



Systèmes d'Exploitation II

Examen Principal

(Session Mai 2024)

Exercice 1 : Interblocage [7 Pts]

On considère un système dans lequel quatre processus P1, P2, P3 et P4 doivent partager quatre ressources notées Ra, Rb, Rc et Rd. On suppose que le nombre total d'instances de chaque ressource est :

Ressources système

Ra	Rb	Rc	Rd
8	8	10	10

L'état courant du système est décrit par les instances de ressources dont il dispose (Ressources allouées) chaque processus et son besoin en ressources maximales pour qu'il puisse s'exécuter :

Ressources allouées

-	Ra	Rb	Rc	Rd
P1	2	3	2	2
P2	1	0	2	2
P3	1	0	2	3
P4	2	3	3	1

Ressources maximales

-	Ra	Rb	Rc	Rd
P1	3	4	2	3
P2	7	6	4	8
P3	5	0	5	8
P4	5	5	6	2

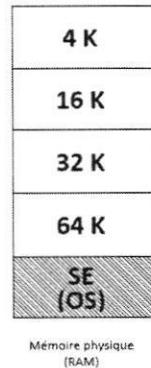
Travail à faire :

1. Calculer les ressources disponibles du système à l'état courant.
2. Calculer les ressources demandées par chaque processus pour terminer son exécution.
3. Utiliser l'algorithme de banquier pour répondre à la question suivante : L'état courant est-il sûr ou risqué ?
4. Le processus P1 demande une ressource supplémentaire de type Rd. Cette demande devrait-elle être satisfaite ?
5. Le processus P2 demande une ressource supplémentaire de type Rc. Cette demande devrait-elle être satisfaite ?
6. Le processus P3 demande deux ressources supplémentaires de type Rd. Cette demande devrait-elle être satisfaite ?

Exercice 2 : Mémoire Centrale [7 Pts]

On considère un système disposant de 160 Ko de mémoire physique (RAM), avec une partition résidente du système d'exploitation (OS) et des partitions fixes réparties comme suit :

- Une Partition réservée au SE (44 Ko),
- Quatre partitions mises à la disposition des utilisateurs, sont initialement vides (64 Ko, 32 Ko, 16 Ko et 4 Ko).



On suppose qu'un processus chargé en mémoire restera jusqu'à la fin de son exécution.

Les processus arrivent dans le système comme le montre le tableau suivant :

Processus	Moment d'arrivée	Taille	Temps CPU
P ₀	0	3 Ko	2 ms
P ₁	0	11 Ko	4 ms
P ₂	0	42 Ko	7 ms
P ₃	1	5 Ko	3 ms
P ₄	1	4 Ko	2 ms
P ₅	2	22 Ko	5 ms
P ₆	2	58 Ko	9 ms
P ₇	5	30 Ko	6 ms
P ₈	8	10 Ko	4 ms
P ₉	10	60 Ko	10 ms

Travail à faire :

1. Si chaque partition dispose de sa propre file d'attente, donner les états d'occupation de la mémoire aux différentes étapes de traitement de ces processus.
2. On suppose maintenant que toutes les partitions utilisent une seule file d'attente commune avec un mode d'allocation basé sur l'algorithme « Meilleur ajustement » (Best Fit). Donner les états d'occupation de la mémoire aux différentes étapes de traitement de ces processus.

Exercice 3 : Mémoire virtuelle [6 Pts]

Soit un système dont l'espace d'adressage de la mémoire virtuelle utilise la technique de la pagination. Pendant l'exécution d'un processus P, la suite des adresses virtuelles (pages) suivantes est appelée :

0, 1, 2, 1, 3, 4, 2, 4, 2, 3, 4, 5, 1, 0, 2, 3, 2, 5.

On suppose que la mémoire physique est initialement vide et que le processus P utilise pendant son execution 4 cases mémoires (cadres/frames).

Travail à faire :

1. Faites l'analyse des défauts de pages pour les deux algorithmes suivants et dire Lequel semble le meilleur ? Justifier !
 - a. LRU,
 - b. LFU.
2. Calculer le nombre minimal de défauts de page possible pour le processus P ?

