

Examen de la session principale en
Algorithmique Avancée & Complexité
8 janvier 2025

Classe : 2IngInfo_{A,B,C,D,E}

Durée : 1^h30'

Nombre de pages : 3 pages

Il est conseillé de lire attentivement les énoncés avant de se plonger sur la feuille des réponses. Les copies propres et bien soignées sont très appréciées. Il sera tenu compte de la lisibilité des réponses. Le barème donné est indicatif. Aucun document n'est autorisé.

Bonne chance !

Problème I. Ordonnancement optimal d'une chaîne de montage

Un constructeur automobile possède un atelier avec deux chaînes de montage comportant chacune n postes de montages. Chaque véhicule doit passer par les n postes dans l'ordre. Le constructeur cherche à déterminer quels sont les postes à sélectionner sur la chaîne 1 et sur la chaîne 2 pour minimiser le délai de transit d'une voiture à travers l'atelier. Les données du problème d'optimisation qu'il doit résoudre sont les suivantes.

- Pour $i = 1, 2$ et $j = 1, \dots, n$, on note $S_{i,j}$ le j -ème poste de la chaîne i ,
- e_i le temps d'entrée d'un véhicule sur la chaîne i ,
- $a_{i,j}$ le temps de montage pour le poste j sur la chaîne i ,
- $t_{i,j}$ le temps de transfert d'un véhicule de la chaîne i vers l'autre chaîne après le poste $S_{i,j}$,
- et finalement x_i le temps de sortie d'un véhicule de la chaîne i (voir Figure 1).

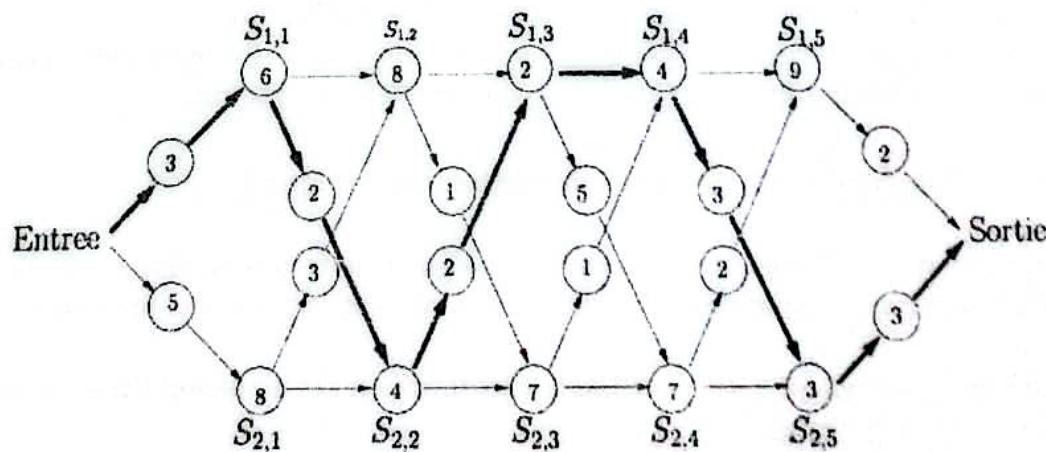


FIGURE 1 – Ordonnancement d'une chaîne de montage

Partie A. Caractérisation d'un ordonnancement optimal (3 pts)

En résumé, chaque véhicule doit passer les n postes de montage qui sont répartis sur les deux chaînes. La solution de ce problème d'optimisation est définie par une séquence $S = \{S_{i_1,1}, S_{i_2,2}, S_{i_3,3}, \dots, S_{i_n,n}\}$, $i_j \in \{1, 2\}$.

- Combien d'itinéraires possibles peut-on avoir entre l'entrée et la sortie ? La première étape consiste à identifier des sous-problèmes dont les solutions optimales vont nous permettre de reconstituer une solution optimale du problème initial. Les sous-problèmes à considérer ici consistent à calculer un itinéraire optimal depuis poste $S_{i,j}$ pour $i = 1, 2$ et $j = 1, \dots, n$ jusqu'à la sortie.

