

Matière : Systèmes d'Exploitation 2

Enseignant : M. Faouzi HAJJEM

Filière : L1 GLSI

Durée : 1 H 30

A.U. : 2021/2022

Date : 17 /05/2022

Examen Principal

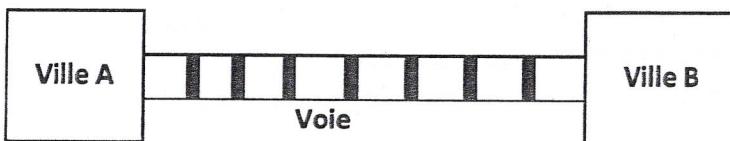
(Session Mai 2022)

Notez bien :

- L'usage de tout document et l'échange de matériels (document, stylo, règle, calculatrice, etc.) entre candidats sont strictement interdits.
- Les candidats ne sont pas autorisés à quitter la salle, temporairement ou définitivement sans remettre leurs copies, même blanches, et signer la liste d'émargement.
- La présentation et la clarté de la feuille d'examen seront prises en compte.
- Cet examen comporte deux pages.

Exercice N° 1 : Synchronisation

Deux villes A et B sont reliées par une seule voie de chemin de fer.



Les règles de circulation sont les suivantes :

- La voie ne doit jamais être allouée simultanément à deux trains allant en sens inverse.
- La voie peut être allouée à un ou plusieurs trains allant tous dans le même sens.
- La priorité de parcours est la même dans les deux sens.

On considère deux classes de processus : les trains allant de A vers B : « **T-AB** » et les trains allant de B vers A : « **T-BA** ».

Processus T-AB

Début

.

Entree_A();

<Circulation sur la voie de A vers B>

Sortie_B();

Fin.

Processus T-BA

Début

Entree_B();

<Circulation sur la voie de B vers A>

Sortie_A();

Fin.

Travail demandé :

1. A quel modèle des problèmes classiques vus en cours, peut-on comparer ce problème de synchronisation ?
2. En utilisant les sémaphores, écrire les codes de quatre procédures *entree_A()*, *entree_B()*, *sortie_A()* et *sortie_B()* de façon que les processus respectent les règles de circulation sur la voie.

Note: Précisez clairement vos déclarations et initialisations.

Exercice N° 2 : Gestion de la mémoire principale

Supposons que l'adressage logique est codé sur 24 bits:

- ✓ Les 2 premiers bits (de poids fort) indiquent le numéro de segment.
- ✓ Les 10 bits suivants donnent le numéro de page.
- ✓ Les 12 derniers bits sont réservés au déplacement dans la page.

L'adresse physique est aussi sur 24 bits :

- ✓ Les 12 bits de poids fort indiquent le numéro de case
- ✓ Les 12 derniers bits spécifient le déplacement dans la case.

Travail demandé :

1. Donnez le nombre de cases (cadres ou frames) en mémoire physique.
2. Expliquez comment convertir une adresse logique en une adresse physique.
3. Soit l'adresse logique 402031_{Hex} , Donnez l'adresse physique correspondante.
On suppose que le segment 01 est chargé dans une zone contiguë commençant à la case (cadre ou frame) numéro 3.

Exercice N° 3 : Gestion de la mémoire Virtuelle

Soit un système qui implémente la pagination à la demande. Ce système dispose de 4 cases de mémoire physique qui sont toutes occupées, à un instant donné, avec des pages de mémoire virtuelle.

La table suivante donne, pour chaque case mémoire:

- ✓ Le moment du chargement de la page qu'elle contient ($T_{\text{chargement}}$)
- ✓ Le temps du dernier accès à cette page ($T_{\text{dernier accès}}$)
- ✓ L'état de la page par des bits : référencé (R), modifié (M) et présence (P).

Les temps sont donnés en tops d'horloge.

