

CATALOGUE

<input type="checkbox"/>	0	<input type="checkbox"/>	0	<input type="checkbox"/>	0	<input type="checkbox"/>	0	<input type="checkbox"/>	0	<input type="checkbox"/>	0	<input type="checkbox"/>	0
<input type="checkbox"/>	1	<input type="checkbox"/>	1	<input type="checkbox"/>	1	<input type="checkbox"/>	1	<input type="checkbox"/>	1	<input type="checkbox"/>	1	<input type="checkbox"/>	1
<input type="checkbox"/>	2	<input type="checkbox"/>	2	<input type="checkbox"/>	2	<input type="checkbox"/>	2	<input type="checkbox"/>	2	<input type="checkbox"/>	2	<input type="checkbox"/>	2
<input type="checkbox"/>	3	<input type="checkbox"/>	3	<input type="checkbox"/>	3	<input type="checkbox"/>	3	<input type="checkbox"/>	3	<input type="checkbox"/>	3	<input type="checkbox"/>	3
<input type="checkbox"/>	4	<input type="checkbox"/>	4	<input type="checkbox"/>	4	<input type="checkbox"/>	4	<input type="checkbox"/>	4	<input type="checkbox"/>	4	<input type="checkbox"/>	4
<input type="checkbox"/>	5	<input type="checkbox"/>	5	<input type="checkbox"/>	5	<input type="checkbox"/>	5	<input type="checkbox"/>	5	<input type="checkbox"/>	5	<input type="checkbox"/>	5
<input type="checkbox"/>	6	<input type="checkbox"/>	6	<input type="checkbox"/>	6	<input type="checkbox"/>	6	<input type="checkbox"/>	6	<input type="checkbox"/>	6	<input type="checkbox"/>	6
<input type="checkbox"/>	7	<input type="checkbox"/>	7	<input type="checkbox"/>	7	<input type="checkbox"/>	7	<input type="checkbox"/>	7	<input type="checkbox"/>	7	<input type="checkbox"/>	7
<input type="checkbox"/>	8	<input type="checkbox"/>	8	<input type="checkbox"/>	8	<input type="checkbox"/>	8	<input type="checkbox"/>	8	<input type="checkbox"/>	8	<input type="checkbox"/>	8
<input type="checkbox"/>	9	<input type="checkbox"/>	9	<input type="checkbox"/>	9	<input type="checkbox"/>	9	<input type="checkbox"/>	9	<input type="checkbox"/>	9	<input type="checkbox"/>	9

← codez votre numéro d'étudiant ci-contre, et inscrivez votre nom et prénom ci-dessous.

Nom et prénom : .....
-----------------------

---

**Réseaux : QCM N°1 (M1 2023–2024)**

Noircir complètement à l'encre la case cochée (une croix ne suffit pas)  
Une seule réponse correcte par question.  
Réponse correcte : 2 pts; Réponse fausse : -0.5 pt; Réponse vide : 0 pt

---

**Question [Unite2]** L'unité du délai de propagation d'un lien peut être en

- Mbits
- Mbps ou Mb/s
- millisecondes
- en %

**Question [Fragmentation1]** La fragmentation d'un message en plusieurs paquets

- améliore toujours le délai de bout-en-bout mais rend les erreurs de transmission plus gênantes
- améliore toujours le délai de bout-en-bout et rend les erreurs de transmission moins gênantes
- peut améliorer le délai de bout-en-bout et rend les erreurs de transmission moins gênantes
- peut améliorer le délai de bout-en-bout mais rend les erreurs de transmission plus gênantes

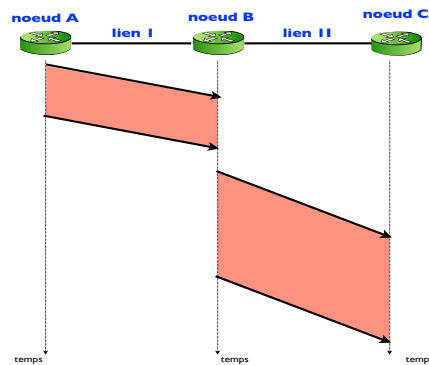
**Question [Delai5]** Le temps d'attente d'un paquet est a priori

