 대해 이해한 내용으로 적절한 것을 있는 대로 고른 것은?

<보 기>



ㄱ. 복사된 추가 데이터의 개수는 모두 8개이다.
ㄴ. 추가 데이터로 확인하면 송신 정보 데이터는 '1011'이다.
ㄷ. 오류가 발생한 위치는 송신 정보 데이터의 세 번째 데이터 비트이다.
ㄹ. 송신기에 추가 데이터 전송을 요청한 후 수신 데이터와 비교해 오류를 정정한다.

① ㄱ, ㄴ ② ㄱ, ㄷ ③ ㄷ, ㄹ
④ ㄱ, ㄴ, ㄷ ⑤ ㄴ, ㄷ, ㄹ

- 출전: ETRI 전파기술연구부 저. 《훤히 보이는 전파기술》
- 정답: 28.② 29.③ 30.②

[38~42] 다음 글을 읽고 물음에 답하시오.

디지털 통신 시스템은 송신기, 채널, 수신기로 구성되며, ⓐ전송할 데이터를 빠르고 정확하게 전달하기 위해 부호화 과정을 거쳐 전송한다. 영상, 문자 등인 데이터는 ⓑ기호 집합에 있는 기호들의 조합이다. 예를 들어 기호 집합 {a, b, c, d, e, f}에서 기호들을 조합한 add, cab, beef 등이 데이터이다. 정보량은 어떤 기호가 발생했다는 것을 알았을 때 얻는 정보의 크기이다. 어떤 기호 집합에서 특정 기호의 발생 확률이 높으면 그 기호의 정보량은 적고, 발생 확률이 낮으면 그 기호의 정보량은 많다. 기호 집합의 평균 정보량*을 기호 집합의 엔트로피라고 하는데 모든 기호들이 동일한 발생 확률을 가질 때 그 기호 집합의 엔트로피는 최댓값을 갖는다.

송신기에서는 소스 부호화, 채널 부호화, 선 부호화를 거쳐 기호를 ⓒ부호로 변환한다. 소스 부호화는 데이터를 압축하기 위해 기호를 0과 1로 이루어진 부호로 변환하는 과정이다. 어떤 기호가 110과 같은 부호로 변환되었을 때 0 또는 1을 비트라고 하며 이 부호의 비트 수는 3이다. 이때 기호 집합의 엔트로피는 기호 집합에 있는 기호를 부호로 표현하는 데 필요한 평균 비트 수의 최솟값이다. 전송된 부호를 수신기에서 원래의 기호로 ⓓ복원하려면 부호들의 평균 비트 수가 기호 집합의 엔트로피보다 크거나 같아야 한다. 기호 집합을 엔트로피에 최대한 가까운 평균 비트 수를 갖는 부호들로 변환하는 것을 엔트로피 부호화라 한다. 그중 하나인 '허프만 부호화'에서는 발생 확률이 높은 기호에는 비트 수가 적은 부호를, 발생 확률이 낮은 기호에는 비트 수가 많은 부호를 할당한다.

채널 부호화는 오류를 검출하고 정정하기 위하여 부호에 잉여 정보를 추가하는 과정이다. 송신기에서 부호를 전송하면 채널의 잡음으로 인해 오류가 발생하는데 이 문제를 해결하기 위해 잉여 정보를 덧붙여 전송한다. 채널 부호화 중 하나인 '삼중 반복 부호화'는 0과 1을 각각 000과 111로 부호화한다. 이때 수신기에서는 수신한 부호에 0이 과반수인 경우에는 0으로 판단하고, 1이 과반수인 경우에는 1로 판단한다. 즉 수신기에서 수신된 부호가 000, 001, 010, 100 중 하나라면 0으로 판단하고, 그 이외에는 1로 판단한다. 이렇게 하면 000을 전송했을 때 하나의 비트에서 오류가 생겨 001을 수신해도 0으로 판단하므로 오류는 정정된다. 채널 부호화를 하기 전 부호의 비트 수를, 채널 부호화를 한 후 부호의 비트 수로 나눈 것을 부호율이라 한다. 삼중 반복 부호화의 부호율은 약 0.33이다.

채널 부호화를 거친 부호들을 채널을 통해 전송하려면 부호들을 전기 신호로 변환해야 한다. 0 또는 1에 해당하는 전기 신호의 전압을 결정하는 과정이 선 부호화이다. 전압의 ⓔ결정 방법은 선 부호화 방식에 따라 다르다. 선 부호화 중 하나인 '차동 부호화'는 부호의 비트가 0이면 전압을 유지하고 1이면 전압을 변화시킨다. 차동 부호화를 시작할 때는 기준 신호가 필요하다. 예를 들어 차동 부호화 직전의 기준 신호가 양(+)의 전압이라면 부호 0110은 '양, 음, 양, 양'의 전압을 갖는 전기 신호로 변환된다. 수신기에서는 송신기와 동일한 기준 신호를 사용하여, 전압의 변화가 있으면 1로 판단하고 변화가 없으면 0으로 판단한다.

* 평균 정보량: 각 기호의 발생 확률과 정보량을 서로 곱하여 모두 더한 것.

38. 윗글에서 알 수 있는 내용으로 적절한 것은?

① 영상 데이터는 채널 부호화 과정에서 압축된다.
② 수신기에는 부호를 기호로 복원하는 기능이 있다.
③ 잉여 정보는 데이터를 압축하기 위해 추가한 정보이다.
④ 영상을 전송할 때는 잡음으로 인한 오류가 발생하지 않는다.
⑤ 소스 부호화는 전송할 기호에 정보를 추가하여 오류에 대비하는 과정이다.