Par exemple, considérons un itinéraire optimal depuis poste $S_{1,j}$ jusqu'à la sortie de l'atelier.

Si $j = n$, il n'y a qu'un seul chemin possible.

Pour $j = n-1, \dots, 1$, il y a deux possibilités. On passe par le poste $S_{1,j}$ et on continue sur un itinéraire optimal depuis $S_{1,j+1}$, ou bien on bascule de chaîne et on prend le chemin optimal $S_{2,j+1}$ jusqu'à la sortie.

ou bien,
on passe par le poste $S_{2,j}$ et on continue sur un itinéraire optimal depuis $S_{2,j+1}$, ou bien
on bascule de chaîne et on prend le chemin optimal $S_{1,j+1}$ jusqu'à la sortie.

Soit f la fonction objectif à minimiser qui décrit le temps passé par un véhicule dans l'atelier de montage.

- Justifier la récurrence suivante :

$$f(i, j) = \begin{cases} a_{i,n} + x_i & j = n; \\ \min(e_1 + f(1, 1), e_2 + f(2, 1)) & j = 0; \\ a_{1,j} + \min(f(1, j+1), t_{1,j} + f(2, j+1)) & i = 1; \\ a_{2,j} + \min(t_{2,j} + f(1, j+1), f(2, j+1)) & i = 2; \end{cases}$$

Partie B. Approche récursive (6 pts)

- Écrire une fonction récursive qui calcule le temps minimal passé par un véhicule dans l'atelier de montage.
- Énoncer la récurrence qui régit la complexité de cette fonction.
- Calculer la complexité de cette fonction en développant cette récurrence. Donner une estimation asymptotique en notation \mathcal{O} de cette complexité.

Partie C. Résolution par la programmation dynamique (11 pts)

L'inconvénient avec la version récursive est que les mêmes sous-problèmes sont traités plusieurs fois. Il serait plus judicieux de sauvegarder les solutions des sous-problèmes dans un tableau.

- Montrer à travers un petit exemple la superposition des sous-problèmes (C'est-à-dire les mêmes sous-problèmes sont rencontrés plusieurs fois).

L'autre condition d'applicabilité de la programmation dynamique est la sous-structure optimale de la solution.

7. Expliquer cette propriété et dire comment notre problème satisfait cette propriété.
8. Quelle autre propriété doit satisfaire un problème pour qu'une approche gloutonne permette de lui trouver une solution optimale ? Est-ce que ce problème satisfait cette propriété ?
 - Si oui, comment ?
 - si non, construire un petit contre-exemple.

Nous considérons le tableau F de dimension $(2) \times (n + 1)$, où les $F[i, j]$ désignent le temps minimal nécessaire pour traverser l'atelier depuis le poste $S_{i,j}$ jusqu'à la sortie.

		Les postes					
		0	S_1	S_2	S_3	...	S_n
Les chaînes	1						
	2						

Nous pouvons immédiatement calculer la dernière colonne $F[1, n]$ et $F[2, n]$. Par la suite, on peut facilement déduire $F[1, n - 1]$ et $F[2, n - 1]$. . . Ainsi de suite jusqu'à calcul de $F[1, 0]$ et $F[2, 0]$.

La solution à notre problème est la valeur minimale entre $F[1, 0]$ et $F[2, 0]$.

9. Écrire la fonction itérative qui retourne la durée minimale de la traversée de l'atelier ;
10. Calculer la complexité de cette fonction ;
11. On peut maintenant déterminer les postes qui permettent d'avoir le temps de montage minimal, en reprenant le tableau F depuis le début ($i = 0$) jusqu'à la fin ($i = n$) et en identifiant les postes qui ont menés à la solution optimale.
Écrire une procédure qui affiche la séquence des postes qui mènent à l'ordonnancement optimal.