Case	$T_{\text{chargement}}$	$T_{\text{dernier accès}}$	R	M	P
0	126	270	0	0	1
1	230	255	1	0	1
2	110	260	1	1	1
3	180	275	1	1	1

Travail demandé :

Indiquez quelle est la page qui sera remplacée en cas d'un défaut de page si l'algorithme de remplacement de page est :

- a. LRU
- b. FIFO
- c. Horloge (Seconde chance)

% Bon travail %

Examen Session Principale

Niveau d'étude : L1-SI	Documents : Non autorisés
Matière : Fondement de réseaux	Nombre de pages : 2
Enseignant : Ben Othman Soufiene	Date : 17/05/2022

Exercice 1 : (5 points)

1. A quoi servent un ETTD et un ETCD ?
2. Quelles sont les différences entre une transmission parallèle et une transmission série ?
Quelle est celle qui est la plus utilisée ?
3. Quelles sont les différences entre les communications Simplex, Half Duplex et Full Duplex ?
4. Quelles sont les différences entre une transmission asynchrone et une transmission synchrone ?
5. Préciser à quelle couche OSI appartient chacun les termes suivants : PPP, Routeur, Hub, UDP, RJ45, Fanion, Switch, Numéro de port.

Exercice 2 : (5 points)

1. Rappelez les principes des codages suivants : NRZ, RZ, Bipolaire, Manchester (ou biphase), Manchester différentiel, puis Miller.
2. Représentez le signal binaire 0100 0010 1000 0111 en bande de base codé selon les codes NRZ, RZ, Bipolaire, Manchester (ou biphase), Manchester différentiel, puis Miller.

Exercice 3 : (5 points)

Une entreprise a demandé et reçu l'adresse réseau 185.177.0.0.

1. Déterminer la classe de ce réseau ?
2. Quel(s) octet(s) représente (nt) la portion réseau d'une adresse IP de cette classe ?
3. Quel(s) octet(s) représente (nt) la portion hôte d'une adresse IP de cette classe ?
4. Quel est le masque par défaut de sous-réseau ?
5. Déterminer le nombre maximal de machines que nous pouvons l'avoir dans ce réseau ?

Exercice 4 : (5 points)

On considère le réseau d'adresse 194.168.1.0.

1. Déterminer, par calcul, le masque par défaut et l'adresse de diffusion.
2. Combien peut-on adresser de composants (équipements adressables) dans ce réseau ?

L'administrateur décide d'utiliser le masque suivant : 255.255.255.192 pour créer des sous-réseaux.

3. Combien peut-il en créer avec ce masque ?
4. Donner les adresses de ces sous-réseaux.
5. Déduire les adresses de diffusion.
6. A quel sous-réseau appartient l'imprimante d'adresse 194.168.1.130 ?

On applique maintenant le masque de réseau 255.255.255.224 pour créer des sous-réseaux ;

7. Combien peut-on créer de sous-réseaux avec ce masque ?
8. Combien d'adresses disponibles dans chaque sous-réseau ?
9. Déterminer l'adresse du sous-réseau de la machine d'IP 194.168.1.70.
10. Quelle est l'adresse de diffusion du sous-réseau de la question précédente ?

Session de Mai 2022
Epreuve d'Algèbre II

Exercice Soit $A = \begin{pmatrix} 2 & -1 & 1 \\ 1 & 0 & -1 \\ 2 & -2 & 1 \end{pmatrix}$

1. (a) Déterminer $P_A(X)$ le polynôme caractéristique de A .
 (b) Vérifier que 1 est une racine de $P_A(X)$.
 (c) Déduire les valeurs propres de A ainsi que les sous-espaces propres.
 (d) A est-elle diagonalisable ? Justifier votre réponse.
2. Soit $P = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$.
 - (a) Montrer que P est inversible et calculer P^{-1} .
 - (b) Vérifier que $P^{-1}AP = D$ avec $D = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 3 \end{pmatrix}$.
3. (a) Déterminer, sans calculer A^4 , les valeurs propres de A^4 .
 (b) A^4 est-elle diagonalisable ? Si oui, donner une base B de \mathbb{R}^3 formée par des vecteurs propres de A^4 .
4. Après avoir donné D^n , calculer A^n pour tout $n \in \mathbb{N}$.
5. On donne les suites réelles $(a_n)_n$, $(b_n)_n$ et $(c_n)_n$ définies par $a_0 = c_0 = 1$,