39. 윗글을 바탕으로, 2가지 기호로 이루어진 기호 집합에 대해 이해한 내용으로 적절하지 <u>않은</u> 것은?

① 기호들의 발생 확률이 모두 1/2인 경우, 각 기호의 정보량은 동일하다.
② 기호들의 발생 확률이 각각 1/4, 3/4인 경우의 평균 정보량이 최댓값이다.
③ 기호들의 발생 확률이 각각 1/4, 3/4인 경우, 기호의 정보량이 더 많은 것은 발생 확률이 1/4인 기호이다.
④ 기호들의 발생 확률이 모두 1/2인 경우, 기호를 부호화하는 데 필요한 평균 비트 수의 최솟값이 최대가 된다.
⑤ 기호들의 발생 확률이 각각 1/4, 3/4인 기호 집합의 엔트로피는 발생 확률이 각각 3/4, 1/4인 기호 집합의 엔트로피와 같다.

40. 윗글의 '부호화'에 대한 내용으로 적절한 것은?

① 선 부호화에서는 수신기에서 부호를 전기 신호로 변환한다.
② 허프만 부호화에서는 정보량이 많은 기호에 상대적으로 비트 수가 적은 부호를 할당한다.
③ 채널 부호화를 거친 부호들은 채널로 전송하기 전에 잉여 정보를 제거한 후 선 부호화한다.
④ 채널 부호화 과정에서 부호에 일정 수준 이상의 잉여 정보를 추가하면 부호율은 1보다 커진다.
⑤ 삼중 반복 부호화를 이용하여 0을 부호화한 경우, 수신된 부호에서 두 개의 비트에 오류가 있으면 오류는 정정되지 않는다.

41. 윗글을 바탕으로 <보기>를 이해한 내용으로 적절한 것은? [3점]

<보 기>

날씨 데이터를 전송하려고 한다. 날씨는 '맑음', '흐림', '비', '눈'으로만 분류하며, 각 날씨의 발생 확률은 모두 같다. 엔트로피 부호화를 통해 '맑음', '흐림', '비', '눈'을 각각 00, 01, 10, 11의 부호로 바꾼다.

① 기호 집합 {맑음, 흐림, 비, 눈}의 엔트로피는 2보다 크겠군.
② 엔트로피 부호화를 통해 4일 동안의 날씨 데이터 '흐림비맑음흐림'은 '01001001'로 바뀌겠군.
③ 삼중 반복 부호화를 이용하여 전송한 특정 날씨의 부호를 '110001'과 '101100'으로 각각 수신하였다면 서로 다른 날씨로 판단하겠군.
④ 날씨 '비'를 삼중 반복 부호화와 차동 부호화를 이용하여 부호화하는 경우, 기준 신호가 양(+)의 전압이면 '음, 양, 음, 음, 음, 음'의 전압을 갖는 전기 신호로 변환되겠군.
⑤ 삼중 반복 부호화와 차동 부호화를 이용하여 특정 날씨의 부호를 전송할 경우, 수신기에서 '음, 음, 음, 양, 양, 양'을 수신했다면 기준 신호가 양(+)의 전압일 때 '흐림'으로 판단하겠군.

42. 문맥을 고려할 때, 밑줄 친 말이 ⓐ~ⓔ의 동음이의어가 아닌 것은?

① ⓐ: 공항에서 해외로 떠나는 친구를 전송(餞送)할 계획이다.
② ⓑ: 대중의 기호(嗜好)에 맞추어 상품을 개발한다.
③ ⓒ: 나는 가난하지만 귀족이나 부호(富豪)가 부럽지 않다.
④ ⓓ: 한번 금이 간 인간관계를 복원(復原)하기는 어렵다.
⑤ ⓔ: 이 작품은 그 화가의 오랜 노력의 결정(結晶)이다.

- 정답: 38.② 39.② 40.⑤ 41.④ 42.④

◆ 22년 10월 고3 14~17번

[14 ~ 17] 다음 글을 읽고 물음에 답하시오.

일반적으로 거리는 두 개의 지점이 공간적으로 ⓐ 떨어진 정도를 나타내는 물리적 개념이다. 2차원 평면에 두 지점이 (0, 0)과 (1, 1)에 있다면 두 지점 사이의 최단 거리는 두 점을 잇는 직선의 길이 $\sqrt{2}$가 된다. 한편 거리는 추상적인 성질이나 가치에 대한 차이를 나타내는 척도로도 사용될 수 있다. 이럴 경우 떨어진 정도를 나타내는 기능은 유지되지만, 기준이나 관점에 따라 거리를 계산하는 방법이 달라진다.

거리의 개념은 디지털 데이터에도 적용될 수 있다. 데이터 간의 거리는 추상적 거리의 개념으로, 데이터가 표현하려는 정보에 따라 측정 방법이 다르다. 00, 11과 같은 2비트의 데이터가 2진수로 표현된 수치를 가리킨다면 00과 11의 거리는 두 수치의 차인 $|(0\times2^1+0\times2^0)-(1\times2^1+1\times2^0)|=3$이 된다.