Année universitaire 2024-2025

Examen

Matière : Techniques de recherche d'emploi

2ème année - GSIL - Méca - Info - GSI

Mercredi 8 janvier 2025

Durée : 90 mn

I. Lisez la transcription de l'entretien d'embauche répondez aux questions :

10 points

1. Identifiez cinq réponses et comportements positifs du candidat. Justifiez.
2. Identifiez une réponse qui pourrait être améliorée. Justifiez.
3. a. Le candidat a donné deux réponses maladroites. Relevez-lez et dites pourquoi elles pourraient donner une mauvaise impression au recruteur.
3. b. Proposez des réponses alternatives qui corrigeraient ces maladresses.

Recruteur : Bonjour, merci d'être avec nous aujourd'hui. Pouvez-vous vous présenter brièvement ?

Candidat : Bonjour, merci pour cette opportunité. Je m'appelle Monia ZITOUNI, et je suis diplômée en logistique et supply chain de l'Université de Liège. J'ai effectué un stage de six mois dans une entreprise de transport où j'ai travaillé sur l'optimisation des flux.

Recruteur : Pourquoi avez-vous choisi ce domaine ?

Candidat : Honnêtement, je n'avais pas d'idée précise après le lycée, alors j'ai suivi cette filière parce qu'elle semblait avoir des débouchés.

Recruteur : Qu'est-ce qui vous a motivé à postuler pour ce poste ?

Candidat : Votre entreprise est connue pour son expertise en logistique internationale. Je souhaite débuter ma carrière dans une structure qui me permettrait d'apprendre rapidement et de contribuer à des projets d'envergure.

Recruteur : Pouvez-vous me parler d'une expérience pratique durant vos études ou stages ?

Candidat : Lors de mon stage, j'ai analysé les itinéraires de livraison et proposé une nouvelle organisation qui a permis de réduire les coûts de transport de 8 %.

Recruteur : Avez-vous rencontré des difficultés dans ce projet ?

Candidat : Oui, il a été difficile de convaincre certains conducteurs d'adopter les nouveaux itinéraires. J'ai organisé des réunions pour expliquer les bénéfices et répondre à leurs questions, ce qui a permis une meilleure adhésion.

Recruteur : Quelles compétences pensez-vous pouvoir apporter à ce poste ?

Candidat : Je maîtrise des outils comme Excel avancé et des logiciels de gestion logistique comme SAP. Je suis aussi capable d'analyser des données pour proposer des solutions efficaces.

Recruteur : Quelle est votre plus grande faiblesse ?

Candidat : Franchement, je n'aime pas trop qu'on me donne des ordres. Je préfère travailler à ma façon, même si ça peut créer des tensions.

Recruteur : Où vous voyez-vous dans cinq ans ?

Candidat : J'aimerais évoluer vers un poste de responsable logistique, où je pourrais superviser des projets complexes et développer mes compétences en management.

Recruteur : Que savez-vous de notre entreprise ?
Candidat : Je sais que vous êtes un acteur majeur de la logistique internationale, avec une forte

présence en Europe et en Asie. J'ai également lu que vous investissez dans des solutions de traçabilité, ce qui est très innovant.

Recruteur : Avez-vous des exemples de réussite dans votre parcours ?
Candidat : Pendant mes études, j'ai travaillé sur un projet académique où nous devions simuler

l'organisation logistique d'un entrepôt. Mon équipe a proposé un système qui a obtenu la meilleure note grâce à son efficacité et son originalité.

Recruteur : Comment gérez-vous le stress ?
Candidat : Je prends le temps de planifier mes tâches et je priorise les plus importantes. Pendant

mon stage, j'ai appris à rester calme en cas de pression et à demander de l'aide si nécessaire.

Recruteur : Avez-vous des questions pour nous ?
Candidat : Oui, pourriez-vous m'en dire plus sur les opportunités de formation que vous proposez

pour les jeunes recrues ?

Recruteur : Bien sûr, nous proposons un programme de formation continue pour accompagner nos collaborateurs. Avez-vous d'autres questions ?
Candidat : Non, je pense avoir une bonne vision du poste et de l'entreprise. Merci beaucoup pour vos réponses détaillées.

Recruteur : Merci à vous pour cet échange. Avez-vous un dernier mot à ajouter ?

