

CONCOURS D'AGREGATION D'INFORMATIQUE

SESSION 2020

EPREUVE DE RESEAUX ET DE BASES DE DONNEES

Durée : 4H

Coefficient : 3

- Les calculatrices, téléphones, tablettes, ordinateurs et autres appareils électroniques similaires, ainsi que les documents sont interdits.
- La qualité de la rédaction sera un facteur important d'appréciation des copies. Les candidats sont invités à produire des raisonnements clairs, complets et concis.
- Les candidats peuvent utiliser les résultats énoncés dans les questions ou parties précédentes, en veillant dans ce cas à préciser la référence du résultat utilisé.
- Dans le cas où un(e) candidat(e) repère ce qui lui semble être une erreur d'énoncé, il (elle) le signale très lisiblement sur sa copie, propose la correction et poursuit l'épreuve en conséquence.
- De même, si cela vous conduit à formuler une ou plusieurs hypothèses, il vous est demandé de la (ou les) mentionner explicitement.

L'épreuve est composée de 4 parties indépendantes à rédiger sur des feuilles séparées.

NB : La copie que vous rendrez ne devra comporter aucun signe distinctif tel que le nom, la signature, etc.

N.B. L'épreuve est composée de 4 parties indépendantes à rédiger sur des feuilles séparées.

Partie I

Réseaux informatiques, sécurité et web

A- Réseaux informatiques

Question 1 :

Faites correspondre chaque élément de la colonne A avec l'élément approprié de la colonne B.

Colonne A	Colonne B
1. 10Base T	A. Est connu sous le nom de Fast Ethernet
2. Token Ring	B. Utilise la topologie en bus en étoile avec un câble UTP
3. 10Base5	C. Utilise le passage de jeton dans la topologie bus
4. 10Base2	D. Utilise une méthode d'accès déterministe
5. 100BaseX	E. Utilise la topologie en bus local avec un câble Ethernet fin
6. Token Bus	F. Utilise la topologie en bus avec un câble Ethernet épais

Question 2 :

Une entreprise à succursales multiples utilise l'adresse IP 196.179.110.0. Pour une gestion plus fine de ses sous-réseaux, le responsable informatique désire pouvoir affecter une adresse IP propre à chaque sous-réseau des 10 succursales.

- 2.1- De quelle classe d'adressage s'agit-il ?
- 2.2- Donner et expliquer la valeur du masque sous-réseau correspondant à ce besoin.
- 2.3- Combien de machine chaque sous réseau pourra-t-il comporter et pourquoi ?
- 2.4- Définissez l'adresse de broadcast du sous réseau 3 ? (expliquer)

Question 3 :

- 3.1- Citez des méthodes de commutation interne implémentées par les commutateurs actuels. Laquelle offre le temps de latence le plus court pour une trame et pourquoi ?
- 3.2- Qu'est ce qu'un :

a- Commutateur racine ?

Comment est-il choisi ?

b- Root-port ?

Comment est-il choisi ?

c- Designated-port ?

Comment est-il choisi ?

3.3- C'est quoi

a- Le protocole de Spanning-Tree ?

Quel est son intérêt ?

b- La technique VLSM ?

Quel est son intérêt ?

c- La technique NAT/PAT ?

Quel est son intérêt ?

d- La technique CIDR ?

Quel est son intérêt ?

3.4- Considérons l'adresse CIDR 193.53.32.0/20 d'un réseau. Quel serait le nombre d'hôtes que comporterait ce réseau ?

3.5- Citez 2 inconvénients et 2 avantages des NAT ?

Question 4 :

4.1- Pourquoi le protocole IPv4 a recours parfois à la fragmentation de ses datagrammes ?

4.2- Quel est l'équipement qui effectue la fragmentation des datagrammes ?

4.3- Où se fait le ré-assemblage des datagrammes ?

Quels sont les champs du datagramme qui sont concernés par ce processus ?

4.4- Le processus de fragmentation/défragmentation est coûteux dans IPv4, c'est pourquoi il a été éliminé dans IPv6. Comment IPv6 résout ce problème ?