그런데 2비트의 데이터 00이나 11이 어떤 상태를 나타내는 부호라면 거리는 두 부호가 구별되는 정도라 할 수 있다. 해밍 거리는 부호의 관점에서 부호들 간의 거리를 표현하는 방법 중 하나이다. 해밍 거리는 길이가 같은 두 부호를 비교하였을 때 두 부호의 같은 자리에 있는 서로 다른 문자의 개수로 나타낸다. 예를 들어 세 개의 부호 00, 01, 11이 있다면 00과 01의 해밍 거리는 1이고, 00과 11의 해밍 거리는 2이다. 이때 부호들 간의 최소 해밍 거리는 1이고, 최대 해밍 거리는 2이다.

부호들 간의 최소 해밍 거리를 충분히 멀게 한다면 통신이나 저장 과정에서 발생하는 오류를 검출하여 수정할 수 있다. 예를 들어 전송하려는 1비트의 원시 부호 0과 1이 있고 부호 단위로 송수신한다고 가정해 보자. 송신자가 1을 보낸다면 수신자는 0이나 1 중 하나를 받게 될 것이고, 송신자가 어떤 데이터를 보냈는지 알 수 없기 때문에 오류가 발생하더라도 오류가 있는지 알 수 없다. 이 경우 부호들 간의 최소 해밍 거리는 1이다. 0이나 1을 송수신하는 대신 원시 부호(x) 뒤에 확인 부호(p)를 덧붙여 xp에 해당하는 2비트 단위의 전송 부호를 만들어 보자. ㉠ 전송 부호는 고정된 원시 부호에 확인 부호를 덧붙이고, 확인 부호는 원시 부호에 대한 1의 개수가 짝수가 되도록 만든다는 규칙을 정한다면 전송 부호는 00과 11이 된다. 만일 수신자가 01이나 10 중 하나를 받은 경우 전송 부호에 오류가 있음을 알 수 있다. 하지만 어느 자리에서 오류가 났는지 알 수 없기 때문에 오류를 수정할 수는 없다.

[A] 00이나 11을 송수신하는 대신 p와 동일한 규칙의 확인 부호(q)를 한 번 더 덧붙여 xpq에 해당하는 3비트 단위의 전송 부호 000과 111 중 하나를 송수신한다고 가정해 보자. 한 자리의 오류만 있다고 가정하면 수신자가 001, 010, 100, 011, 101, 110 중 하나를 받은 경우 오류 발생 자리를 검출하여 수정할 수 있다. 예를 들어 110의 경우 x인 1에 대해 p와 q는 각각 1이 되어야 1의 개수가 짝수가 되지만 q가 0이므로 1의 개수가 홀수이다. 따라서 오류 발생 자리를 검출하여 110을 111로 수정할 수 있다. 이 경우 전송 부호 간의 최소 해밍 거리가 3이어서 한 자리의 오류를 검출하여 수정할 수 있는 것이다.

원시 부호에 확인 부호를 충분히 덧붙이면 전송 부호의 길이는 길어지지만 전송 부호들 간의 최소 해밍 거리도 함께 멀어져 오류가 많이 발생하더라도 오류를 검출하여 수정하는 것이 가능하다. 하지만 동일한 정보를 보낼 때 덧붙이는 확인 부호의 개수가 늘어나면 보내야 하는 데이터의 양이 늘어나 전송 효율이 낮아진다.

14. 윗글을 통해 알 수 있는 내용으로 적절하지 않은 것은?

① 2진수로 표현된 수치를 가리키는 데이터들 간의 거리는 수치 간의 차로 표현될 수 있다.
② 추상적인 성질이나 가치의 차이를 나타내는 척도로 거리의 개념이 사용될 수 있다.
③ 물리적 개념에서의 거리는 두 지점이 공간적으로 떨어져 있는 정도를 나타낸다.
④ 00과 11의 2진수 수치의 차이와 해밍 거리는 같은 값으로 측정된다.
⑤ 데이터가 표현하려는 정보에 따라 거리를 측정하는 방법이 다르다.

15. [A]와 <보기>를 이해한 내용으로 적절하지 않은 것은? [3점]

< 보 기 >

확인 부호가 오류 발생 자리에 대한 정보가 되도록 규칙을 정하면 전송 부호에서 한 자리 오류가 발생했을 때 수정이 가능하다. 확인 부호를 검사하여 p에 오류가 있으면 [p 자리]를 1로, 오류가 없으면 0으로 표현한다. 같은 방식으로 q에 오류가 있으면 [q 자리]를 1로, 오류가 없으면 0으로 표현한다. 0과 1로 표현된 [p 자리][q 자리]를 계산하면 한 자리의 오류가 발생했을 때 그 자리를 알아낼 수 있다.

송신	수신	규칙			오류 발생 자리
		오류		계산	
		p 자리	q 자리		
000	000	0	0	$0\times2^1+0\times2^0$	□□□
	010		0	$1\times2^1+0\times2^0$	□☑□
	110	0	1	$0\times2^1+1\times2^0$	
	011	1	1	$1\times2^1+1\times2^0$	
⋮	⋮	⋮	⋮	⋮	⋮

① 송신자는 전송 부호 간의 해밍 거리가 3이 될 수 있도록 0은 000으로, 1은 111로 보내는 것이겠군.
② 수신자가 010을 받았다면 [p 자리]의 오류를 1로 표현하여 000으로 판단하겠군.
③ 수신자가 110이나 101을 받았다면 수신한 부호에 있는 0을 1로 수정하여 모두 111로 판단하겠군.
④ 수신자가 011을 받았다면 [p 자리]와 [q 자리] 모두에 오류가 있는 경우이므로 두 자리의 오류를 수정하겠군.
⑤ 수신자가 111을 받았다면 [p 자리]와 [q 자리]의 오류를 모두 0으로 표현하여 오류가 없는 것으로 판단하겠군.