$$b_0 = -1 \text{ et pour } n \geq 0 \quad \left\{ \begin{array}{l} a_{n+1} = 2a_n - b_n + c_n \\ b_{n+1} = a_n - c_n \\ c_{n+1} = 2a_n - 2b_n + c_n \end{array} \right.,$$

et on pose $X_n = \begin{pmatrix} a_n \\ b_n \\ c_n \end{pmatrix}$.

- (a) Montrer que, pour $n \geq 0$, $X_{n+1} = AX_n$ puis que $X_n = A^nX_0$.
 (b) En déduire, pour $n \geq 0$, a_n , b_n et c_n en fonction de n .



Session de Mai 2022
Epreuve d'Analyse II

Exercice 1 (8pts) Soit $(a_n)_n$ la suite de nombres réels définie par $a_0 = 2$, et la relation de récurrence $a_{n+1} = a_n^2 - a_n + 1$, $n \geq 0$.

1. (a) Calculer a_1 et a_2 .
(b) Montrer que la suite $(a_n)_n$ est croissante.
(c) En déduire que $\lim_{n \rightarrow +\infty} a_n = +\infty$.
2. On considère la série $\sum_{n \geq 0} b_n$ avec $b_n = \frac{1}{a_n}$.
(a) En utilisant le critère de d'Alembert, justifier la convergence de cette série.
(b) Montrer que pour tout $k \in \mathbb{N}$, on a : $b_k = \frac{1}{1 - a_{k+1}} - \frac{1}{1 - a_k}$.
(c) En déduire que $\sum_{n=0}^{+\infty} b_n = 1$.

Exercice 2 (4pts) Déterminer, en justifiant votre réponse, le rayon de convergence des séries entières suivantes:

1. $\sum_{n \geq 0} \left(\frac{n}{n+1} \right)^{n^2} x^n;$
2. $\sum_{n \geq 0} \frac{n+1}{2^{2n}} x^{2n}$.

Exercice 3 (8pts) On se propose de développer en série entière la fonction définie par :

$$f(x) = \ln(x^2 - 4x + 3)$$

1. Donner les ensembles de définition et de dérivabilité de f .
2. Montrer que $\forall x \in \mathbb{R}$, $f'(x) = \frac{1}{x-1} + \frac{1}{x-3}$.
3. Sachant que $\forall x \in]-1, 1[$, $\frac{1}{1-x} = \sum_{n=0}^{+\infty} x^n$, donner un développement en série entière de f' au voisinage de 0.
4. En déduire le développement de f en série entière au voisinage de 0 et préciser le rayon de convergence R .

X



Examen Algorithmique et Structure de données

Durée : 1 h 30 mn

Documents non autorisés

Exercice 1

On peut implémenter une pile à l'aide d'un tableau T dont on connaît la taille maximale (Max). Un indicateur (indSommet) indique à chaque fois la position actuelle du sommet de la pile. La pile peut être donc déclarée comme suit :

```

Const
    Max=100
Type
    Pile= enregistrement
        T : Tableau[1..Max] d'entier
        indSommet: entier
    Fin enregistrement
Var
    P: Pile

```

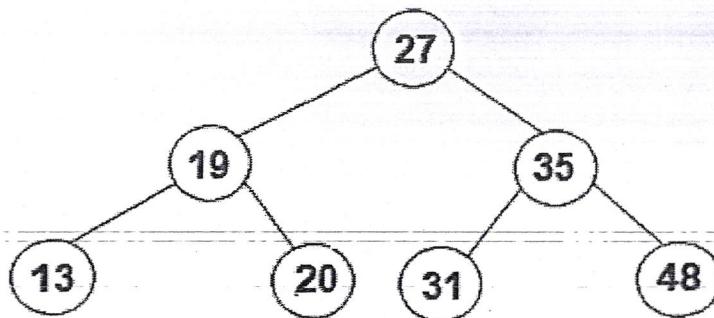
- 1)** Ecrire une procédure **Empiler(x : Entier ; var P : Pile)** qui permet d'ajouter l'élément x au sommet de la pile.
- 2)** Ecrire une procédure **Dépiler(var P : Pile)** qui permet de supprimer le sommet de la pile.