% Bon travail %



EXAMEN

Section: L1SI

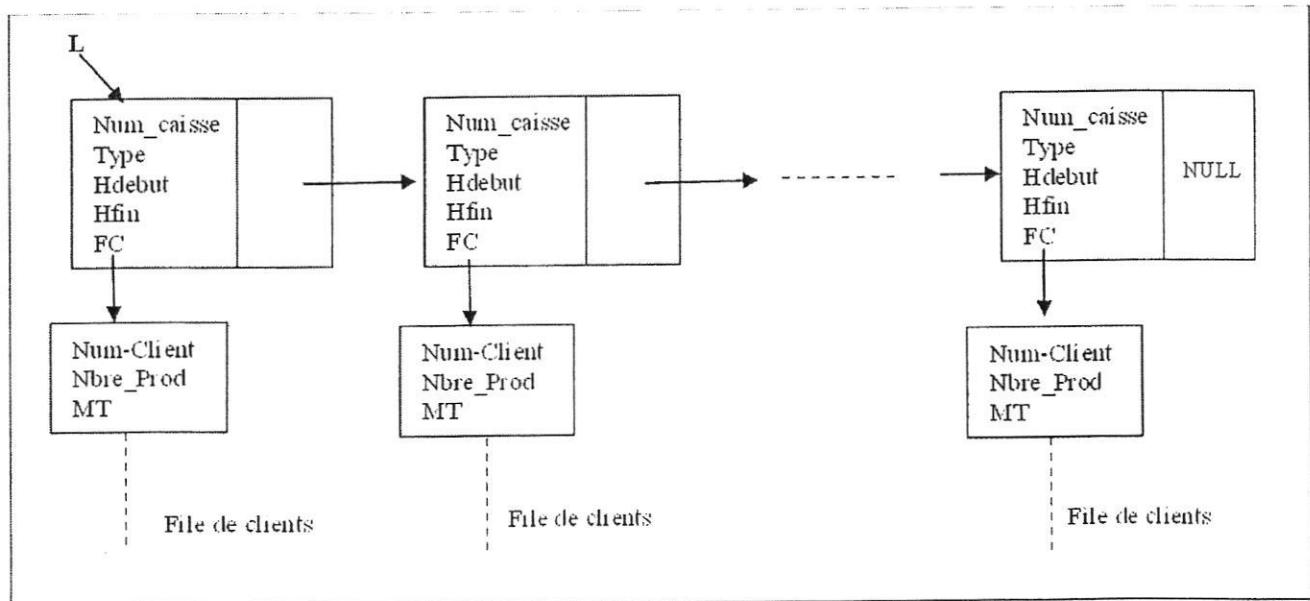
Épreuve d': Algorithmique, structures de données et Complexité

Nature de l'épreuve:	DS <input type="checkbox"/>	EF <input checked="" type="checkbox"/>	Documents:	autorisés <input type="checkbox"/>	non autorisés <input checked="" type="checkbox"/>
Date de l'épreuve:	08/05/2024		Session:	principale <input checked="" type="checkbox"/>	contrôle <input type="checkbox"/>
Durée de l'épreuve:	1H30		Enseignant:	Fethi Mguis	

N.B : L'examen comporte deux pages.

Exercice 1 (12pts)

Un super marché souhaite gérer ses caisses. Une caisse est caractérisée par un numéro, le type (une caisse pour moins de 10 articles ou une caisse normale), heure de mise en service et heure de fin de service. Pour chaque caisse, on lui associe une file d'attente de clients ayant acheté des produits. On définit un client par un numéro d'ordre, le nombre de produits achetés et le montant total payé. L'ensemble des caisses est représenté par une liste simplement chaînée comme l'illustre la figure suivante :



Considérons les types de structures suivantes :

Horaire : enregistrement

heure : entier

minute : entier

Fin Enreg

Client : enregistrement

Num_client : entier //le numéro du client

Nbre_Prod : entier //le nombre de produits

MT : réel //le montant total payé

Fin Enreg

Caisse : enregistrement

Num_Caisse : entier

Type : chaîne

Hdebut : Horaire

Hfin : Horaire

FC : File_Client

Fin Enreg

Cellule : Enregistrement

val : Caisse

*suiv : *cellule*

Fin Enreg

Liste : Enregistrement

*Tête : *Cellule*

Fin Enreg

Le type File_Client représente une file de clients.

Dans la suite, nous considérons données les primitives permettant la manipulation d'une file de clients (Init_file, File_vide, Défiler, Enfiler et Sommet_file). Développer les fonctions suivantes :

1. Une fonction **nbre_caisse_en_service** (L :Liste, h :Horaire) : entier qui calcule le nombre de caisses en service, dans la liste L, à un horaire h donné.
2. Une fonction **nbre_client_caisse** (F :File_Client) :entier qui permet de retourner le nombre de clients existant dans la file F.
3. Une fonction **nbre_client_Total** (L :Liste) : entier qui permet de retourner le nombre de clients total présents devant les caisses. Pour ce faire, utiliser la fonction précédente.
4. Une fonction **plus_grand_CA**(L :Liste) : *Cellule qui retourne un pointeur sur la cellule dont la file de clients possède le maximum de nombre total de produits. Le nombre total de produits d'une file correspond à la somme des nombres de produits achetés par les clients dans cette file.