16. ㉠에 대한 이해로 가장 적절한 것은?

① 전송 부호들 간의 최소 해밍 거리를 멀게 하면 전송하는 데이터의 양은 늘어난다.
② 전송 부호들 간의 최소 해밍 거리가 1이면 전송 과정에서의 오류 검출이 가능하다.
③ 두 전송 부호의 같은 자리에 같은 문자의 개수가 많을수록 해밍 거리는 멀어진다.
④ 덧붙이는 확인 부호가 많아지면 전송 부호들 간의 최대 해밍 거리는 가까워진다.
⑤ 전송 부호들 간의 최소 해밍 거리가 가까워질수록 전송 효율은 낮아진다.

17. ⓐ의 문맥적 의미와 가장 유사한 것은?

① 식당은 본관과 조금 떨어져 있는 별관이다.
② 해가 떨어지자 새는 보금자리로 돌아갔다.
③ 그들의 실력은 평균보다 떨어지는 편이다.
④ 상처가 나서 생긴 딱지가 아물어 떨어졌다.
⑤ 물건을 팔면 본전을 빼고 만 원이 떨어진다.

- 정답: 14.④ 15.④ 16.① 17.①

[28~32] 다음 글을 읽고 물음에 답하시오.

데이터를 주고받을 때, 송신 측은 데이터별로 고유하게 부여된 순서 번호에 ⓐ따라 순차적으로 데이터를 송신하고, 수신 측은 데이터의 순서 번호에 맞추어 송신 측에 응답 데이터를 보내준다. 만약 수신 측에서 데이터 전송 오류가 발생한 것을 파악했다면 오류가 발생한 데이터를 다시 전송해 주도록 송신 측에 요청해야 한다. 이때 자동 반복 요청 방식(ARQ)을 주로 사용한다. ARQ에서 오류가 없는 데이터가 도착할 때 송신 측에 보내는 수신 측의 응답을 ACK, 전송받은 데이터에서 오류가 검출될 경우에 보내는 수신 측의 응답을 NAK라고 한다. 그런데 송신 측에서는 데이터를 전송한 시점부터 타이머를 작동해 지정된 시간 동안 수신 측으로부터 아무런 응답이 없는 경우 '타임 아웃'으로 간주한다. 타임 아웃은 수신 측이 송신 측에 응답을 하지 않거나, 송신 측과 수신 측이 주고받는 데이터가 상대 측에 도달하지 못하고 전송이 중단된 경우에 발생한다. 송신 측은 타임 아웃이 되는 동시에 데이터를 재전송한다.

ARQ는 정지-대기 ARQ, 고-백-앤 ARQ, 선택적 재전송 ARQ 등으로 그 유형을 나눌 수 있다. 정지-대기 ARQ는 가장 단순한 자동 반복 요청 방식으로, 수신 측은 송신 측으로부터 받은 데이터를 먼저 수신 측의 버퍼*인 수신 윈도우에 저장한 후 오류 검사를 실시한다. 그 결과에 따라 수신 측은 ACK 또는 NAK를 전송한 후 해당 데이터를 수신 윈도우에서 삭제한다. 송신 측이 수신 측으로부터 ACK를 수신하면 그다음 데이터를 전송하고, NAK를 수신하거나 타임 아웃이 되면 그에 해당하는 데이터를 재전송한다.

고-백-앤 ARQ는 송신 측이 수신 측의 응답을 기다리지 않고 연속해서 순서 번호가 부여된 데이터를 전송하는 방식으로, 오류가 발생하면 오류가 발생한 데이터를 포함하여 이후에 전송된 모든 데이터를 재전송한다. 이 방식에서 수신 측은 데이터를 수신 윈도우에 하나씩 저장하는데, 송신 측으로부터 오류가 없는 데이터를 수신한 경우에는 무조건 ACK를 ⓑ보내지만 오류가 있는 데이터를 수신한 경우에는 NAK를 보내거나 무시할 수 있다. 그리고 오류가 발생한 순번 이후의 데이터에 대해서는 수신을 거부한다. 오류가 있는 데이터에 대해 NAK를 보내는 방식을 명시적 방법, NAK를 보내지 않고 무시하는 방식을 묵시적 방법이라고 한다. 명시적 방법을 사용할 경우 송신 측은 NAK를 수신하거나 타임 아웃이 되면 이에 해당하는 데이터부터 순서대로 모든 데이터를 재전송하지만, 묵시적 방법을 사용할 경우 송신 측은 타임 아웃 시간 동안 ACK를 수신하지 않았을 때만 이에 해당하는 데이터부터 순서대로 모든 데이터를 재전송한다.

선택적 재전송 ARQ는 데이터 전송의 기본 원리가 고-백-앤 ARQ와 ⓒ같지만, 오류가 발생할 경우 송신 측에서는 오류가 발생한 데이터만 재전송한다. 수신 측은 먼저 도착한 데이터의 오류 검사가 끝나지 않았더라도 수신한 데이터는 모두 수신 윈도우에 저장한다. 오류가 발생한 이후의 순번 데이터는 ACK를 보내지 않고 수신 윈도우에 저장한 다음, 재전송된 데이터가 도착하면 해당 데이터에 대한 ACK를 보낸 후, 수신 윈도우에 저장된 데이터와 함께 순서 번호를 맞추어 다음 단계로 전달한다. 이 방식 역시 명시적 방법과 묵시적 방법으로 ⓓ나눌 수 있다.