Exercice 2

- 1)** Dessiner l'arbre binaire de recherche obtenu par insertions successives des éléments de clés **30, 22, 38, 51, 17, 23, 34** en partant de l'arbre vide.
- 2)** Écrire une fonction qui permet d'afficher les valeurs des nœuds d'un arbre binaire de recherche par ordre croissant.

Exercice 3

Soit l'arbre binaire de recherche suivant :



- a) Donner le résultat obtenu en utilisant le parcours suivant (**Parcours1**) de l'arbre:

```

Procédure Parcours1(B : Arbre)
Début
  Si B ≠ nil Alors
    Parcours1(B^.FilsG)
    Parcours1(B^.FilsD)
    Ecrire(B^.val)
  Fin Si
Fin
  
```

- b) Écrire une fonction **EstVide(B :Arbre) : Booléen** qui renvoie vrai si l'arbre est vide et faux si non.
- c) Écrire une fonction **EstUneFeuille(B :Arbre) : Booléen** qui renvoie vrai si le nœud de l'arbre n'admet aucun fils et faux si non.



X
Le poly de Fondements des Bases de Données (éventuellement manuscrit) est non autorisé. Tout autre document est exclu. Le barème est donné à titre indicatif, et il est susceptible d'être modifié. Tout résultat déjà établi en cours peut être cité sans besoin de le redémontrer. Les exercices sont indépendants les uns des autres.

EXERCICE N°1

1. Lequel des énoncés suivants est l'ordre d'occurrence correct dans une instruction SQL typique ?

- A. select, group by, where, having
- B. select, where, group by, having
- C. select, where, having, group by select, having, where, group by
- D. select, having, where, group by

2. Quelle est la différence entre une clé primaire et une clé unique ?

- A. La clé primaire ne peut pas être une variable de date alors que la clé unique peut être
- B. Vous ne pouvez avoir qu'une seule clé primaire alors que vous pouvez avoir plusieurs clés uniques
- C. La clé primaire peut prendre des valeurs nulles mais la clé unique ne peut pas prendre de valeurs nulles
- D. Aucun de ces

3. Parmi les affirmations suivantes, lesquelles sont vraies pour UPDATE en SQL ?

- A. Vous ne pouvez mettre à jour qu'une seule table à l'aide de la commande UPDATE
- B. Vous pouvez mettre à jour plusieurs tables à l'aide de la commande UPDATE
- C. Dans la commande UPDATE, vous devez lister les colonnes à mettre à jour avec leurs nouvelles valeurs (séparées par des virgules).
- D. Pour mettre à jour plusieurs enregistrements ciblés, vous devez spécifier la commande UPDATE à l'aide de la clause WHERE.

4. Laquelle des affirmations suivantes est correcte à propos de la commande 'CREATE TABLE' lors de la création d'une table ?

- A. Nous devons attribuer un type de données à chaque colonne
- B. Nous avons de la flexibilité dans SQL. Nous pouvons attribuer un type de données à la colonne même après avoir créé une table
- C. Il est obligatoire d'insérer au moins une seule ligne lors de la création d'un tableau
- D. Aucun de ces

5. Lesquels des énoncés suivants sont les synonymes de « colonne » et « ligne » d'un tableau ?

- A. Ligne = [Tuple, enregistrement]
- B. Colonne = [Champ, Attribut]
- C. Ligne = [Tuple, Attribut]
- D. Colonnes = [Champ, enregistrement]

6. Supposons qu'on vous donne une table/relation « EMPLOYÉ » qui comporte deux colonnes (« Name » et « Salary »). La colonne Salaire de cette table contient des valeurs NULL. Maintenant, je veux découvrir les enregistrements qui ont des valeurs nulles.