Exercice 2 (4pts)

Soit $P = (a_1, \dots, a_n)$ une pile non vide de n entiers (avec a_1 le sommet de la pile). On désire modifier le contenu de la pile P pour qu'elle contienne (a_2, \dots, a_n, a_1) (c'est-à-dire le sommet a_1 est déplacé au fond de la même pile). Cette opération s'appelle la rotation d'une pile.

1. Définir les structures nécessaires pour résoudre le problème.

2. Ecrire la procédure **Rotation(P : *Pile)**.

N.B : Vous pouvez appeler les primitives Init_Pile, Pile_Vide, Empiler, Dépiler et Sommet_Pile sans les ré-implémenter.

Exercice 3 (4pts)

Soit n le nombre de nœuds d'un arbre binaire, f son nombre de feuilles et h sa hauteur.

1. Quelle est la hauteur maximale d'un arbre binaire à n nœuds ?
2. Quel est le nombre maximal de feuilles d'un arbre binaire de hauteur h ?

Bon Travail

Session de Mai 2024
Epreuve d'Analyse II

Documents et calculatrices : Non autorisés

Exercice 1 (8pts) Soit $(u_n)_{n \geq 0}$ la suite réelle définie par :

$$u_0 = \frac{1}{3} \text{ et pour } n \geq 0, u_{n+1} = u_n - u_n^2.$$

1. Calculer u_1 et u_2 .
2. Montrer que $\forall n \in \mathbb{N} : u_n > 0$.
3. (a) Démontrer que la suite $(u_n)_{n \geq 0}$ est décroissante.
(b) En déduire que la suite $(u_n)_{n \geq 0}$ est convergente et déterminer sa limite.
4. Déterminer, en justifiant votre réponse, la nature des séries numériques

$$\sum_{n \geq 0} u_n^2 \text{ et } \sum_{n \geq 0} (-1)^n u_n.$$

Exercice 2 (6pts) Déterminer, en justifiant votre réponse, le rayon de convergence des séries entières suivantes:

1. $\sum_{n \geq 1} n \ln(n) x^n$.
2. $\sum_{n \geq 0} \frac{(n!)^2}{(2n+1)!} x^n$.
3. $\sum_{n \geq 0} \left(\frac{n^2 + 1}{(2n+1)^2} \right)^{3n} x^{2n+1}$.

Exercice 3 (6pts) Sachant que

$$\forall x \in]-1, 1[, \ln(1+x) = \sum_{n=1}^{+\infty} (-1)^{n+1} \frac{x^n}{n} \text{ et } \forall x \in \mathbb{R}, e^x = \sum_{n=0}^{+\infty} \frac{x^n}{n!},$$

déterminer le rayon de convergence R des séries entières réelles suivantes, puis calculer leurs sommes sur $] -R, R[$.

1. $\sum_{n \geq 0} \frac{2n-1}{n!} x^n$.
2. $\sum_{n \geq 1} \frac{(-1)^{n+1}}{n(n+1)} x^n$.

EXAMEN SESSION PRINCIPALE FONDEMENTS DES BD SECTION L1SI 1, 2, 3, 4 & 5 Durée 1h30 ANNEE UNIVERSITAIRE : 2023-2024 Documents non autorisés : Tous documents du cours/td/tp, notes manuscrites (nb : pas de livres)		PROF M ^e J.RAOUF
---	--	-----------------------------

EXERCICE N°1 QCM

N.B

Choisir la ou les bonnes réponses

Recopier sur la feuille de l'examen le tableau suivant

Compléter le tableau en choisissant la ou les bonnes réponses

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
A.									
B.									
C.									
D.									

1. Marier les différents types de requête à leur utilité

- | | |
|-----------------|-------------------|
| A. Sélection | 1. Requête Delete |
| B. Insertion | 2. Requête Update |
| C. Suppression | 3. Requête Select |
| D. Modification | 4. Requête Insert |

2. Lequel des opérateurs suivants est utilisé pour comparer une valeur à une liste de valeurs littérales qui ont été spécifiées ?