그런데 NAK를 수신하거나 타임 아웃이 발생하여 송신 측이 데이터를 재전송하기 위해서는 송신 측에게도 전송한 데이터를 저장하기 위한 버퍼가 필요한데, 이 버퍼를 송신 윈도우라고 한다. 송신 윈도우에 보관된 데이터는 수신 측에게 전송되었으나, 아직 ACK를 받지 못한 데이터라 할 수 있다. 송신 측이 수신 측으로부터 ACK를 받지 않고도 전송할 수 있는 데이터의 최대 개수를 송신 윈도우 크기라고 한다. 또한 수신 측이 전송받은 데이터에 대한 응답을 보내지 않고도 저장할 수 있는 데이터의 최대 개수를 수신 윈도우 크기라 하는데, 이러한 윈도우의 크기는 데이터 통신 방식에 따라 차이가 난다. 정지-대기 ARQ는 송신 측과 수신 측 모두 하나의 데이터와 그 데이터에 대한 응답 값을 주고받는다는 점에서 송신 윈도우와 수신 윈도우의 크기는 모두 1이 된다. 이와 달리 고-백-앤 ARQ의 경우 송신 측은 ACK를 받지 않아도 여러 개의 데이터를 전송할 수 있기 때문에 수신 윈도우의 크기만 1이 된다. ㉠선택적 재전송 ARQ는 수신 윈도우 크기가 여러 개의 데이터를 송신할 수 있는 송신 윈도우의 크기와 같아 데이터를 더욱 빠르게 전송할 수 있다.

한편 송신 윈도우에 저장된 데이터의 관리는 일반적으로 데이터의 전송이 순서 번호를 기반으로 ⓔ이루어지는 '슬라이딩 윈도우 프로토콜*'에 의해 진행되는데, 이 프로토콜에서는 낮은 순서 번호부터 차례로 데이터 전송이 처리되며 ACK의 회신에 따라 윈도우에 새로 추가될 데이터의 순서 번호도 순차적으로 높은 번호로 이동한다. 이 과정에서 순서 번호에 해당하는 데이터들이 수신 측에 전송된다. 예를 들어, 순서 번호의 최댓값이 9, 송신 윈도우의 크기가 3인 데이터를 전송할 경우, 먼저 '0번, 1번, 2번' 3개의 데이터를 전송한다. 0번 데이터에 대한 ACK가 도착하면 0번 데이터는 송신 윈도우에서 삭제되고, 3번 데이터가 송신 윈도우에 저장되어 수신 측으로 전송된다. 만약 동시에 1번과 2번 데이터의 ACK가 도착하면 송신 윈도우에는 3번 데이터만 남게 되기 때문에 4번과 5번 데이터가 송신 윈도우에 저장되어 수신 측으로 전송된다. 이러한 방식으로 데이터를 전송하다 9번 데이터에 대한 ACK가 도착했다면 다음에 전송되는 데이터는 순서 번호가 0이 되며, 송신 측의 데이터가 모두 전송될 때까지 이 과정이 반복된다.

* 버퍼 : 동작 속도가 크게 다른 두 장치 사이에 접속되어 속도 차를 조정하기 위하여 이용되는 일시적인 저장 장치.
* 프로토콜 : 컴퓨터와 컴퓨터 사이, 또는 한 장치와 다른 장치 사이에서 데이터를 원활히 주고받기 위하여 약속한 여러 가지 규약.

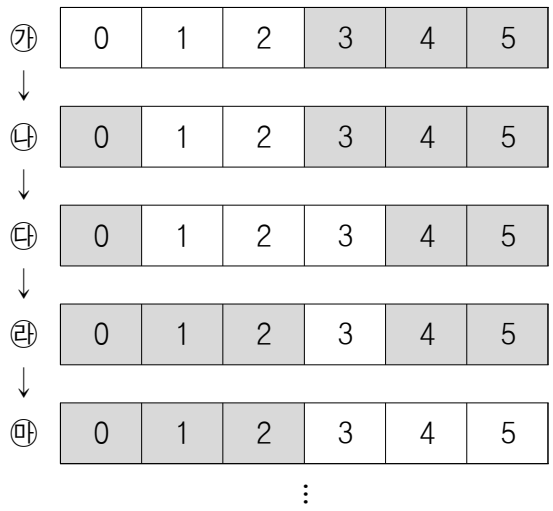
28. 윗글을 통해 알 수 있는 내용으로 가장 적절한 것은?

① 정지-대기 ARQ에서 수신 측은 NAK를 보낸 후에도 해당 데이터를 수신 윈도우에 저장한다.
② 고-백-앤 ARQ에서 수신 윈도우는 정지-대기 ARQ와 마찬가지로 데이터를 하나씩 저장한다.
③ 선택적 재전송 ARQ와 고-백-앤 ARQ 모두 송신 측은 ACK를 수신한 후에 다음 순번의 데이터를 전송한다.
④ 송신 윈도우의 크기는 송신 측이 수신 측으로부터 동시에 받을 수 있는 ACK의 최대 개수에 따라 결정된다.
⑤ 데이터 전송 과정에서 송신 측이 보내는 데이터는 송신 윈도우 크기보다 큰 순서 번호부터 전송된다.

29. 윗글을 바탕으로 <보기>의 '슬라이딩 윈도우 프로토콜'을 이해한 것으로 적절하지 <u>않은</u> 것은?

