

## CSE4175 기초컴퓨터네트워크

### 2022년 1학기 기말고사

※ 답안지에 답을 작성하면 됩니다. 과정을 반드시 적을 필요는 없으나, 답이 틀린 경우에 과정이 맞으면 부분점수를 받을 수 있습니다.

#### 1. 다음 명제에 대해 참, 거짓을 쓰시오. (총 12점, 문제당 1점)

(1) 최신 Wi-Fi의 표준인 802.11ax에서는 1024QAM 모듈레이션을 지원한다. 다른 configuration이 모두 같고 SNR이 충분히 높다고 가정할 때, 1024QAM을 사용하는 경우 전송률은 QPSK를 사용할 때의 5배가 된다.

(2) IP header의 flag field에는 'MORE FRAGMENTS (M)' 플래그가 있다. 이 비트가 0이면 해당 패킷은 분할되지 않은 패킷이다.

(3) IP 패킷을 분할 (fragmentation) 할 때, IP header는 맨 처음 조각에만 붙인다.

(4) Classless addressing에는 세 가지 규칙이 있다. 16개의 주소가 필요하여 210.115.227.208에서 210.115.227.223까지 할당하였다면 규칙에 맞게 할당한 것이다.

(5) 내 스마트폰이 사용하는 MAC address는 내가 장소를 변경할 때마다 달라질 수 있다.

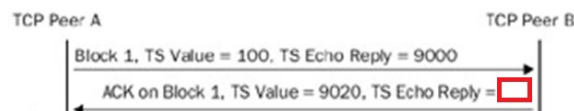
(6) Count-to-infinity 문제는 distance vector 라우팅에서 발생할 수 있는 문제로, 이러한 문제가 발생하는 이유는 라우터가 목적지까지의 경로 정보 전체를 알지 못하기 때문이다.

(7) OSPF (Open Shortest Path First)는 link-state routing 방식의 라우팅 프로토콜이다.

(8) UDP는 송신자가 메시지를 송신한 순서대로 수신자가 수신하는 것을 보장하지 않기 때문에 UDP 헤더에는 시퀀스 넘버 (sequence number) 필드가 없다.

(9) TCP 연결 종료 시에, 클라이언트가 서버에게 FIN 메시지를 보내고, 서버가 그에 대한 ACK만 보내고 자신은 클라이언트에게 FIN을 보내지 않으면 half-close 상태가 된다. 이러한 상태에서는 서버는 클라이언트에게 데이터를 보낼 수 있으나 클라이언트는 서버에게 더 이상 데이터를 보낼 수가 없다.

(10) 아래 그림은 TCP 헤더의 옵션 중 timestamp에 관한 것이다. 네모 안에 들어가는 값은 120이다.



(11) TCP Vegas에서는 패킷 손실이 발생하지 않은 상황에서도 CWND 값을 줄일 수가 있다.

(12) DHCP는 목적지의 IP 주소는 알지만 MAC 주소는 알지 못할 때 MAC 주소를 알아내기 위해 사용하는 프로토콜이다.

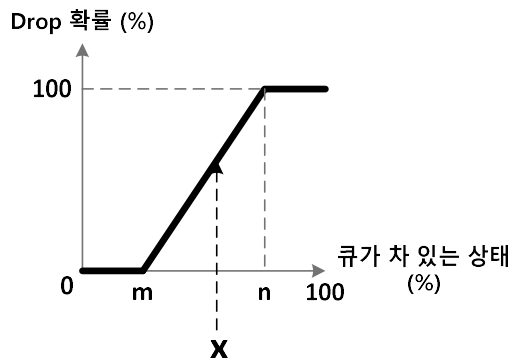
2. 빈 칸에 들어갈 내용을 알맞게 쓰시오. (한글/영어 둘 다 괜찮음) (총 12점, 문제당 2점)

(1) 210.115.227.64/27 은 어떤 주소의 집합을 표현한 것이다. 이 주소 집합에 속하는 맨 마지막 IP 주소를 쓰시오. ( )

(2) TCP 헤더에 있는 'Window Size' 필드는 수신자의 남은 버퍼 사이즈를 송신자에게 알려줌으로써 송신자가 그 이상으로 데이터를 전송하지 않게끔 하는 필드이다. 이렇게 수신자의 버퍼 상태에 따라 송신 속도를 제어하는 것을 ( ) 이라고 한다.