- A. ALL
- B. IN
- C. BETWEEN
- D. ANY

3. Quelle est l'utilité de COUNT en SQL ?

- A. Retourne le nombre de valeurs distinctes
- B. Renvoie le nombre de groupes
- C. Retourne le nombre total de valeurs
- D. Renvoie le nombre de colonnes

4. Quelle instruction apporte des modifications aux attributs de la base de données ?

- A. CHANGE
- B. ALTER
- C. ALTERNATE
- D. UPDATE

5. Est-ce que « Datetime » et « Timestamp » sont de même type de données ?

- A. Oui
- B. Non
- C. Dépend de SGBD
- D. Aucune de ces réponses

6. Quelle commande est utilisée pour afficher toutes les tables existantes dans une base de données ?

- A. SHOW TABLES
- B. SHOW TABLE
- C. SHOW
- D. Aucun de ces réponses

7. Quelle ligne suivante produira une erreur ?

- A. SELECT * FROM personne WHERE personne_id=1;
- B. SELECT nom FROM personne;
- C. SELECT nom, prenom, personne_id FROM personne;
- D. Aucun de ces réponses

8. Quelle clé est utilisée pour lier deux tables dans SQL?

- A. Clé primaire
- B. Clé étrangère
- C. Clé primaire et étrangère
- D. Aucun de ces réponses

9. Trouver l'erreur ?

```
CREATE TABLE personne (
personne_id SMALLINT,
nom VARCHAR,
PRENOM VARCHAR,
date_naissance DATE
CONSTRAINT pk_personne PRIMARY KEY (personne_id)
);
INSERT INTO
personne( personne_id, nom, PRENOM, Date_naissance )
VALUES(1,'a', 'b', 09-1994-02);
```

- A. Erreur dans les valeurs de données
- B. Aucune erreur
- C. Toute autre erreur
- D. Aucun de ces réponses

10. Quelle est l'utilité de l'option NOT NULL placée après le type de donnée d'un champ ?

- A. Interdire les espaces dans un champ
- B. Obliger l'entrée de contenu dans un champ
- C. Interdire la valeur 0 dans un champ
- D. Permettre l'absence de contenu dans un champ

EXERCICE N°2

On utilise le schéma relationnel suivant pour stocker des informations concernant des droits d'accès à des tables d'une base de données

- Utilisateur (id, nom, prenom)
- id : clé d'Utilisateur, nombre entier
- nom de l'utilisateur, chaîne de caractères
- prénom de l'utilisateur, chaîne de caractères
- Table (nomTable, # idCreateur)
- nomTable: clé de Table, chaîne de caractères
- idCreateur: identifiant du créateur de la table
- Privilège (id, # nomTable, # idUtilisateur, droit)
- id: clé de privilège, nombre entier
- nomTable: identifiant de table chaîne de caractères
- idUtilisateur : identifiant d'utilisateur
- droit ∈ {SELECT, INSERT, UPDATE, DELETE, ALL}

1. Décrire en SQL la création des tables de cette base de données en exprimant toutes les contraintes d'intégrité mentionnées ci-dessus.
2. Ecrire une requête SQL permettant de lister les noms et prénoms des utilisateurs ayant le droit de détruire la table 'Marques'
3. Donnez les noms des tables pour lesquelles le créateur de la table n'a pas le droit 'DELETE'.
4. Quelles sont les noms des tables sur lesquelles au moins 4 utilisateurs ont le droit 'ALL'?
5. Quels sont les utilisateurs (donner leur nom et leur prénom) ayant le droit 'ALL' sur toutes les tables ?

EXERCICE N°3

On considère le Schéma de la base de données CINEMA :

- FILM (NUMF, TITRE, GENRE, ANNEE, DUREE, BUDGET, REALISATEUR, SALAIRE REAL)
- PERSONNE (NUMP, PRENOM, NOM, DATENAIS)
- ACTEUR (NUMA, AGENT, SPECIALITE, TAILLE, POIDS)
- DISTRIBUTION (NUMF, NUMA, ROLE, SALAIRE)

L'attribut REALISATEUR de la relation FILM est l'identifiant d'une PERSONNE. Il en est de même pour les attributs NUMA et AGENT de la relation ACTEUR.

Donner les requêtes SQL permettant de répondre aux questions suivantes. Lorsque cela est possible, on donnera également les requêtes équivalentes en calcul relationnel de n-uplets (tuples), puis on donnera un plan d'exécution en utilisant l'optimisation à base de règles.

1. Retrouver la liste de tous les films.
2. Retrouver la liste des films dont la longueur dépasse 180 min.
3. Donner la liste de tous les genres de film.
4. Donner le nombre de films par genre.
5. Trouver le/les titre(s) et l'/les années(s) du/des film(s) le(s) plus long(s).
6. Trouver tous les "couples d'acteurs", i.e., les acteurs ayant joués le "Premier" rôle dans un même film (sans doublons).
7. Trouver le nom des personnes qui ne sont ni agents, ni acteurs et ni réalisateurs.
8. Donner le nom et le prénom des réalisateurs qui ont joué dans au moins un de leurs propres films
9. Quel est le total des salaires des acteurs du film "Nuits blanches à Seattle".
10. Pour chaque film de Spielberg (titre, année), donner le total des salaires des acteurs.