<보 기>

송신 측에서 수신 측에 전송하려는 데이터의 개수는 12개이다. 송신 측은 순서 번호의 최댓값을 5로 설정한 후, 슬라이딩 윈도우 프로토콜을 이용하여 데이터를 전송하였다. 아래는 데이터 전송 과정에서 송신 윈도우의 데이터 저장 상태를 도식화한 것이다.



* ㉮: 송신 윈도우의 최초 저장 상태
* □: 윈도우에 저장된 데이터 / * □: 윈도우에 저장되지 않은 데이터

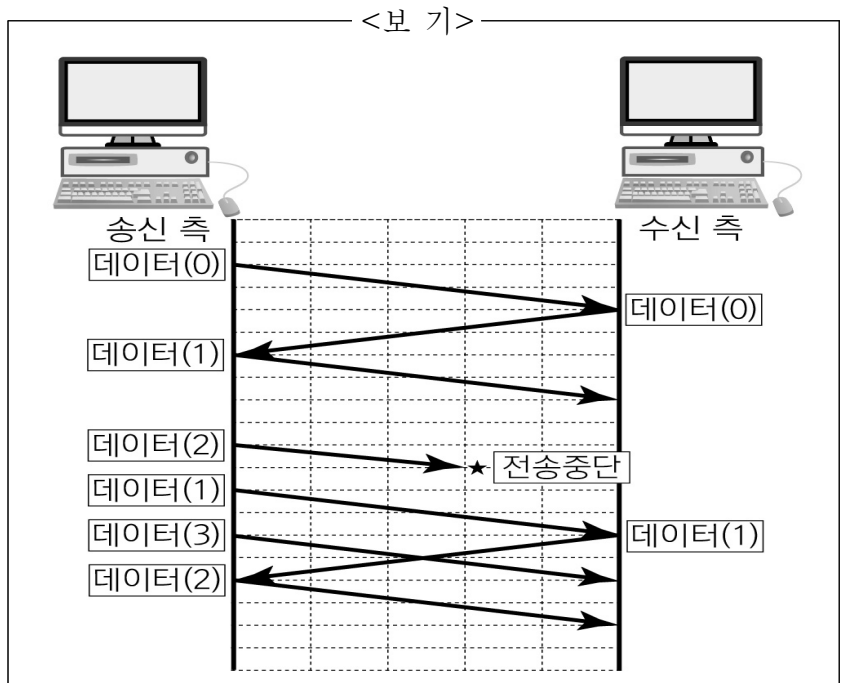
① ㉮를 통해 알 수 있는 송신 윈도우의 크기는 3이다.
② ㉰에서 순서 번호 '3'에 해당하는 데이터가 저장된 것은 ㉮에서 보낸 데이터의 ACK가 모두 도착했기 때문이다.
③ '㉯→㉰' 과정에서 송신 윈도우에 추가된 데이터의 수는 '㉱→㉲' 과정에서 송신 윈도우에 추가된 데이터의 수보다 적다.
④ ㉲에서 전송한 데이터에 대한 ACK가 모두 도착했다면, 바로 다음에 전송되는 데이터의 순서 번호는 ㉮와 같다.
⑤ '㉮→㉲'의 과정이 한 번 더 반복된 후 송신 측이 보낸 데이터의 ACK가 모두 도착했다면, 송신 윈도우에 저장된 데이터의 수는 0개이다.

30. ㉠의 이유를 추론한 것으로 가장 적절한 것은?

① 먼저 도착한 데이터부터 순서대로 데이터 오류 검사를 실시하기 때문에
② 오류 검사가 끝나면 수신 윈도우에 저장된 데이터가 모두 삭제되기 때문에
③ 수신 윈도우에 저장된 데이터의 순번과 상관없이 ACK를 보낼 수 있기 때문에
④ 순번이 빠른 데이터의 오류 검사가 끝나지 않아도 데이터의 수신이 가능하기 때문에
⑤ 데이터에 오류가 발생하면 해당 데이터가 재전송될 때까지 데이터 수신을 거부하기 때문에

31. <보기>는 자동 반복 요청 방식을 이용한 데이터 전송 오류 제어 과정의 일부를 도식화한 것이다. 윗글을 참고하여 <보기>를 이해한 내용으로 적절하지 <u>않은</u> 것은? [3점]

<보 기>



* ( )의 숫자는 데이터의 순서 번호를 나타냄.
* 최초 전송된 데이터(2)는 수신 측에 도달하지 못한 것을 나타냄.

① 데이터(1)을 재전송한 후 데이터(3)을 전송하는 것을 보니 <보기>의 오류 전송은 선택적 재전송 ARQ 방식에 해당하겠군.
② 처음 수신한 데이터(1)에 대한 응답 값을 수신 측이 전송하지 않은 것으로 보아 <보기>는 묵시적 방법에 해당하겠군.
③ 데이터(1)을 전송한 후 데이터(1)을 재전송하는 데 걸린 시간은 '타임 아웃'으로 설정된 시간에 해당되겠군.
④ 송신 측이 데이터(2)를 재전송한 이유는 최초 전송된 데이터(2)에 대해 수신 측이 NAK를 보내지 않았기 때문이겠군.
⑤ 수신 측이 데이터(3)과 재전송된 데이터(2)에 대해 ACK를 보낸다면 데이터(2)와 데이터(3)은 순서 번호에 맞추어 다음 단계로 전달되겠군.

32. 문맥상 ⓐ~ⓔ의 단어와 가장 가까운 의미로 쓰인 것은?