4.5- Expliquez chacune des commandes suivantes :

a- ping 127.0.0.1

b- ping @IP de la machine source.

c- ping @IP de la passerelle par défaut.

d- ping @IP de la machine distante.

e- ping @URL

B- Sécurité et web

- 1- Quel est le rôle d'un serveur web ?
- 2- Dans un pare-feu (firewall), quel est le rôle d'une zone démilitarisée (DMZ) ?
- 3- Qu'est-ce qu'un proxy inverse (reverse proxy) (on parle aussi de proxy applicatif ou d'application) ?
- 4- Qu'est-ce que le chiffre de César ? Quels sont les moyens qu'un attaquant qui ne connaît pas les paramètres de chiffrement, peut utiliser pour « casser » un texte chiffré par cette méthode (c'est-à-dire découvrir le texte initial (non-chiffré) sans autre information que le texte chiffré) ?
- 5- A quoi sert un VPN ? Comment est-ce mis en œuvre ?
- 6- Qu'est-ce qu'une signature numérique ?
- 7- Qu'est-ce que le masquage d'adresse (dit aussi : « translation par port source ») ?
- 8- Est-ce que l'utilisation du protocole XDR permet à des micro-ordinateurs PC ou Macintosh de dialoguer avec des serveurs RS-6000 ? Justifier.
- 9- Qu'est-ce que le protocole IPSec ? Préciser la couche d'implémentation ainsi que les modes de fonctionnement.

Le protocole SSL a été développé par Netscape puis rebaptisé TLS en 2001 après son rachat par l'IETF (Internet Engineering Task Force).

10.1- Quelle est la couche d'implémentation de ce protocole ?

10.2- Citer trois exemples de protocoles applicatifs qui utilisent ou qui peuvent utiliser SSL/TLS (justifiez vos réponses).

Partie II

Système d'exploitation

1- Ordonnancement des processus.

On considère un système monoprocesseur et les 4 processus P1, P2, P3 et P4 qui effectuent des calculs et des entrées/sorties dans le temps selon le tableau ci-dessous :

Processus P1	Processus P2	Processus P3	Processus P4
Calcul : 3 ms	Calcul : 4 ms	Calcul : 2 ms	Calcul : 7 ms
E/S : 5 ms	E/S : 2 ms	E/S : 3 ms	
Calcul : 2 ms	Calcul : 3 ms	Calcul : 2 ms	
E/S : 1 ms	E/S : 1 ms		
Calcul : 1 ms	Calcul : 1 ms		
Durée d'exécution:	12 ms	11 ms	7 ms

On considère que l'ordonnancement des processus se fait selon une politique à **tourniquet** avec un **quantum** de 2 ms, les dates d'arrivée des processus P1, P2, P3 et P4 sont respectivement 0, 1, 5, 6 et que l'ordre de services des demandes d'E/S se fait selon une politique FIFO.

N.B: Un processus qui fait une E/S se retrouvera dans la file d'attente des processus **bloqués** jusqu'à ce que l'E/S soit effectuée.

1.1- Donner l'évolution dans le temps de l'exécution des 4 processus P1, P2, P3 et P4 en précisant les périodes où chaque processus est à l'état Elu (en exécution).



P1-

P2-

P3-

P4-

1.2- Préciser pour chaque processus, la date de début, la date de fin et le temps de traitement.

Processus	Date d'arrivée	Date de début	Date de fin	Temps de traitement
P1	0			
P2	1			
P3	5			
P4	6			

1.3- Calculer le temps de traitement moyen.

2- Gestion de la mémoire

La gestion de la mémoire peut être faite par partitions non contiguës en utilisant la pagination ou par mémoire virtuelle. Donner les points en commun et ceux différents des deux méthodes.

3- Programmation shell

On veut automatiser la gestion des utilisateurs par des scripts shell, pour cela, on doit créer les fonctions suivantes :

3.1- La fonction **existe** qui recherche un utilisateur dans un fichier. Le nom de l'utilisateur et le choix du fichier sont définis par deux options qui sont transmises à la fonction. Si l'option est égale à la lettre u, la recherche s'effectue dans le fichier /etc/passwd. Si l'option est égale à la lettre g, la recherche s'effectue dans le fichier /etc/group. La fonction transmet un code retour égal à 0 en cas de succès et 1 dans le cas contraire.

3.2- La fonction **creer_liste_users** qui automatise la création de comptes saisis dans un fichier.

4- Commandes Linux

On veut automatiser la gestion des utilisateurs par des scripts shell, pour cela, on doit créer les fonctions suivantes :

4.1- Que va lister la commande suivante `find / -name '*'`

4.2- Quelle option de la commande « ls » utilisée pour afficher le numéro d'inode d'un fichier.

4.3- Un administrateur système souhaite programmer un arrêt du système dans une heure en informant les utilisateurs. Quelle commande exécute-t-il ?

1. Shutdown -t +60 'Arrêt maintenance'
2. Shutdown -h +60 'Arrêt maintenance'
3. Shutdown -s -h +60 'Arrêt maintenance'
4. Shutdown -c +60 'Arrêt maintenance'

4.4- Par quel moyen le système Linux indexe les fichiers ?