Matière :	ELECTRONIQUE ANALOGIQUE
Enseignant :	Kamel JARRAY
Filière :	Classe LITIC
Durée :	90 min
Documents :	Non autorisés

Exercice 1 :

Soit le quadripôle passif ayant la structure en π (figure 1).

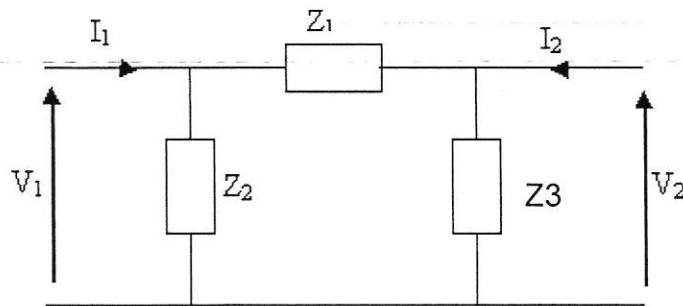


Figure 1

A) Déterminer :

1. l'impédance d'entrée du quadripôle chargé sur une impédance Z_u quelconque.
2. l'impédance de sortie du quadripôle alimenté en entrée par un générateur d'impédance interne Z_g .

B) On caractérise ce quadripôle par ses fonctions de transfert définies par :

$$V_1 = Z_{11}I_1 + Z_{12}I_2$$

$$V_2 = Z_{21}I_1 + Z_{22}I_2$$

1- Calculer en fonction de Z_1 , Z_2 et Z_3 les coefficients Z_{11}, Z_{12}, Z_{21} et Z_{22}

2- Le quadripôle est utilisé pour l'adaptation d'une charge résistive R_u à un générateur de tension de résistance interne R_g .

Etablir la relation qui doit lier les éléments du montage pour qu'il y ait adaptation d'impédance.

Exercice 2 :

On se propose d'étudier le montage de la figure 2. L'amplificateur opérationnel supposé idéal, et il fonctionne en régime linéaire. Il est alimenté sous $\pm 15V$.

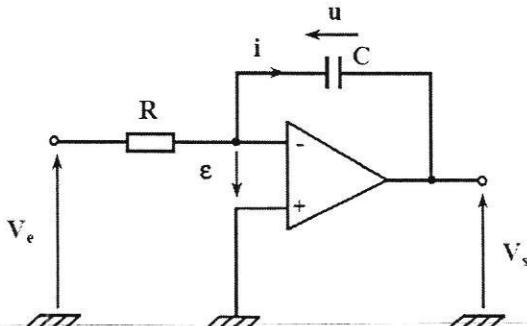


Figure 2

- 1) Quelle est la fonction réalisée par ce montage.
- 2) A l'aide du théorème de Millmann, donner l'expression de la tension V_- sur l'entrée négative de l'AOP.
- 3) Quelle est la valeur de la tension ϵ ?
- 4) Exprimer le courant i en fonction de V_e et R .
- 5) Exprimer i en fonction de C et $\frac{du}{dt}$ puis en fonction de C et $\frac{dVs}{dt}$.
- 6) A partir des expressions précédentes, montrer que l'on peut écrire : $\frac{dVs}{dt} = -\frac{Ve}{RC}$
- 7) On suppose que la tension ve , strictement inférieur à zéro volt. Dans quel sens évolue V_s en fonction du temps ?
- 8) On suppose que la tension ve , strictement supérieur à zéro volt. Dans quel sens évolue V_s en fonction du temps ?
- 9) La tension Ve est une tension périodique alternative sinusoïdale. La fréquence de cette tension est de 1 kHz et son amplitude est 5 V.

D'autre part, $R = 10 \text{ k}\Omega$ et $C = 10 \text{ nF}$.

Représenter les courbes de $ve(t)$ et $vs(t)$.

Bon Courage

Examen Session Principale	
Niveau d'étude : L1-SI	Documents : Non autorisés
Matière : Fondement de réseaux	Nombre de pages : 2
Enseignant : Ben Othman Soufiene	Date : 06/05/2024

Exercice 1 : (5 points)

Soit la séquence binaire suivante : 100111101

1. Codez la séquence envoyée en bande de base par codage Manchester et codage Miller.
2. Pour l'envoyer dans un réseau étendu, codez la séquence précédente en large bande : utilisant la modulation d'amplitude à 2 niveaux (3v, 6v), de fréquences à 2 fréquences (f_1, f_2), et de phase à 2 phases ($\pi/2, 3\pi/2$).