Name	Salary
Saurav	Null
Ankit	1000
Falzan	2000
Sunil	3000
Kunal	4000

Quel sera le résultat des requêtes suivantes ?

Query 1. SELECT * FROM EMPLOYEE WHERE Salary <> NULL;

Query 2. SELECT * FROM EMPLOYEE WHERE Salary = NULL;

- A. La requête 1 donnera les 4 dernières lignes en sortie (à l'exclusion de la valeur nulle)
- B. La requête 2 donnera la première ligne en sortie (seul enregistrement contenant une valeur nulle)
- C. La requête 1 et la requête 2 donneront toutes les deux le même résultat
- D. Je ne peux pas dire

7. Les tableaux A, B ont trois colonnes (à savoir : 'id', 'age', 'name') chacun. Ces tables n'ont pas de valeurs "nulles" et il y a 100 enregistrements dans chacune des tables.

Voici deux requêtes basées sur ces deux tables « A » et « B » :

Query1: `SELECT A.id FROM A WHERE A.age > ALL (SELECT B.age FROM B WHERE B.name = 'Ankit')`

Query2: `SELECT A.id FROM A WHERE A.age > ANY (SELECT B.age FROM B WHERE B.name = 'Ankit')`

Maintenant, laquelle des déclarations suivantes est correcte pour la sortie de chaque requête ?

- A. Le nombre de tuples dans la sortie de la requête 1 sera supérieur ou égal à la sortie de la requête 2
- B. Le nombre de tuples dans la sortie de la requête 1 sera égal à la sortie de la requête 2
- C. Le nombre de tuples dans la sortie Query 1 sera inférieur ou égal à la sortie de Query 2
- D. Je ne peux pas dire

8. Qu'est-ce qui est vrai à propos de la relation (tableau) sous différentes formes normales (1NF, 2NF, 3NF) ?

- A. Si une relation satisfait les conditions de 1NF. Il satisfera automatiquement aux conditions de 2NF
- B. Si une relation satisfait les conditions de 2NF. Il satisfera automatiquement aux conditions de 1NF
- C. Si une relation satisfait les conditions de 3NF. Il satisfera automatiquement aux conditions de 1NF
- D. Si une relation satisfait les conditions de 2NF. Il satisfera automatiquement aux conditions du 3NF

Sélectionnez la bonne option :

9. Considérons une relation R avec le schéma R (A, B, C, D, E, F) avec un ensemble de dépendances fonctionnelles F comme suit :

{AB->C, BC->AD, D->E, CF->B}

Lequel des éléments suivants sera le résultat de DA+ ?

Remarque : Pour tout X, X+ est la fermeture de X.

- A. AD
- B. DAE
- C. ABCD
- D. ABCDEF

Tableau : AV1

Name	Salary	Company	Designation
Saurav	1000	AV1	Junior Data Scientist
Ankit	800	AV1	Data Scientist
Sunil	1200	AV2	Senior Manager
Kunal	1400	AV2	CEO
Deepak	1100	AV3	Data Entry Operator
Swati	1200	AV3	BDE
Faizan	900	AV1	Deep Learning Expert

Les questions suivantes sont basées sur le tableau ci-dessus.

10. Quelle sera la sortie de la requête suivante ?

Query 1: `Select name from AV1 where name like '%a%`'

- A. Saurav, Ankit, Kunal, Deepak, Swati, Faizan
- B. Saurav, Kunal, Deepak, Swati, Faizan
- C. Kunal, Deepak, Swati, Faizan
- D. Aucune des choses ci-dessus

11. Quelle sera la sortie de la requête ci-dessous ?

Query: `SELECT Name from AV1 where Name LIKE '%_____%`;

Remarque : L'opération ci-dessus contient 6 traits de soulignement ('_') utilisés avec l'opérateur LIKE.