2^16 (3) TCP 연결을 맺은 상대방으로부터 패킷을 수신하였는데, TCP 헤더 안에 있는 Window Size 필드의 값이 65,535이고, TCP 헤더의 옵션 중 Window Scale 필드의 값이 4이었다. 그러면 나의 RWND (Receiver Window)의 값은 얼마가 되는가? ( ) bytes

(4) 라우터에서 RED (Random Early Detection) 기술을 사용한다고 하자. 아래 그래프에서 X축은 라우터의 큐가 얼마나 차 있는가를 나타낸 것이고, Y축은 그에 따른 패킷 드랍확률을 표시한 것이다. 현재 큐가 차 있는 상태가 x라고 할 때, 패킷 드랍 확률 y는 얼마가 되는가? (x는 m과 n 사이라고 가정, 답은 식으로 쓰시오.) ( y = )



(5) 어떤 TCP 연결에서 timeout 값을 지정하기 위하여 exponential averaging 방식으로 RTT (왕복 시간)의 추정값(estimatedRTT)을 구하려고 한다. RTT 추정 값을 계산하기 위한 식은 아래와 같으며, a는 0.8로 정하였다.

$$\bullet \quad \text{estimatedRTT} = a \times \text{estimatedRTT} + (1-a) \times \text{sampleRTT}$$

RTT 값을 4번 측정한 결과 순서대로 (100ms, 200ms, 170ms, 80ms) 였다면, 이 네 개를 측정하고 난 후의 estimatedRTT는 얼마인가? 단, 맨 처음에는 추정값이 없으므로 측정값(sampleRTT)이 그대로 추정값이 된다.

( ) ms

(6) TCP congestion control에는 두 가지의 단계 (phase)가 있다. 이 중에서 CWND가 SSThresh 값에 도달하면 시작되는 단계로 RTT마다 CWND를 1씩 증가시키는 단계를 ( ) phase 라고 한다.

※ 3-4번은 그림 1(다음 페이지에 있음)에 있는 네트워크를 참조하여 해결하도록 한다.

### 3. Distance Vector Routing (총 8점, 문제당 2점)

그림 1과 같은 망에서 distance vector routing을 이용하여 라우팅 테이블을 구성하려고 한다. 아래의 물음에 답하시오.

라우터 A, D, E, G, H의 초기 라우팅 테이블은 다음과 같다.

A의 테이블			D의 테이블			E의 테이블			G의 테이블			H의 테이블		
목적지	거리	다음	목적지	거리	다음	목적지	거리	다음	목적지	거리	다음	목적지	거리	다음
B	30	B	A	2	A	A	-	-	A	-	-	A	-	-
C	-	-	B	-	-	B	14	B	B	-	-	B	-	-
D	2	D	C	-	-	C	-	-	C	-	-	C	-	-
E	-	-	E	15	E	D	15	D	D	4	D	D	-	-
F	-	-	F	-	-	F	11	F	E	-	-	E	4	E
G	-	-	G	4	G	G	-	-	F	-	-	F	-	-
H	-	-	H	-	-	H	4	H	H	5	H	G	5	G
I	-	-	I	-	-	I	-	-	I	-	-	I	1	I

(1) 초기 상태에서, E가 H에게 distance vector 메시지를 전송하였다. H가 메시지를 수신하고 라우팅 테이블을 업데이트 한 후의 H의 테이블을 작성하시오. (아래 **(1) H의 테이블**에 작성)

(2) (1)번 이벤트가 있고 난 후에, H가 G에게 distance vector 메시지를 전송하였다. G가 메시지를 수신하고 라우팅 테이블을 업데이트 한 후의 G의 테이블을 작성하시오. (아래 **(2) G의 테이블**에 작성)