- plus court si le paquet est petit
- plus long si le paquet est long
- indépendant de la taille du paquet
- négligeable quelle soit la taille du paquet

**Question [RC3]** Dans le mode Store & Forward d'un réseau à commutation de paquets

- un routeur ne peut envoyer un paquet sur un de ses ports de sortie que s'il a complètement reçu le paquet sur un de ses ports d'entrée
- un routeur peut envoyer le début d'un paquet sur un de ses ports de sortie même s'il n'a pas complètement reçu le paquet sur un de ses ports d'entrée
- un routeur peut envoyer le début d'un paquet sur un de ses ports de sortie dès qu'il a reçu les 150 premiers octets du paquet sur un de ses ports d'entrée
- un routeur ne peut envoyer un paquet sur un de ses ports de sortie que si le buffer associé à ce port de sortie est vide

**Question [Liens1]** Le schéma ci-dessous représente le déroulement du délai de bout-en-bout



d'un paquet sur 2 liens.

- La capacité d'émission du lien II et son délai de propagation sont plus grands que ceux du lien I
- La capacité d'émission du lien II et son délai de propagation sont plus petits que ceux du lien I
- La capacité d'émission du lien II est plus grande que celle du lien I mais son délai de propagation est plus petit
- La capacité d'émission du lien II est plus petite que celle du lien I mais son délai de propagation est plus grand

**Question [Pileprotocolaire2]** L'architecture en 5 couches protocolaires d'Internet

- implique parfois la redondance de fonctions
- permet à la couche  $n+1$  d'offrir des services à la couche  $n$
- ne permet pas la modularité
- le cross-layer (échanges entre couches  $n$  et  $n + 2$ )

**Question [Boutenbout2]** Que signifie de bout-en-bout pour un protocole ?

- il est implémenté seulement sur les terminaux
- il permet d'établir une connexion entre les terminaux
- il assure un transfert de données fiable
- il assure un transfert bidirectionnel (dans les deux sens) entre les terminaux

**Question [Paquet]** Un datagramme, un segment et une trame appartiennent respectivement

- à la couche 3, couche 4 et couche 2
- à la couche 2, couche 3 et couche 4
- à la couche 4, couche 3 et couche 2
- à la couche 4, couche 2 et couche 3

**Question [Delai2]** Pour un même lien, le temps de propagation d'un paquet de données, de plus grande taille qu'un ACK (paquet d'acquittement), est en général

- plus court pour un paquet que pour un ACK
- plus long pour un paquet que pour un ACK
- identique à celui d'un ACK
- long, mais celui d'un ACK est souvent négligeable

**Question [BBPP]** HTTP est un protocole de (1), TCP de (2), et IP de (3)

- (1) bout en bout, (2) bout en bout, (3) proche en proche
- (1) proche en proche, (2) bout en bout, (3) proche en proche
- (1) proche en proche, (2) bout en bout, (3) bout en bout
- (1) bout en bout, (2) proche en proche, (3) proche en proche

## CATALOGUE

**Question [TT3]** Le temps de transfert d'un paquet de 10 kbits sur un lien ayant une capacité de 10 Mb/s et un délai de propagation de 10 ms est de

- 11 ms
- 0,01 s
- 110 ms
- 0,111 s

**Question [TA]** Quel sera le temps d'attente d'un paquet qui se trouve en 6e position dans le buffer d'un routeur (impliquant que 5 autres paquets doivent être transmis avant lui) ? Les paquets ont tous une taille de 1kbits et le lien de communication associé à ce buffer est de 1 Mb/s.