- A. Il renverra les noms où le nombre de caractères dans les noms est supérieur ou égal à 6
- B. Il renverra les noms où le nombre de caractères dans les noms est supérieur à 6
- C. Il renverra les noms où le nombre de caractères dans les noms est inférieur ou égal à 6
- D. Cela donnera une erreur

12. Considérez le schéma relationnel suivant.

Students (rollno: integer, sname: string)

Courses (courseno: integer, cname: string)

Registration (rollno: integer, courseno: integer, percent: real)

Maintenant, laquelle des requêtes suivantes serait capable de trouver les noms uniques de tous les étudiants ayant obtenu un score supérieur à 90 % dans le cours no 107 ?

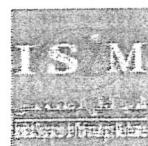
- A. SÉLECTIONNER DISTINCT S.sname FROM Students as S, Registration as R WHERE R.rollno=S.rollno AND R.courseno=107 AND R.percent >90
- B. SÉLECTIONNER UNIQUE S.sname FROM Étudiants comme S, Inscription comme R O WH R.rollno=S.rollno AND R.courseno=107 AND R.percent >90
- C. SELECTIONNER sname FROM Étudiants comme S, Inscription comme R O WH R.rollno=S.rollno AND R.courseno=107 AND R.percent >90
- D. Aucun de ces

EXERCICE N°2

On considère une entreprise de ventes de voitures. Un modèle de voiture est décrit par une marque, une dénomination. Une voiture est identifiée par un numéro de série, et a un modèle, une couleur et un prix d'achat et un cout (prix auquel la voiture est revenue). Des clients, on connaît le nom, le prénom et l'adresse. Parmi les clients, on trouve les anciens propriétaires des voitures d'occasion, ainsi que les personnes ayant acheté une voiture au magasin. Lorsqu'une vente est réalisée, on en connaît le vendeur (dont on connaît le nom, le prénom, l'adresse et le salaire fixe) et le prix d'achat réel (en tenant compte d'un rabais éventuel). Chaque vendeur touche une prime de 5% de la différence entre le prix d'achat et le coût de la voiture. L'entreprise est repartie sur un certain nombre de magasins et chaque vendeur opère dans un magasin unique. Chaque voiture est, ou a été, stockée dans certains magasins et est vendue dans le dernier magasin où elle a été stockée. On garde trace des dates d'arrivée dans et de départ des magasins. Un transfert de voiture entre deux magasins se fait dans la journée.

1. Donner un diagramme Entité/Association pour représenter ces données
2. Donner un schéma de base de données correspondant à ce diagramme
3. Ecrire les requêtes suivantes en SQL
 - i. Afficher la liste des voitures (numéro) vendues après le 15 avril 2007.
 - ii. Afficher la voiture qui rapporte le plus d'argent.
 - iii. Afficher Les bénéfices de chaque magasin pour le mois de janvier 2007.
 - iv. Afficher le vendeur ayant accordé le plus gros rabais.
 - v. Afficher le meilleur client (celui ayant rapporté le plus d'argent à l'entreprise).





Matière : Programmation Python

Niveau/Section : L1/GLSI

Examen (Session Principale)

Date : Mai 2022

Enseignante : Hajar Triki

Durée : 1h30mns

L'épreuve contient 4 pages

Les documents ne sont pas autorisés

Les réponses non justifiées ne seront pas prises en compte

OCM : (4 points)

**Dans le cadre du langage de programmation PYTHON, choisir la (ou les) bonne(s) réponse(s) et
donner un exemple justificatif**

1. Les listes sont :

- a. des collections d'objets qui peuvent être de types différents.
- b. des collections d'objets qui doivent être de même types.
- c. modifiables.
- d. non modifiables

2. Un tuple est :

- a. un tableau d'objets qui peuvent être de types différents.
- b. un tableau d'objets qui doivent être de même types.
- c. modifiable.
- d. non modifiable.