(3) (2)번 이벤트가 있고 난 후에, G가 D에게 distance vector 메시지를 전송하였다. D가 메시지를 수신하고 라우팅 테이블을 업데이트 한 후의 D의 테이블을 작성하시오. (아래 **(3) D의 테이블**에 작성)

(4) (3)번 이벤트가 있고 난 후에, D가 A에게 distance vector 메시지를 전송하였다. A가 메시지를 수신하고 라우팅 테이블을 업데이트 한 후의 A의 테이블을 작성하시오. (아래 **(4) A의 테이블**에 작성)

(1)

H의 테이블		
목적지	거리	다음
A		
B		
C		
D		
E		
F		
G		
I		

(2)

G의 테이블		
목적지	거리	다음
A		
B		
C		
D		
E		
F		
H		
I		

(3)

D의 테이블		
목적지	거리	다음
A		
B		
C		
E		
F		
G		
H		
I		

(4)

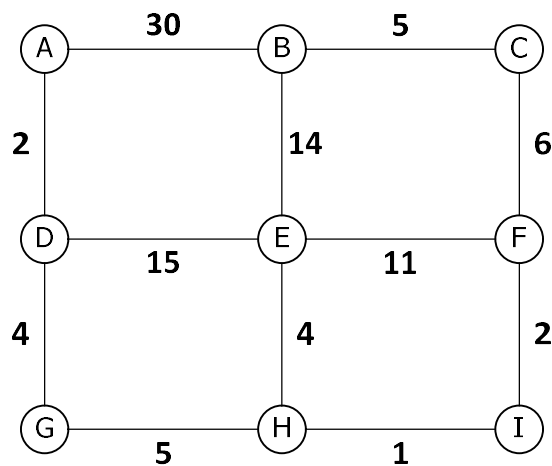
A의 테이블		
목적지	거리	다음
B		
C		
D		
E		
F		
G		
H		
I		

#### 4. Link State Routing (총 8 점, 라인당 1 점)

그림 1과 같은 망에서, 이번에는 link state routing으로 라우팅 테이블을 구성하려고 한다. 노드 A가 모든 노드로부터 link state 정보를 받은 상태라고 가정하고, Dijkstra 알고리즘을 이용하여 A로부터 다른 모든 노드까지의 최단경로를 구하고자 한다. 아래의 테이블은 Dijkstra 알고리즘을 이용하여 최단경로트리 (Shortest Path Tree)를 구하는 과정을 보여주는 테이블이다. 여기서  $D(X)$ 는 A로부터 X까지의 거리를 의미하고,  $P(X)$ 는 최단경로트리에서 X의 부모 노드를 말한다. 아래의 테이블을 완성하시오. (테이블에서 필요한 부분을 채우시오.)

Step	SPT	$D(B), P(B)$	$D(C), P(C)$	$D(D), P(D)$	$D(E), P(E)$	$D(F), P(F)$	$D(G), P(G)$	$D(H), P(H)$	$D(I), P(I)$
0	A	30, A	-	2, A	-	-	-	-	-
1									
2									
3									
4									
5									
6									
7									
8									

그림 1. 문제 3-4에서 사용되는 네트워크. A-I는 라우터. 그래프의 선에 표시된 숫자는 “거리”를 나타낸다.



## 5. Internet Protocol (10점, 문제당 2점)

어떤 IP 패킷을 수신하였는데, IP header의 20바이트가 아래와 같았다. 옵션은 없었다고 가정한다.

**0x450005DC000105C814060000D273E6C6CAB3B115**

그림 2의 IP header 포맷을 참조하여 아래 물음에 답하시오.

- (1) 이 패킷의 IP header의 길이는 몇 바이트인가? (            )
- (2) 이 패킷의 데이터 부분(IP header 제외)의 길이는 몇 바이트인가? (            )
- (3) 이 패킷의 수명은 몇 홉(hop)인가? (            )
- (4) 이 패킷의 목적지 IP address는 무엇인가? Dotted decimal notation으로 쓰시오.  
(            )
- (5) 이 패킷은 어떤 데이터그램의 분할된 한 조각이다. 그렇다면 원래 분할되기 전의 데이터그램의 크기 (IP header 제외)는 몇 바이트인가? (            )

그림 2. IP header 포맷.

