Licence 3 - Informatique - UE Réseaux

EXAMEN Contrôle Continu

6 Décembre 2018 (durée : 1h00)

Documents de cours/TD/TP, téléphones et ordinateurs NON autorisés - Calculatrices autorisées

NUMERO D'ETUDIANT(E) (ne pas indiquer votre nom):

Veuillez reporter vos réponses directement sur le sujet de l'examen.

Justifiez et détaillez vos réponses (utiliser si besoin les copies de couleurs).

Exercice 1 : Algorithme de contrôle d'accès au médium : CSMA/CD (7 points)

Soient 4 ordinateurs (A, B, C et D) situées dans le département Marketing de votre entreprise et qui souhaitent transmettre des requêtes SQL à un serveur de Base de données situé dans le même réseau local de technologie Ethernet à 100 Mbit/s. On souhaite déterminer dans quel ordre les requêtes SQL arriveront au serveur, sachant que :

- La durée d'émission d'une trame-requête SQL est de 4 tranches de temps (slot-times).
- A l'instant t=0, les stations A et C souhaitent transmettre une trame chacune. A t=4, les stations B et D décident de transmettre à leur tour une trame.
- Le tirage aléatoire pour chaque station est donné dans le tableau ci-dessous :

Stations	A	В	C	D
1er tirage	2/3	1/2	1/4	3/4
2ème tirage	1/4	3/5	1/2	1/4
3ème tirage	1/8	1/3	1/8	1/8
4 ^{ème} tirage	1/6	1/8	2/3	1/32
5 ^{ème} tirage	1/16	1/4	3/5	1/16

1.1 Quelle est la t	raille (en octets) d'une trame contenant une requête SQL ?	(Entourer la bonne réponse)
(1) 64 octets	(2) 128 octets	(3) 256 octobe

- 1.2 Quelle est la durée d'une tranche de temps dans un réseau Ethernet à 100 Mbit/s ? (Entourer la bonne réponse)
 - (1) 5,12 secondes

(2) 5,12 millisecondes

(3) 5,12 microsecondes

(3) 256 octets

1.3 Représenter sous la forme d'un diagramme temporel démarrant à T=0, les événements (silence, collisions, transmission de trames) survenant dans le réseau lors de l'envoi des trames-requêtes SQL.

T=0										
	_						00	-		

Licence 3 - Informatique - UE Réseaux

Exercice 3 : Contrôle des erreurs de transmission (5 points)

Les protocoles de communications mettent en œuvre des algorithmes de contrôle des erreurs binaires de transmission. Nous nous intéressons dans cet exercice, aux 2 algorithmes « bits de parité » et « somme de contrôle », aussi appelé « CHECKSUM » en anglais. On considèrera dans l'exercice une parité PAIRE. Le bit de parité pour chaque caractère est ajouté à GAUCHE pour compléter l'octet (bit de poids fort).

On souhaite protéger le message HELLO avant transmission aux moyens de ces 2 algorithmes.

On vous donne les codes ASCII (sur 7 bits et en décimale) des caractères du message à transmettre : H (72) E(69) L (76) L (76) 0 (81)

 dans une première étape, on vous propose de calculer le bit de parité pour chaque caractère du message, et de reporter la valeur décimale du nouveau caractère dans la table ci-dessous.

	Code ASCII du caractère en décimale	Code ASCII avec bit de parité en représentation décimale
н	72	
F	69	
	76	
ī	76	
0	81	

2. dans une seconde étape, on vous demande de calculer la somme de contrôle (en décimale) du message incluant les bits de parité. Sachant que la représentation des caractères est dorénavant sur 8bits. Veuillez détailler vos calculs. Votre Réponse :

Valeur du CHECKSUM (en	
décimale)	
decimale)	

- 3. La somme de contrôle (ou CheckSum en anglais) permet au protocole IP d'effectuer un contrôle des erreurs binaires sur les paquets IP. Quelle(s) parties du paquet IP sont protégées par ce checksum ? Entourer la bonne réponse :
 - a) entête IP (header) seul
 - b) données (payload) seules
 - c) entête IP (header) et données (payload)

Licence 3 - Informatique - UE Réseaux

Annexe:

La procédure BACKOFF utilise 3 fonctions :

- random() : tire un nombre réel aléatoire entre 0 et 1.
- int() : rend la partie entière d'un réel
- délai() : calcul le délai d'attente multiple d'un slot_time (51.2 microsec) et est compris entre [0, 2 4].

Avec k = min (n, 10), n = nbre de ré-émission déjá faites

```
Procedure BACKOFF (no_tentative : entier, VAR maxbeckoff : entier)

Const alot-time=51.2 (microsecondes); limite_tentative=16;

Ver delai : entier;

BEGIN

Si (no_tentative =1)

Alors maxbackoff =2 (borne de temps d'attente maximale)

Sinon

Si (tentative < limite_tentative)

Alors maxbackoff = maxbackoff*2;

Sinon maxbackoff = 2 10 (su detu de 10 essats le bome devient constante)

fai délai := int(random() *maxbackoff)

attendre (delai*slot_time)

END
```