- 5 ms
- 0,05 s
- 6 ms
- 0,06 s

**Question [HTTP2]** HTTP est

- un protocole de transport
- HTTP est un protocole applicatif, qui fonctionne en mode client-serveur et qui repose principalement sur TCP
- HTTP est un protocole applicatif, qui fonctionne en mode client-serveur et qui repose toujours sur UDP
- HTTP est un protocole applicatif, qui fonctionne en mode pair-à-pair et qui repose toujours sur TCP

**Question [Cookies3]** Les cookies

- correspondent à des identifiants choisis par les clients
- sont des valeurs choisies par les serveurs pour des clients
- ne sont jamais diffusés par les clients
- correspondent à des préférences exprimées par les clients auprès des serveurs

**Question [DNS1]** DNS

- est un protocole de transport
- est un protocole applicatif, qui repose toujours sur UDP et qui permet de retrouver l'adresse IP d'une machine hôte
- est un protocole applicatif, qui fonctionne en mode client-serveur et qui repose sur UDP ou sur TCP
- ne tombe jamais en panne

**Question [Processus]** Les processus

- sont définis au niveau de la couche Transport
- communiquent avec la couche Transport via une socket
- sur deux machines différentes communiquent directement entre eux par passage de messages sans passer par la pile TCP/IP
- sont identifiables grâce aux adresses IP des machines seulement

**Question [Multiplexage1]** Dans le multiplexage de la couche Transport sans connexion (par exemple UDP), le numéro de port source est utile

- pour sélectionner la bonne socket sur la machine destination
- pour l'acquittement des données en couche Transport
- pour réguler le débit d'émission
- pour le trajet retour si des paquets doivent être envoyés de la machine destination vers la machine source

**Question [UDP2]** Le protocole de transport UDP assure principalement les services (fonctions) suivants

- le multiplexage et la détection d'erreur bien que la détection d'erreur ne soit pas un service obligatoire
- la détection d'erreur et le contrôle de flux
- le multiplexage et la correction d'erreur
- le multiplexage et la détection d'erreur car ce sont deux services obligatoires pour UDP

**Question [Socket3]** Les paquets UDP destinés à un même nœud

- sont toujours envoyés vers la même socket
- sont envoyés vers des sockets différentes selon leur numéro de port destination
- sont envoyés vers des sockets différentes selon leurs numéros de port source et destination
- ne peuvent pas être acheminés vers des sockets car seuls les paquets TCP peuvent l'être

**Question [SocketDNS]** On suppose qu'à  $t_0$ , un nœud A vient d'envoyer une requête à un serveur DNS en utilisant UDP. Juste après, à  $t_1$ , le nœud A et deux autres nœuds (B et C) envoient chacun 1 requête à ce même serveur DNS. Combien de sockets seront ouvertes sur le serveur dues aux requêtes initiées à  $t_0$  et  $t_1$ ? On suppose qu'initialement (avant  $t_0$ ) aucune socket n'est ouverte sur le serveur.

- 4
- 1
- 2
- 3

**Question [UDP1]** Une application peut-elle bénéficier d'un service de transfert de données fiable si elle repose sur UDP?

- Oui, en implémentant le service de transfert de données fiable dans la couche Application
- Oui, si les protocoles de couche basses ne produisent pas d'erreur de transmission
- Oui, si les tampons ("buffers") sont de tailles suffisantes
- Non, ce n'est pas possible

**Question [CS1]** On considère un segment UDP qui peut être décomposé en deux mots de 16 bits tels que  $mot_1 = 0011001110100110$  et  $mot_2 = 1010001001011100$ . La somme de contrôle de ce segment UDP correspond à

- 0010100111111101
- 1101011000000010
- 1011001101100011
- 1111000011110000

**Question [CS3]** On considère un segment UDP qui peut être décomposé en deux mots de 16 bits tels que  $mot_1 = 1101110001111111$  et  $mot_2 = 1001101110110001$ . La somme de contrôle de ce segment UDP correspond à

- 1000011111001110
- 1000011111001100
- 1011001101100010
- 1000000011110000