Exercice 2 : (5 points)

Une connexion numérique est établie entre 02 points A et B de la terre via un satellite géostationnaire située à 25000 km de A et 32000 km de B. Un signal est émis depuis A avec un débit de transmission de 64 Kbits/sec et la vitesse de propagation est égale à 300000 km/sec avec un temps de traitement au niveau du satellite de 0.5 ms.

1. Quel est le temps total de transfert d'un paquet de 512 octets.
2. Sachant que la station A désire envoyer vers B une chaîne d'informations de la taille =32Kbits. Calculer le temps T2 total de transfert de cette chaîne.

Exercice 3 : (5 points)

Une machine utilise l'adresse IP 141.16.112.31 et le masque 255.255.224.0

1. Quel est la classe de ce réseau.
2. Donner l'adresse de sous réseau auquel appartient cette machine.
3. Combien de machines peuvent être utilisées dans ce réseau.
4. Donner les adresses la plus basses et la plus haute.
5. Donner l'adresse de diffusion.

Exercice 4 : (5 points)

Une entreprise dispose de l'adresse IP publique 172.150.0.0/16 pour son réseau principal. Elle décide de diviser ce réseau en plusieurs sous-réseaux pour ses différents départements. Voici le nombre des postes dans chaque département :

- ❖ Département des ventes : 50 postes de travail.
- ❖ Département du développement : 70 postes de travail.
- ❖ Département du marketing : 30 postes de travail.

1. Déterminez le nombre de bits de sous-réseau nécessaires pour satisfaire les besoins de l'entreprise.
2. Calculez le masque de sous-réseau.
3. Allouer des sous-réseaux pour chaque département.
4. Proposer une configuration d'adresses IP pour chaque sous-réseau en fournissant les plages d'adresses IP pour les postes de travail de chaque département.

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
 Université de Gabès
 Institut Supérieur de l'Informatique de Médenine

Filière: L1SI

Date: Mai 2024

Nbre de pages : 1

Session Principale

Durée : 1h30mn

Examen : Algèbre II

NB : Il sera tenu compte de la présentation des copies et de la bonne rédaction.

Exercice 01

Soient $A = \begin{pmatrix} -7 & 0 & 6 \\ -8 & 1 & 6 \\ -12 & 0 & 10 \end{pmatrix}$, $B = (e_1, e_2, e_3)$ la base canonique de \mathbb{R}^3 et f

l'endomorphisme de \mathbb{R}^3 de matrice A par rapport à B .

- 1) Déterminer le rang de A .
- 2) a) Déterminer le polynôme caractéristique de A
 b) En déduire que 1 et 2 sont les valeurs propres de A
- 3) Déterminer les sous espaces propres de A et leur dimension. En déduire que A est diagonalisable.
- 4) Soit P la matrice de colonnes $u = (3, 2, 4)$, $v = (0, 1, 0)$ et $w = (-2, -2, -3)$.
 - a) Calculer P^2 . En déduire que P est inversible et donner P^{-1} (sans calcul).
 - b) En déduire que $S = (u, v, w)$ est une base de \mathbb{R}^3 .
- 5) Montrer que S est formée de vecteurs propres de A et donner la matrice D de f par rapport à la base S .

Exercice 02

Soit $\mathcal{B}_c = (e_1, e_2, e_3)$ la base canonique de \mathbb{R}^3 .

$f : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ une application linéaire définie dans la base \mathcal{B}_c par la matrice :

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

1. Calculer $\det(A)$, En déduire le rang de f .
2. Montrer que $\text{Ker}(f) \neq \{0\}$.
 On définit les trois vecteurs u , v et w par :
 $A(u) = 0$; $(A - I_3)(v) = 0$ et $(A - 2I_3)(w) = 0$
3. Trouver ces trois vecteurs.
4. Montrer que $B = (u, v, w)$ est une base de \mathbb{R}^3 .
5. En déduire la matrice D de f dans la base B .

Bon travail



Matière : Programmation Python

Niveau/Section :L1/SI

Examen (Session Principale)

Date : 10 Mai2024

Enseignante : Hajar Triki

Durée :1h30mns

-
- L'épreuve contient 4 pages
 - Les documents ne sont pas autorisés
 - Justifiez vos réponses pour qu'elles soient prises en compte
-

Partie I : OCM : (5 points)

**Dans le cadre du langage de programmation PYTHON, choisir, en justifiant, la (ou les) bonne(s)
réponse(s)**

1. Python est :

- a. un langage compilé.
- b. un langage interprété.
- c. un langage à typage dynamique.