VER 4 bits	HLEN 4 bits	Service 8 bits	Total length 16 bits	
Identification 16 bits			Flags 3 bits	Fragmentation offset 13 bits
Time to live 8 bits		Protocol 8 bits	Header checksum 16 bits	
Source IP address				
Destination IP address				
Option				

### 6. Prefix Routing (10점, 문제당 2점)

아래는 서울에 있는 라우터의 테이블이다. 아래의 목적지 IP address에 대하여, next hop을 쓰시오. (각 2점)

목적지	next hop
70.0.0.0 / 8	유럽
70.48.0.0 / 15	스페인
70.64.0.0 / 14	프랑스
70.176.0.0 / 15	체코
70.236.0.0 / 14	노르웨이
70.64.0.0 / 10	서유럽
70.96.0.0 / 13	독일
70.160.0.0 / 11	동유럽
70.241.0.0 / 16	터키
70.224.0.0 / 12	스칸디나비아

(1) 70.36.115.98            (                    )

(2) 70.232.10.101        (                    )

(3) 70.67.235.12         (                    )

(4) 70.100.35.64         (                    )

(5) 70.178.96.20         (                    )

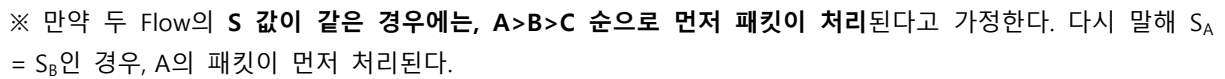
## 7. TCP Congestion Control (총 9점, 문제당 3점)

송신자와 수신자 간의 RTT가 100ms로 일정하다고 했을 때, TCP 연결을 설정한 후 아래와 같이 이벤트가 발생하였다. (100ms와 900ms의 CWND 값은 힌트로 제공)

0ms	→ connection 시작 (CWND = 1)
100ms	→ CWND = 2
800ms	→ timeout 발생 (CWND가 1이 됨)
900ms	→ CWND = 2
1900ms	→ fast retransmission (3 dup ack) 발생
2700ms	→ fast retransmission (3 dup ack) 발생
3200ms	→ fast retransmission (3 dup ack) 발생
4300ms	→ timeout 발생

- (1) 1400ms에 송신자의 CWND 값은 얼마인가? (3점) (                      )
- (2) 3000ms에 송신자의 CWND 값은 얼마인가? (3점) (                      )
- (3) 5000ms에 송신자의 CWND 값은 얼마인가? (3점) (                      )

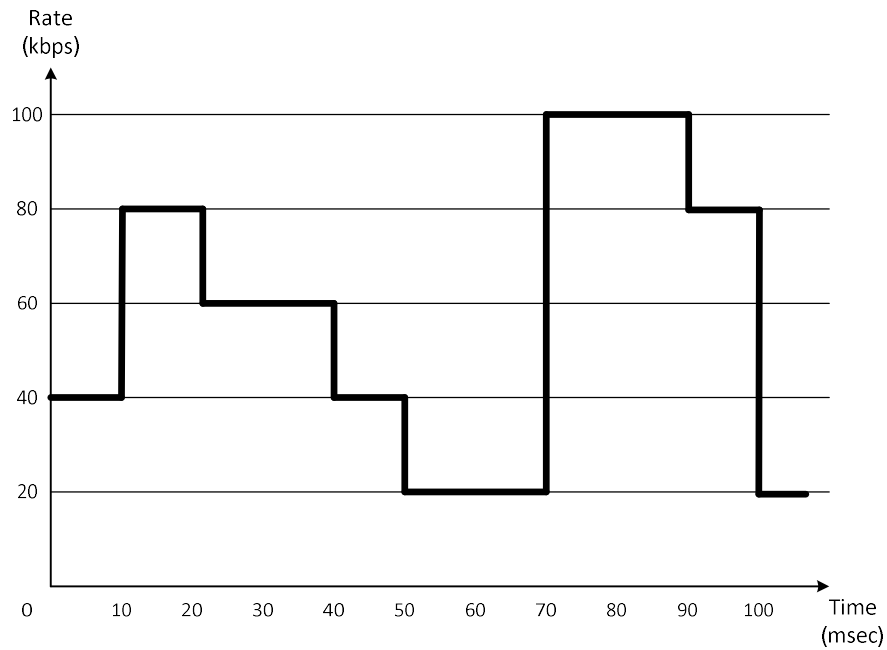
아래 그림은 weighted fair scheduling을 사용하는 라우터의 큐이다. 아래 그림에서 괄호 안의 숫자는 패킷의 번호이고, 그 옆의 숫자는 패킷의 크기이다. 예를 들어 패킷 (1)의 크기는 8이다.