① ⓐ: 그들은 법에 <u>따라</u> 문제를 해결했다.
② ⓑ: 관중들은 선수들에게 응원을 <u>보내느라</u> 정신이 없었다.
③ ⓒ: 여행을 할 때에는 신분증 <u>같은</u> 것을 가지고 다녀야 한다.
④ ⓓ: 수익은 공정하게 <u>나누어야</u> 불만이 생기지 않는다.
⑤ ⓔ: 열심히 노력했더니 소원이 <u>이루어졌다</u>.

- 정답: 28.② 29.② 30.④ 31.④ 32.①

◆ 22년 3월 고1 26~30번

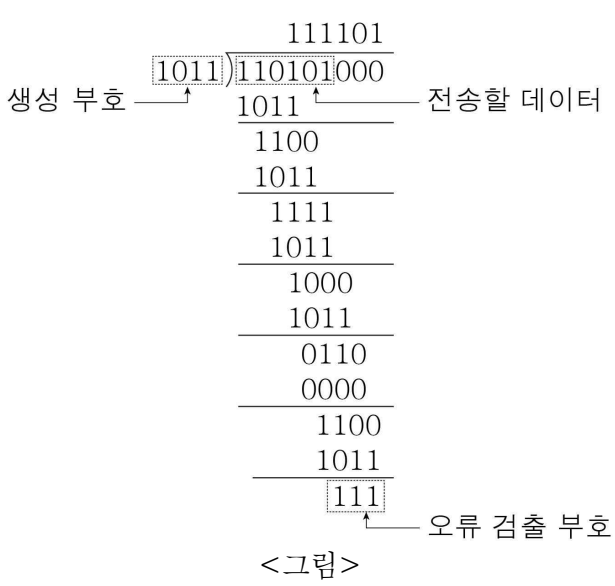
[26 ~ 30] 다음 글을 읽고 물음에 답하시오.

컴퓨터 네트워크에서 데이터가 전송될 때 수신된 데이터에 오류가 있는 경우가 있다. 오류를 검출하기 위해 송신기는 오류 검출 부호를 포함한 데이터를 전송하고 수신기는 수신한 데이터를 검사하여 오류가 있으면 재전송을 요청한다.

수신한 데이터에 오류가 있는지 검출하는 가장 간단한 방식은 ㉠ 패리티 검사이다. 이 방식은 전송할 데이터에 패리티 비트라는 오류 검출 부호를 추가하는 방법으로, 패리티 비트를 추가하여 데이터의 1의 개수를 짝수나 홀수로 만든다. 1의 개수를 짝수로 만드는 방식을 짝수 패리티, 홀수로 만드는 방식을 홀수 패리티라고 하고 송·수신기는 모두 같은 방식을 사용해야 한다. 예를 들어 짝수 패리티를 사용한다면 송신기는 항상 데이터의 1의 개수를 짝수로 만들어서 전송하지만 만일 수신한 데이터의 1의 개수가 홀수가 되면 수신기는 오류가 발생했다고 판단하는 것이다. 하지만 패리티 검사는 ㉮ 수신한 데이터에서 짝수 개의 비트에 오류가 동시에 있으면 이를 검출하기 어렵다. 또한 오류의 발생 여부를 검출할 수 있을 뿐 데이터 내 오류의 위치는 알아낼 수 없다.

전송할 데이터를 2차원 배열로 구성해서 패리티 비트를 생성하면 오류의 발생 여부뿐만 아니라 오류의 위치도 알아낼 수 있다. 예를 들어 송신기가 1100011 1111111을 전송한다고 하자. 송신기는 이를 $\begin{matrix}1100011\\1111111\end{matrix}$과 같이 2차원 배열로 구성하고 가로 방향인 모든 행과 세로 방향인 모든 열에 패리티 비트를 생성한 후 이를 포함한 데이터를 전송한다. 수신기는 수신한 데이터의 각각의 행과 열의 1의 개수를 세어 오류를 검사한다. 만약 어떤 비트에 오류가 발생하면 그 비트가 포함된 행과 열에서 모두 오류가 검출된다. 따라서 오류가 발생한 위치를 알 수 있다. 다만 동일한 행 또는 열에서 짝수 개의 오류가 발생하면 오류가 발생한 정확한 위치를 알 수 없다.

㉡ CRC 방식은 미리 선택된 생성 부호를 사용해서 오류 검출 부호를 생성하는 방식이다. 전송할 데이터를 생성 부호로 나누어서 오류 검출 부호를 생성하는 데 모듈로-2 연산을 활용한다. 모듈로-2 연산은 자릿수가 제한된 상태에서 나머지를 구하는 연산으로 해당 자릿수의 비트 값이 같으면 0, 다르면 1이 된다.



<그림>

<그림>과 같이 생성 부호가 1011이고 전송할 데이터가 110101인 경우를 보자. 전송할 데이터는 오류 검출 부호를 추가해야 하기 때문에 그만큼의 비트가 더 필요하다. 송신기는 전송할 데이터의 오른쪽 끝에 생성 부호의 비트 수보다 하나 작은 비트 수만큼 0을 추가한 후 이를 생성 부호로 나누고 그 나머지가 오류 검출 부호가 된다. 송신기는 오류 검출 부호를 포함한 데이터 ㉢ 110101111만을 전송하고 수신기는 수신한 데이터를 송신기와 동일한 생성 부호로 나눈다. 수신한 데이터는 전송할 데이터에 나머지를 추가했으므로 오류가 없다면 생성 부호로 나누었을 때 나머지가 0이 된다. 이때 나머지가 0이 아니면 수신한 데이터에 오류가 있다고 판단한다. CRC 방식은 복잡하지만 여러 개의 오류가 동시에 생겨도 이를 검출할 수 있어서 오류 검출 확률이 높다.