Candidat : Je tiens à vous remercier pour cette opportunité. Cet entretien a renforcé ma motivation à rejoindre votre entreprise et à contribuer à vos projets logistiques. Je reste à votre disposition pour toute information complémentaire.

Recruteur : Merci, nous reviendrons vers vous sous peu. Passez une bonne journée.

Candidat : Merci beaucoup, bonne journée à vous également.

II. Situation :

10 points

Vous avez été choisi(e) lors de la semaine d'intégration pour prendre la parole devant les nouveaux étudiants et parler de la recherche d'emploi et essentiellement de l'importance d'un bon CV.

Rédigez votre discours en 20 lignes environ dans lequel vous expliquez l'importance de cette intervention pour l'avenir professionnel de vos camarades. Votre discours devra inclure des conseils sur la rédaction d'un CV efficace et des mises en garde sur les erreurs à ne pas commettre. Concluez en encourageant vos camarades à participer aux différents ateliers de la journée et à bien se préparer aux techniques de recherche d'emploi pendant leur cursus.

AVIS IMPORTANT AUX ETUDIANTS

1. Chacune des feuilles de votre copie doit comporter une étiquette code à barres placée à l'endroit indiqué « coller ici votre code à barres ».
2. Une copie d'examen comporte une seule «feuille principale» et des «feuilles suites». Sur chacune de vos feuilles, le code à barres est obligatoire.
3. Cette feuille d'examen est strictement personnelle. Elle ne doit comporter aucun signe distinctif. Elle doit être écrite en noir et/ou bleu.
4. Le non respect de l'une de ces recommandations peut faire attribuer la note ZERO à l'épreuve.

NOTE

--

 Coller ici votre
code à barre

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20

00	25	50	75

Examen Session Principale
Matière : SGBD
Enseignantes : O.LAMOUCHI, S.SAIDI
Date : 10/01/2025
Filière / Classes : 2 ING INFO A, B, C et D.
Durée : 1h30
Barème indicatif : 5-7-8
Documents : Non autorisés.
Nbre. de pages : 08
Calculatrice : Non autorisée.

 Noms, prénoms et signature
de l'enseignant correcteur


Exercice 1: Cocher la ou les bonnes réponses :

1.1. Les vues permettent :

- Une écriture simplifiée de requêtes complexes
- D'affiner la gestion de la confidentialité des données.
- D'offrir aux utilisateurs une vision des données indépendante de la façon dont elles sont structurées dans la BD.
- D'offrir aux utilisateurs une vision des données dépendante de la façon dont elles sont structurées dans la BD.

1.2. Les index implicites sont créés :

- lorsque une clause de contrainte UNIQUE est spécifiée.
- lorsque une clause de contrainte FOREIGN KEY est spécifiée.
- lorsque une clause de contrainte PRIMARY KEY est spécifiée.
- lorsque une clause de contrainte CHECK est spécifiée.

1.3. Une séquence :

- peut être partagée par plusieurs utilisateurs.
- peut être partagée par plusieurs objets.
- ne peut plus être supprimée, une fois utilisée.
- peut être implicite ou explicite

1.4. Quels sont les rôles standard Oracle nécessaires pour un utilisateur qui a besoin de créer des Objets ?

- GRANT
- RESSOURCE
- DBA

1.5. Quel(s) attribut(s) du curseur est (sont) utilisé(s) pour savoir si le dernier ordre Fetch a permis de ramener une ligne d'information.

- %Found
- %NotFound
- %ISOpen
- %Rowcoun

1.6. Quel est ou quels sont les avantages de passer par un bloc PL/SQL plutôt que de choisir la solution purement SQL lors de l'exécution de plusieurs instructions LMD (Insert, Update et Delete).

- Il n'y a pas de différences.
- Les transactions ne sont pas gérées.
- Les exceptions peuvent être traitées.
- L'exécution est plus rapide.