1. Lien symbolique
2. Pipe
3. Lien physique
4. Inode

4.5- En Linux, quelle commande permet de changer les droits d'accès d'un fichier ou un répertoire ?

1. lsmod
2. chmod
3. attrib
4. chown

4.6- Quelle option de la commande « rm » est utilisée pour supprimer un répertoire avec tous ses sous-répertoires ?

1. -o
2. -p
3. -r
4. -b

4.7- Quelle commande crée un fichier vide si le fichier n'existe pas ?

1. cat
2. touch
3. read
4. ed

4.8- Laquelle de ces commandes permettra à l'utilisateur de chercher une expression particulière dans un fichier ?

1. ls
2. find
3. grep
4. whereis

Partie III

Modélisation UML

Modélisation du processus de réservation de vols dans une agence de voyage.

On souhaite créer un système sous forme d'une application web permettant la gestion des réservations de vols effectuées dans une agence de voyage.

Dans cette agence, des compagnies aériennes proposent différents vols dont chacun est ouvert à la réservation et fermé sur offre de la compagnie.

Chaque vol, identifié par un numéro et ayant un nombre de places limité, part d'un aéroport de départ à une date de départ et arrive à un aéroport d'arrivée à une date d'arrivée.

Chaque aéroport, identifié par un numéro, possède un nom, existe dans une ville donnée et dessert une ou plusieurs villes.

Un vol peut comporter des escales dans un ou plusieurs aéroport(s). Une escale est faite dans un aéroport entre une date d'arrivée et une date de départ.

Un client qui peut être une personne physique (un particulier) ou une personne morale (entreprise, administration, ...) peut effectuer une ou plusieurs réservations de vols, pour des passagers différents. Ces derniers sont des personnes physiques ayant un passeport dont le numéro est fourni pour la réservation. Une réservation concerne un seul vol et un seul passager.

Une réservation peut être confirmée ou annulée par la suite.

Un agent de l'agence s'occupe de la gestion des vols qui sont importés depuis un fichier envoyé par chaque compagnie.

Un simple visiteur du site peut consulter les vols disponibles de l'agence. Pour pouvoir faire des réservations, il doit s'inscrire. Son compte sera validé par la suite par un administrateur après un appel téléphonique.

Un client, après authentification, peut faire des réservations selon la disponibilité de places des vols désirés. Il pourra ensuite faire le suivi de ses réservations.

Lorsqu'une réservation est validée ou refusée par l'agent, le client reçoit une notification par mail et par SMS. Suite à la validation d'une réservation, le nombre de places du vol est décrémenté.

Le client doit confirmer la réservation au moins une semaine avant le vol. Si ce délai est dépassé, le client recevra un rappel par mail et par SMS lui accordant un jour de plus pour confirmer ou

annuler sa réservation. Si ce délai est dépassé sans réaction du client, sa réservation est annulée automatiquement et il est notifié par mail et par SMS.

Le client doit payer le billet de sa réservation confirmée au moins 48 heures avant le vol pour pouvoir recevoir son billet, sinon sa réservation est annulée automatiquement et il est placé dans la liste noire pour ne plus pouvoir faire de réservations.

L'agent peut déplacer un client de la liste noir vers la liste blanche.

Si un vol est annulé par l'agent, toutes les réservations le concernant seront annulées et les clients seront notifiés par mail et SMS.

Travail à faire :

- 1- A partir des éléments qui vous sont fournis ci-dessus, élaborer le diagramme de classes (en y ajoutant tout attribut que vous jugez pertinent et qui n'a pas été décrit ci-dessus).
- 2- Donnez un diagramme des cas d'utilisations décrivant les différents acteurs et les fonctionnalités proposées par le système.
- 3- Donnez un diagramme d'activités décrivant le processus de réservation.

Partie IV

Bases de données

Cette partie est à traiter en langage Python et le SGBD SQLite3. Les modules python numpy et matplotlib.pyplot sont déjà importés.

Présentation

Les systèmes de transport intelligents (STI) sont l'application des TIC au domaine des transports. Ils reposent sur des fonctions associées à l'intelligence artificielle, sensorielle, de communication, de traitement de l'information et du comportement adaptatif. On trouve les STI dans plusieurs champs d'activité : dans l'optimisation de l'utilisation des infrastructures de transport, dans l'amélioration de la sécurité routière. Le recours aux STI s'intègre aussi dans un contexte de développement durable : ces nouveaux systèmes concourent à la maîtrise de la mobilité et son optimisation pour plus de respect de l'environnement.