2. Les tuples sont :

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

3. Dans un dictionnaire :

- a. la clé peut être de type simple, de type str, de type tuple ou de type frozen-set.
- b. la clé ne doit être ni de type liste, ni de type set ni de type dictionnaire.
- c. la clé peut être d'un type modifiable.
- d. la clé doit être d'un type non modifiable.

4. Une application graphique :

- a. ne supporte aucun conteneur graphique.
- b. peut avoir plusieurs conteneurs graphiques.
- c. ne peut contenir qu'un seul conteneur graphique.

5. Un gestionnaire d'évènement :

- a. est unique dans une application graphique.
- b. peut être lié à n'importe quel objet graphique.
- c. est lié toujours à un seul type des objets graphiques.

Partie II : Analyse d'un code Python : (5points)

Soit le programme **Python** suivant :

```
from tkinter import *

fen = Tk()
fen.geometry("600x150")
fen.title("Est-il premier ?")

def action(event):

    L_div=[]
    nb_div=0
    i=2
    N=int(saisie_number.get())
    while i<N:

        if N%i==0:
            res['text']='nombre non premier'
            break
        i+=1
    else:
        res['text']='nombre premier'

lbl = Label (fen, text = "Entrer un nombre:")
lbl.place (x = 50, y = 20)
saisie_number= Entry (fen)
saisie_number.place (x = 300, y = 20)

saisie_number.bind('<KeyPress-P>' , action)

res = Label (fen, text = "nombre non premier")
res.place (x = 50, y = 50)

fen.mainloop ()
```

1. Délimitez les étapes de création d'une application graphique dans ce programme.
2. Comment définir l'interactivité d'une application python.
3. Détaillez le rôle de l'instruction : `saisie_number.bind('<KeyPress- P>',action)` .
4. Donnez le résultat d'exécution de ce programme.

Partie III : Analyse de données avec Python: (10 points)

On veut réaliser une analyse des données météorologiques de **n** régions réparties dans le monde.

On considère, dans ce cadre, un **dictionnaire** python appelé **Meteo_Data** comportant ces données.

- Les clés de ce dictionnaire sont les régions. Chaque région est représentée par le tuple (**pays, ville**).
- Les valeurs sont des dictionnaires appelés **data** dont les clés sont les paramètres de mesure météorologiques suivants :
 - **TA** : température de l'air : **chaine de caractères**
 - **Vent** : sous la forme d'un **tuple (vitesse, direction)**
 - **CC** : couverture nuageuse : **nombre réel**

Exemple :

```
Meteo_Data={('Tunisie','Medenine'):{'TA':'28°C','vent':('20km/h','nord'),'cc':15.3},('Algérie','Constance'):{'TA':'18°C','vent':('5km/h','ouest'),'cc':20.5},('Egypte','Caire'):{'TA':'36°C','vent':('10km/h','est')},'cc': 0.3}}
```

On demande d'écrire un programme Python permettant de :

1. Saisir le nombre des régions **n**.
2. Construire le dictionnaire **Meteo_Data** par **n** données météorologiques.
3. Afficher la ville ayant la température la plus élevée.
4. Construire un dictionnaire **Meteo**, à partir du dictionnaire **Meteo_Data**, indiquant la météo de chaque ville selon sa couverture nuageuse dont :

- *Les clés : les tuples (ville,pays)*
- *Les valeurs : 'nuageux' si 'cc' >= 70.0
'partiellement couvert' si 10.0 < 'cc' < 70.0
'ensoleillé' si 'cc' <= 10*

5. Trouver la **liste** des villes dont l'état de la météo est 'nuageux'.
6. Afficher un message d'alerte d'une tempête pour chaque ville du dictionnaire **Meteo_Data**
dont la vitesse de vent dépasse 90 km/h. Dans chaque message, on doit indiquer le nom de la ville et la direction du vent sous la forme :
nom_ville, Attention! TEMPETE! parvenue du, **direction_du_vent**

😊 Bon Travail 😊

Université de Gabes
Institut Supérieur de l'Informatique de Médenine
Examen (session principale)



Matière: Atelier programmation 2
Enseignants: Mme Ibtissem Laouer & Mme Dhaya Nafkha
Niveau/Section : L1SI

Documents: Non autorisés
Durée: 1H30
Date: Mai 2024

Exercice 1 :

Soit P un pointeur qui pointe sur un tableau A :

```
int A[] = {10, 23, 13, 4, 56, 67, 78, 8, 9};  
int *P;  
P = A ;
```

Quelles valeurs ou adresses fournissent ces expressions ?