1.7. Quelles sont les instructions qui sont possibles dans un bloc PL/SQL ?

- Select..into
- Insert, Update, Delete
- Rollback
- Create, Alter, Drop

1.8. Quel est l'exemple correct du littéral de type booléen ?

- NULL (a)
- TRUE (b)
- Les deux réponses a et b
- Aucune de ces réponses

1.9. Les curseurs implicites sont définis dans la section :

- BEGIN
- DECLARE
- END
- Aucune de ces réponses

1.10. Une exception de type TOO_MANY_ROWS est soulevée par Oracle pour indiquer :

- Que l'ordre Fetch renvoie plusieurs lignes.
- Qu'il faut utiliser un curseur
- Que l'ordre Select renvoie plusieurs lignes
- Que la variable possède une seule valeur

1.11. Dans quel(s) type(s) de déclencheurs est-il possible d'interroger la table sur laquelle est défini le déclencheur ?

- Before select
- Before Insert
- Before Update
- Before Delete

1.12. Les triggers de type INSTEAD OF sont utilisés:

- Aussi bien sur les vues que pour les tables.
- Pour les tables, elles permettent de proposer un traitement à la place d'un autre.
- Pour les vues, elles permettent surtout de modifier à travers des vues multi-tables.

1.13. Dans le dictionnaire de données, on peut stocker:

- Des procédures
- Des fonctions
- Des blocs anonymes
- Des déclencheurs

1.14. Comment pallier aux interblocages (deadlocks) de transactions?

- Avortement et re-soumission de transaction(s)
- Mise en attente et re-soumission de transaction(s)

1.15. Quels sont les niveaux d'isolation que le SGBD Oracle implante,

- Serializable
- Read Uncommitted
- Read Committed
- Repeatable Read

1.16. On parle de concurrence de programmes si

- Ils s'exécutent sur la même machine
- Ils communiquent avec le même serveur de données
- Ils peuvent échanger des messages

1.17. L'entre croisement dans une exécution concurrente signifie que

- Les opérations d'une transaction sont réordonnées pour être compatibles avec celles des autres transactions
- Les opérations d'une transaction alternent avec celles des autres transactions en fonction du moment où elles sont soumises
- Le serveur exécute complètement une transaction, avant de passer à la suivante

1.18. En travaillant en mode par défaut d'oracle, j'effectue une mise à jour d'un tuple t, et je n'ai pas encore validé ni annulé. Quelle affirmation est vraie parmi celles ci-dessous?

- Si je lis t je ne vois pas encore ma mise à jour
- Si une autre transaction lit t, elle ne voit pas encore ma mise à jour
- Les autres transactions peuvent modifier t

1.19. Une lecture est qualifiée de non reproductive

- Si on obtient des résultats différents pour une même requête effectuée à intervalles réguliers
- Si on obtient des résultats différents pour une même requête effectuée dans la même transaction
- S'il s'agit de lire dans une table temporaire

1.20. Une exécution concurrente est sérialisable si

- Elle se termine sans panne ni blocage ni rejet
- L'état de la base à la fin pourrait être obtenu par une exécution en série des transactions
- Les opérations de chaque transaction s'exécutent dans l'ordre où elles sont soumises

Coller ici votre
code à barre

Exercice 2 :

Soit le schéma relationnel Gest_Consultation (voir annexe page 04).

La procédure `p_Annulation_const` (`num_matricule`, `num_ss`, `date_cons`) ci-dessous permet l'annulation d'une consultation d'un patient avec un médecin à une date donnée.

```
CREATE OR REPLACE PROCEDURE p_Annulation_const (nummat
    medecin.num_matricule%TYPE, numss patient.num_ss%TYPE, datecons
    consultation.date_cons%TYPE) IS
    ErrdateCons EXCEPTION ;
```

```
BEGIN
```

```
/* la procédure P_Afficher_Info_cons va afficher le nom du médecin, sa
spécialité, le nom du patient et l'heure de la consultation. */
```

```
P_Afficher_Info_cons(nummat, numss, datecons) ;
```

```
/* Vérification de la date de la consultation*/
```

```
IF (date_cons <=SYSDATE) THEN
    RAISE ErrDateCons ;
END IF ;
```

```
/* Mise à jour de la consultation*/
```

```
UPDATE Consultation SET annulee :=TRUE
WHERE num_matricule= nummat AND num_ss = numss
AND date_cons = datecons ;
COMMIT ;
```

```
EXCEPTION
```

```
WHEN ErrdateCons THEN DBMS_OUTPUT.PUT_LINE('La date de la consultation doit être
supérieur ou égale à la date d''aujourd''hui!!!')
END p_Annulation_const ;
```

2.1. Définir la procédure `P_Afficher_Info_cons` qui va chercher et afficher le nom du médecin, sa spécialité, le nom du patient et l'heure de la consultation.