3. Dans un dictionnaire:

- a. La valeur peut être de n'importe quel type et la clé doit être de type non modifiable.
- b. La clé peut être de n'importe quel type et la valeur doit être de type non modifiable.

4. Python est :

- a. un langage interprété.
- b. un langage compilé.
- c. différentie entre les structures de données par la déclaration.
- d. différentie entre les structures de données par l'utilisation .

Exercice 2 : (4 points)

Soit le programme Python tkinter suivant :

```
import tkinter
fen = tkinter.Tk()
fen.geometry ("800x700")

def action (event):
    lTuples=[]
    T = entiers_saisis.get()
    L=T.split(',')
    print(L)
    for n in L:
        for m in L:
            if int(m) + int(n) == 10:
                lTuples.append((int(m),int(n)))
    tuples.insert(0.0,lTuples)

lbl1 = tkinter.Label (fen, text = "Enter your list of integers :")
lbl1.place (x = 50, y = 20)

entiers_saisis= tkinter.Entry (fen)
entiers_saisis.place (x = 250, y = 20)

entiers_saisis.bind('<KeyPress-T>',action)

lbl2 = tkinter.Label (fen, text = "the list of tuples = 10 : ")
lbl2.place (x = 50, y = 100)
tuples = tkinter.Text (fen)
tuples.place (x = 250, y = 100)

fen.mainloop ()
```

dans le programme ci-dessus

1. Délimitez les différentes étapes de création d'une application graphique.
2. Quel est le rôle de l'instruction : `entiers_saisis.bind('<KeyPress-T>',action)`
3. Donnez le résultat d'exécution de ce programme.

Exercice 3 : (12 points)

On veut calculer les moyennes générales de n étudiants à partir des notes attribués dans 4 matières pour chacun d'eux. Ces notes concernent celles du DS, TP et Examen.

Etant donné un **dictionnaire python** représentant l'évaluation dans une matière dont les clés sont les noms des étudiants et les valeurs sont les listes des notes obtenues comme le montre l'exemple suivant :

```
matièreM= {"Ahmed": [12, 15, 17], "Mohamed": [15, 13, 16], "Raed": [13, 14, 15] }
```

On demande de :

1. Ecrire une fonction **evalMatiere** qui prend comme argument un entier n représentant le nombre des étudiants. Cette fonction permet de saisir, pour chaque étudiant, son nom, les 3 notes (DS,TP,Examen) sous la forme d'une liste et de retourner un dictionnaire dont les clés sont les noms des étudiants et les valeurs sont les listes des notes obtenues.
2. Ecrire une fonction **moyenne** en python qui prend comme argument une liste de 3 nombres réels et retourne leur moyenne en suivant la formule suivante:

$$20\% \text{premier nombre} + 10\% \text{deuxième nombre} + 70\% \text{troisième nombre}$$

3. Ecrire une fonction **moyenneM** en python qui prend comme argument un dictionnaire représentant l'évaluation d'une matière et qui retourne un dictionnaire dont les clés sont les noms des étudiants et les valeurs sont leurs moyennes dans la matière.

4. Ecrire une fonction `moyenneG` en python qui prend comme arguments 4 dictionnaires de moyennes de 4 matières différentes et retourne un dictionnaire dont les clés sont les noms des étudiants et les valeurs sont leurs moyennes générales qui se calculent de la manière suivante :

Somme des moyennes d'un étudiant dans les quatres matières/4

5. Ecrire une fonction `etudiantAdmis` qui retourne un dictionnaire dont les clés sont les étudiants admis et les valeurs des clés sont les moyennes obtenues (moyenne supérieures ou égales à 10).

6. Ecrire une fonction `etudiantNonAdmis` qui retourne un dictionnaire dont les clés sont les étudiants non admis et les valeurs des clés sont les moyennes obtenues (moyenne inférieures ou égales à 10).