Dans ce sujet, on cherche à développer les briques d'un système d'aide à la décision à partir de la base de données "RR.db", présentée ci-dessous, qui modélise un réseau routier d'une ville.

- **Croisement** (*NC* : entier, *NomC* : chaîne, *Longitude* : réel, *Latitude* : réel)
- **Voie** (*NV* : entier, *NomV* : chaîne)
- **Tronçon** (*NT* : entier, *NV* : entier, *NC1* : entier, *NC2* : entier, *Sens* : entier, *Longueur* : entier, *Largeur* : entier)
- **Bouchon** (*NT* : entier, *Sens* : entier, *Dateheure* : date, *VM* : réel)
- **Datamart** (*NT* : entier, *Sens* : entier, *Dateheure* : date, *VM* : réel, *Durée* : entier)

Le réseau routier est modélisé par un ensemble de croisements, de tronçons de voies reliant ces croisements et analysé grâce au relevé instantané de l'état du trafic routier sur le réseau.

- Un croisement de nom *NomC*, situé à une *Longitude* et une *Latitude*, est identifié d'une manière unique par son numéro *NC*.
- Une voie de nom *NomV*, identifiée par le numéro unique *NV*, est formée de tronçons.
- Un tronçon, de *Longueur* et de *Largeur* en mètre, fait partie d'une seule voie et identifié d'une manière unique par son numéro *NT*. Un tronçon relie directement deux croisements de numéros *NC1* et *NC2*. Le sens de circulation dans le tronçon est informé par l'attribut *Sens* qui prend la

valeur 1 si le sens est du croisement $NC1$ vers le croisement $NC2$, -1 pour le sens inverse, 2 pour le double sens et 0 pour les tronçons piétons.

L'état du trafic routier est surveillé par un système de vidéo surveillance et ses perturbations sont enregistrées dans la base de données. S'il y'a une perturbation du trafic sur un tronçon NT dans le sens 1 ou -1, à l'instant $DateHeure$, alors un enregistrement relatif à celle-ci avec la vitesse du trafic sur ce tronçon VM en km/h est ajouté à la table Bouchon.

Si la vitesse change sur le tronçon NT sans qu'elle dépasse la vitesse seuil V_{seuil} (constante du système) alors le champ VM de l'enregistrement correspondant est mis à jour. Si la vitesse du trafic devient supérieure à V_{seuil} alors le trafic est considéré normal (fluide) sur le tronçon et l'enregistrement correspondant est supprimé de la table Bouchon puis archivé dans la table *Datamart* avec la durée en minutes de la perturbation.

Remarque : Dans un tronçon à double sens, chaque sens a son propre état : fluide (par défaut), perturbé, Par conséquent, il peut être stocké et manipulé par sens (voir exemple en Annexe). On propose les aléas V , C , T , B et A dans l'ordre pour les tables *Voie*, *Croisement*, *Tronçon*, *Bouchon* et *Datamart*.

1- Définition de données :

- 1.1- Proposer des clés primaires pour les relations de la base de données.
- 1.2- Ecrire un script SQL qui permet de définir la table *Datamart*.

2- Algèbre relationnelle :

- 2.1- Donner une requête équivalente plus efficace de la requête suivante :

$$\pi_{NT, NomV} (Voie \bowtie Tronçon)$$

- 2.2- Sachant que $card(Voie)=100$ et $card(Tronçon)=800$ et que les chaînes de caractères sont de 30 octets et les entiers sont codés sur 2 octets signés. Donner la taille maximale de l'espace mémoire utilisé lors de l'exécution de cette requête. Comparer avec votre proposition.

3- Interrogation de la base de données :

Ecrire les requêtes SQL qui permettent d'afficher :

- 3.1- Les noms des croisements les plus desservis par les tronçons. Un tronçon qui desserve un croisement est compté une seule fois quelque soit son sens de circulation.
- 3.2- Le numéro de tronçon, le numéro de sa voie et le sens de circulation des k tronçons les plus perturbés par ordre croissant des durées de perturbation.

3.3- Les numéros des tronçons fluides, leur numéro de voie, les noms de leurs deux croisements et leurs sens triés par numéros de voies et de tronçons.

4- Interaction python - Base de données :

Le système de vidéo surveillance n'envoie que les changements dans le réseau, sous forme de tuple d'objets, au système de traitement pour informer les usagers, leur apporter assistance et persister la base de données. L'objet transmis est composé d'attributs __NT, __Sens, __Dateheure, __VM, représentant la vitesse du trafic __VM, son sens __Sens, observée à l'instant __Dateheure, sur un tronçon de numéro __NT qui devient fluide (ou perturbé) ou s'il est déjà perturbé mais la vitesse du trafic change.