[illegible][illegible]



### 9. Token bucket (총 9점, 문제당 3점)

어떤 라우터에서 token bucket을 사용하여 traffic shaping을 하려고 한다. 아래의 그림은 시간에 따른 트래픽 발생률을 나타낸 것이다. 이 기간 동안에 token bucket에서 token이 바닥나 delay가 발생하는 일이 생기지 않도록 하기 위해서는 bucket의 사이즈  $B$ 를 최소한 얼마로 해야 하는가? 아래의  $r$ 값 (token이 채워지는 속도)에 대하여  $B$ 를 계산하시오. (※ 주의: x축의 단위는 ms이다. y축의 단위는 kbps이다.)



(1)  $r = 30\text{kbps}$  (3점) (                      ) bits

(2)  $r = 50\text{kbps}$  (3점) (                      ) bits

(3)  $r = 70\text{kbps}$  (3점) (                      ) bits

## 10. RSA algorithm (총 7점)

비대칭 키 암호화 알고리즘 중 하나인 RSA를 이용하여 메시지를 전송하려고 한다. Alice가 송신자이며, Bob이 수신자이다. Bob은 public key ( $e, n$ )과 private key ( $d, \Phi$ )를 생성하여, public key를 공개한다.

Bob이 public key와 private key를 생성하기 위해서는, 먼저 두 개의 소수  $p, q$ 를 선택해야 하고,  $e$ 를 선택해야 한다. Bob이 다음과 같이 선택하였다고 하자.

$$p = 3, q = 11, e = 17$$

(1) Bob의 private key 값 중  $d$ 로 사용될 숫자는 얼마인가? (3점) (                      )

(2) Alice가 Bob에게 암호화하여 보낸 암호문(ciphertext)이  $C=2$ 라고 했을 때, 원래 Alice가 보내고자 했던 평문(plaintext)은 얼마인가? (4점)

(                      )

## 11. Diffie-Hellman algorithm (총 7점)

Diffie-Hellman 알고리즘은 secret key를 직접적으로 전송하지 않고 송신자 Alice와 수신자 Bob이 공유할 수 있도록 하는 알고리즘이다. 도청자가 중간에서 도청한 내용만 가지고는 secret key를 계산해내기 어렵다.

- 먼저 Alice는  $g$ 와  $p$ 값을 선택하여 Bob에게 보낸다.  $p$ 는 소수여야 한다.  $g=2$ 이고  $p=11$ 이라 가정하자.
- 다음으로 Alice와 Bob은 각각  $x$ 와  $y$ 를 선택한다. Alice는  $x=7$ 를 선택하였고, Bob은  $y=12$ 를 선택하였다.

(1) Alice가 Bob에게 보내는  $R_1$ 값은 얼마인가? (2점) (                      )

(2) Bob이 Alice에게 보내는  $R_2$ 값은 얼마인가? (2점) (                      )

(3) 공통된 secret key는 얼마인가? (3점) (                      )

- 한 학기 동안 수고 많으셨습니다! -