26. 윗글에서 알 수 있는 내용으로 적절하지 않은 것은?

① CRC 방식은 모듈로-2 연산을 사용해서 생성 부호를 만들어 낸다.
② 패리티 검사에서 송신기와 수신기는 동일한 패리티 방식을 사용해야 한다.
③ CRC 방식에서 생성 부호의 비트 수는 오류 검출 부호의 비트 수보다 하나가 더 많다.
④ 짝수 패리티는 패리티 비트를 포함한 데이터의 1의 개수가 짝수인지 여부를 검사한다.
⑤ CRC 방식은 여러 개의 오류가 동시에 생겨도 검출할 수 있어서 오류 검출 확률이 높다.

27. ㉠과 ㉡에 대해 이해한 내용으로 적절하지 않은 것은?

① ㉠은 ㉡과 달리 데이터에 포함된 1의 개수가 짝수나 홀수가 되도록 오류 검출 부호를 생성한다.
② ㉡은 ㉠과 달리 데이터의 오류를 검출하기 위해 송신기와 수신기 모두에서 오류 검사를 해야 한다.
③ ㉠과 ㉡은 모두, 수신한 데이터의 오류 발생 여부를 수신기가 판단한다.
④ ㉠과 ㉡은 모두, 데이터를 전송하기 전에 오류 검출 부호를 생성해야 한다.
⑤ ㉠과 ㉡은 모두, 전송할 데이터가 같더라도 오류 검출 부호는 다를 수 있다.

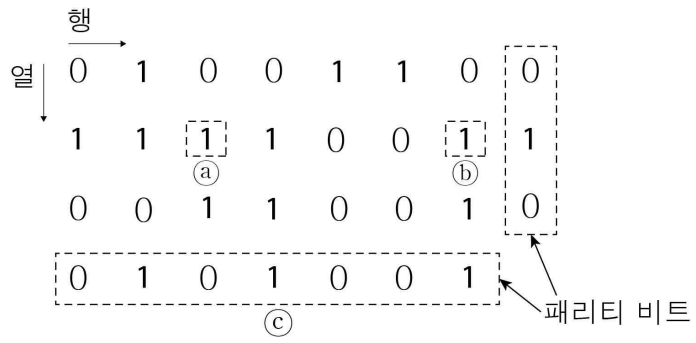
28. ㉮의 이유로 가장 적절한 것은?

① 송신기가 패리티 비트를 생성하는 것이 불가능하기 때문에
② 전송되는 데이터에 포함된 1의 개수가 항상 홀수로 나타나기 때문에
③ 전송되는 데이터에 포함된 1의 개수가 항상 짝수로 나타나기 때문에
④ 오류가 발생했을 때 전송되는 패리티 비트의 크기가 늘어나기 때문에
⑤ 수신한 데이터가 정상일 때와 수신한 데이터에 오류가 있을 때의 패리티 비트가 동일하기 때문에

29. 윗글을 바탕으로 <보기>를 설명한 내용으로 적절하지 <u>않은</u> 것은? [3점]

< 보 기 >

송신기는 오류 검출 방식으로 홀수 패리티를 활용하기로 하였다. 수신기는 수신한 데이터에 오류가 있다고 다음과 같이 판단하였다.



(단, 패리티 비트의 오류는 없다고 가정한다.)

① 첫 번째 행은 패리티 비트를 포함한 데이터의 1의 개수가 홀수이므로 오류가 없다고 판단했을 것이다.
② 여섯 번째 열은 패리티 비트를 포함한 데이터의 1의 개수가 홀수이므로 오류가 없다고 판단했을 것이다.
③ ⓐ가 포함된 행과 열의 패리티 비트를 포함한 데이터의 1의 개수가 각각 짝수이므로 수신기는 ⓐ를 오류라고 판단했을 것이다.
④ 수신한 데이터에서 ⓑ도 0으로 바뀌어서 수신되었다면 데이터의 오류 발생 여부를 검출할 수 없었을 것이다.
⑤ 짝수 패리티를 활용했다면 송신기는 ⓒ를 1010110으로 생성했을 것이다.

30. <보기>는 수신기가 ㉢의 오류를 검사한 연산이다. 윗글을 바탕으로 <보기>를 이해한 내용으로 적절하지 <u>않은</u> 것은?

< 보 기 >

```
      111101
1011)110101111
     1011
      1100
      1011
       1111
       1011
        1001
        1011
         0101
         0000
          1011
          1011
             0
```

① 수신기는 송신기와 동일한 생성 부호인 '1011'을 사용하여 모듈로-2 연산을 하였군.
② 수신기가 수신한 데이터의 오른쪽 끝에 있는 '111'은 송신기에서 생성한 오류 검출 부호이군.
③ 수신기가 모듈로-2 연산을 할 때는 수신한 데이터에 생성 부호보다 하나 작은 비트 수만큼의 0을 추가하지 않았군.
④ 수신기가 연산한 몫인 '111101'이 송신기가 전송한 데이터와 동일하기 때문에 수신기는 오류가 없다고 판단했겠군.
⑤ 수신기가 연산한 결과의 나머지가 0이 아니었다면 수신기는 송신기에 재전송을 요청했겠군.

- 정답: 26.① 27.② 28.⑤ 29.④ 30.④