4.1- Proposer un modèle d'architecture logicielle d'un tel système.

4.2- Que signifie les deux sous tirets __ devant les attributs de l'objet ? Proposer une classe pour cet objet.

4.3- Ecrire la fonction *Deviation(P)* qui prend en paramètre l'objet *P* représentant un tronçon perturbé et propose des numéros de tronçons, autorisés et non perturbés, de déviations reliant directement les deux croisements *NC1* et *NC2* du tronçon perturbé.

4.4- Ecrire la fonction *Maj(T)* qui prend en paramètre un tuple d'objets *T* des changements dans le réseau et met à jour la base de données. Proposer une solution avec moins d'accès disque.

La table *Datamart* permet de faire des analyses sur le trafic dans le réseau et les paramètres déterminants pour son amélioration. Pour cela, nous allons mettre en place des outils d'affichage de ses paramètres et des corrélations afin d'avoir des tableaux de bord décisionnels.

On analysera la durée totale des perturbations en fonction de la largeur, la longueur, les tranches horaires, par mois, par tronçon et par sens, le nombre de perturbation par tronçon, ...

Pour réduire le nombre d'enregistrements, on ne considère que ceux de durées > 5 mn.

4.5- Ecrire un script qui permet d'afficher une figure composée de 3 sous figures (voir annexe) :

- Figure 1 : Histogramme des occurrences des tronçons perturbés (quelque soit le sens).
- Figure 2 : Un nuage de points représentant la durée totale des perturbations (en heure) par numéro de tronçon (des points en 'o' dans le sens *1* et des points en 'x' dans le sens *-1*).
- Figure 3 : Durée totale (en heure) par tranche horaire 06h-10h59, 11h-15h59 et 16h-21h.

Annexe

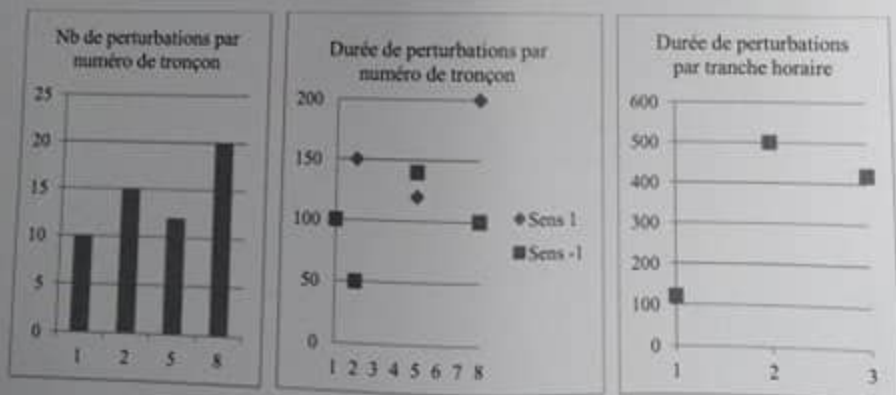
Bases de données

Exemple de base de données :

Croisement			Voie		Tronçon				
NC	NomC	...	NV	NomV	NT	NV	NC1	NC2	Sens
1	"C1"		1	"V1"	1	1	3	4	1
2	"C2"		2	"V2"	2	1	2	4	-1
3	"C3"		3	"V3"	3	1	1	3	2
4	"C4"		4	"V4"	4	2	2	1	1

Bouchon			Datamart				
NT	Sens	...	NT	Sens	Dateheure	VM	Durée
3	1		3	1	2018-10-20 08:10:15	2	15
2	-1		3	1	2019-11-25 09:20:13	5	25
			3	-1	2020-05-10 10:15:30	10	30

Figure :



Fonctions utiles :

subplot (*nbLignes*, *nbColonnes*, *indice*) avec indice : ordre de la sous figure avec $1 \leq k \leq n*m$
plot (*x*, *y*, 'o', *label* = '...')
hist (*x*) : histogramme des occurrences des données *x*.
xlabel ('...')
ylabel ('...')
title ('...')
grid (*True*)
legend ()
strftime ('format', *v*) avec format est %Y-%m-%d %H:%M:%S.%s
strftime ('%s', 'now') : date et heure du système en millisecondes.
import sqlite3
db = sqlite3.connect ('RR.db')
db.close ()
cur = db.cursor ()
db.commit ()
cur.fetchone ()
cur.fetchall ()
cur.execute (*Requête*)