7. Ecrire un programme principal qui permet de saisir le nombre des étudiants et de tester ces fonctions.

Examen [Session Principale (Semestre 2)]

Matière : Atelier de programmation 2

Niveau d'Etude : LSI 1

Enseignant: Mme Ibtissem Laouer

Nombre de pages : 3

Durée : 1 H 30

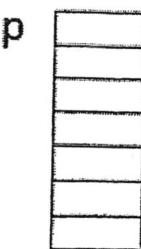
Documents : Non autorisés

Exercice 1 :

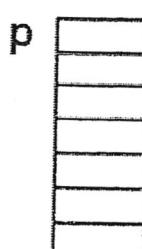
Donnez les états successifs de la pile p dans la séquence suivante :

```
int main()
{
    int v;
    Pile *p= Initialiser();
    Empiler(p,13);
    Empiler (p,5);
    Empiler (p,10);
    Empiler (p,8);
    afficher(p);           //état 1
    v=Depiler(p);
    v=Depiler(p);
    afficher(p);           //état 2
    return 0 ;
}
```

État 1 :



État 2 :



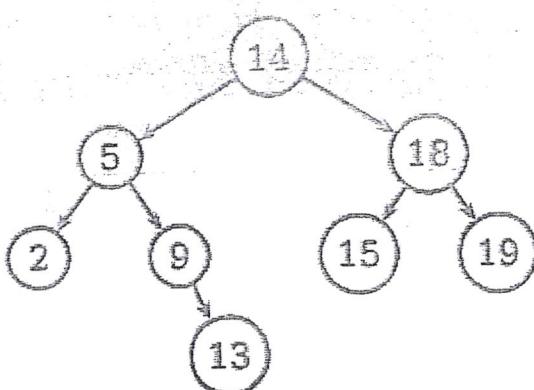
Exercice 2 :

On veut créer une liste des documents d'une imprimante. Cette liste contient des documents à imprimer sachant qu'un document est caractérisé par :

- Un nom
 - Le nombre de pages
1. Définir la structure Document.
 2. Définir la structure Liste représentant une liste des documents à imprimer.
 3. Ecrire la fonction Ajouter qui ajoute un document à la fin de la liste.

Exercice 3 :

Soit l'arbre binaire de recherche suivant :



Soit la structure C suivante qui représente chaque nœud de l'arbre:

```
typedef struct Noeud Noeud;
struct Noeud{
    int valeur;
    Noeud *gauche;
    Noeud *droite;
} T;
```

1. Écrire une fonction récursive qui affiche toutes les feuilles de l'arbre T.
2. Écrire une fonction récursive qui affiche l'arbre T en ordre croissant.
3. Donner le résultat de l'arbre après l'insertion d'un nœud contenant la valeur 11

Exercice 4 :

Soit la structure C suivante qui représente chaque élément de la file:

```
typedef struct Element Element;
struct Element{
    int valeur;
    Element *suivant;
};
```

On se donne trois files F1, F2 et F3. La file F1 contient des nombres entiers positifs. Les files F2 et F3 sont initialement vides. En n'utilisant que ces trois :

1. Écrire la fonction void multiple5(Element *F1, Element *F2) qui place tous les entiers de F1 qui sont des multiples de 5 dans F2.
2. Écrire la fonction void copier (Element *F1, Element *F2) qui copie le contenu de F1 dans F2. Le contenu de F1 après l'appel de la fonction doit être identique à celui avant l'appel.
3. Écrire la fonction void pairImpair(Element *F1, Element *F2, Element *F3) qui place les entiers pairs de F1 dans F2 et les entiers impairs de F1 dans F3.
4. Ecrire un programme C permettant de :
 - Initialiser la file F1, F2 et F3.
 - Remplir la file F1 par n éléments, sachant que n est un entier strictement positif saisi au clavier.
 - Afficher F1.
 - Appeler la fonction multiple5 et afficher la file F2.
 - Appeler la fonction copier et afficher F2.
 - Appeler la fonction pairImpair et afficher F2 et F3.

Indication : on suppose que les fonctions : Element *InitialiserFile (), Element *enfiler(Element *, int), Element *defiler(Element *, int) et void afficher(Element *) sont déjà implémentées.

Bon travail