1. *P+2
2. *(P+2)
3. &A [4] -3
4. A+3
5. P+(*P-8)
6. *(P+*(P+8) -A [7])

Exercice 2 :

Quelle structure de données (liste, pile, file) choisir pour chacune des tâches suivantes :

- ❖ Représenter un répertoire téléphonique (nom, adresse et téléphone).
- ❖ Envoyer des fichiers à un serveur d'impression. (Un serveur qui permet de partager une ou plusieurs imprimantes entre plusieurs utilisateurs (ou ordinateurs) situés sur un même réseau).
- ❖ Gérer dans un logiciel les commandes Annuler et répéter

Exercice 3 :

Soit la structure C suivante qui représente chaque élément de la liste chaînée :

```
Typedef struct Element Element ;  
struct Element{  
    int val;  
    Element *suiv ;  
};
```

Soit L une liste chaînée d'entiers non vide.

1. Écrire une fonction *int Dernier (Element *L)* qui renvoie le contenu de dernier élément de la liste.
2. Écrire une fonction *void Permuter (Element *L)* qui permute le contenu du premier et le dernier élément de la liste.

Exercice 4 :

- A. Si l'on suppose que les opérations suivantes sont effectuées sur deux piles P1 et P2 contenant des caractères, qu'est-ce qui sera affiché à l'écran ?

```
char ch;  
ElementP *P1=NULL, *P2=NULL;  
P1=Empiler(P1, '6');  
P1=Depiler(P1, &ch); printf("%c\n", ch); //ligne 1  
P2=Empiler(P2, ch);  
P2=Empiler(P2, '+');  
P2=Empiler(P2, '3');  
P1=Empiler(P1, '8');  
P2=Depiler(P2, &ch); printf("%c\n", ch); //ligne 2  
P2=Empiler(P2, ch);  
P2=Depiler(P2, &ch); printf("%c\n", ch); //ligne 3  
P1=Empiler(P1, ch);  
P2=Depiler(P2, &ch); printf("%c\n", ch); //ligne 4  
P1=Empiler(P1, ch);  
P2=Depiler(P2, &ch);  
P1=Empiler(P1, ch);  
while (P1 != NULL)  
{ P1=Depiler(P1, &ch); printf("%c", ch); } //ligne 5
```

Si l'on suppose que les opérations suivantes sont effectuées sur une file F contenant des caractères. Donnez les états successifs de la file dans la séquence suivante :

```
ElementF *F=NULL;  
char ch;  
F=Enfiler(F,'A');  
F=Enfiler(F,'B');  
F=Enfiler(F,'C');      //état 1  
F=Defiler(F,&ch);      //état 2  
F=Enfiler(F,'a');      //état 3  
F=Defiler(F,&ch);      //état 4  
F=Enfiler(F,'b');  
F=Enfiler(F,'C');      //état 5
```

Exercice 5 :

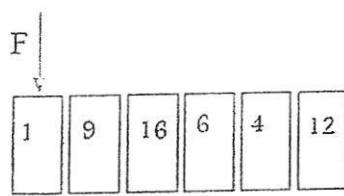
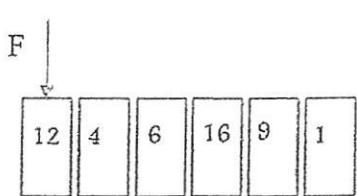
Trouver les manipulations à faire pour implémenter les fonctions ci-dessous, on doit utiliser dans chaque cas une ou plusieurs structures de données intermédiaires (pile ou file) :

Indication : on suppose que les fonctions :

```
ElementP *Empiler(ElementP *p,int v),  
ElementP *Depiler(ElementP *p, int *v),  
ElementF *Enfiler(ElementF *f,int v) et  
ElementF *Defiler(ElementF *f, int *v) sont déjà implémentées.
```

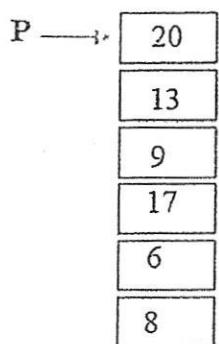
1. La fonction *ElementF *InverserFile(ElementF *F)* permet d'inverser l'ordre des éléments de la file F.

Exemple :

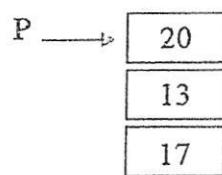


2. La fonction $\text{ElementP} * \text{Supprimer}(\text{ElementP} *P, \text{int } x)$ permet de supprimer tous les éléments de la Pile P inférieur à une valeur donnée x, en conservant les éléments dans le même ordre.

Exemple : $x = 10$



Etat initial



Etat final