2.2. Créer un trigger qui va suite à l'annulation ou la modification de la date d'une consultation, ajouter dans la table **Consultation_annulee** les informations suivantes : l'identifiant du médecin, l'identifiant du patient, la date de la consultation initiale, l'heure de la consultation et si elle est annulée (L'attribut **annulee** passe à TRUE si c'est une annulation ou si c'est un changement de date).

Consultation_annulee (num_matricule, num_ss, date_cons, heure_cons, annulee)

Exercice 3 :

Soit H une exécution reçue par le serveur de données du système de gestion des consultations lors de l'ajout d'une nouvelle consultation.

H : r1(x), r2(x), w3(x), r1(y), r4(y), r4(z), w1(x), c1, w4(z), w3(y), w3(z), c4, c3, w2(z), r2(x), w2(t), c2

3.1. Identifier les transactions concurrentes de H.

3.2. Quels sont les anomalies produites suite à l'exécution des transactions T2 et T3 ? Il faut préciser la ou les propriétés ACID qui non pas étaient respectées.

O : r2(x), w3(x), w3(y), w3(z), c3, w2(z), r2(x), w2(t), c2

3.3. Déterminer en justifiant votre réponse si la procédure *p_Annulation_const* puisse provenir de l'une des transactions de H.

3.3. Vérifier si H est sérialisable en identifiant les conflits et en construisant le graphe de sérialisation.

3.4. Quelle est l'exécution obtenue par verrouillage à deux phases à partir de H?

ANNEXE

Soit le schéma relationnel **Gest_Consultation** suivant:

```

CREATE TABLE Medecin (num_matricule
NUMERIC(6) NOT NULL CONSTRAINT
medecin_id_pk PRIMARY KEY,
nom_med VARCHAR(20) NOT NULL,
prenom_med VARCHAR(20) NOT NULL,
tel_med NUMERIC(8),
adr_med VARCHAR(35),
specialite VARCHAR(25),
code_serv NUMERIC(7));

CREATE TABLE Service (code_serv
NUMERIC(7) NOT NULL CONSTRAINT
serv_id_pk PRIMARY KEY,
nom_serv VARCHAR(25)NOT NULL,
chefserv NUMERIC(6),
tel_serv NUMERIC(8),
CONSTRAINT service_id_fk FOREIGN KEY
(chefserv) REFERENCES medecin
(num_matricule));

ALTER TABLE medecin ADD CONSTRAINT
medecin_id_fk FOREIGN KEY (code_serv)
REFERENCES service (code_serv);

```

```

CREATE TABLE MedNonDispo(num_matricule
NUMERIC(6) NOT NULL,
dated DATE NOT NULL,
datef DATE NOT NULL,
CONSTRAINT mnond_numm_fk FOREIGN KEY
(num_matricule) REFERENCES medecin
(num_matricule),
CONSTRAINT mnond_id_pk PRIMARY KEY
(num_matricule,dated));

```

```

CREATE TABLE Patient (num_ss NUMERIC(7)
NOT NULL CONSTRAINT patient_id_pk
PRIMARY KEY,
nom_pat VARCHAR(25)NOT NULL,
prenom_pat VARCHAR(20) NOT NULL,
adr_pat VARCHAR(30),
tel_pat NUMERIC(8) NOT NULL,
date_nais DATE,
sex_pat CHAR(1),
CONSTRAINT pat_pm_ck CHECK (sex_pat
IN('f','m')));

```

```

CREATE TABLE Consultation (num_cons
NUMERIC(7) NOT NULL CONSTRAINT
cons_id_pk PRIMARY KEY,
num_matricule NUMERIC(6) NOT NULL,
num_ss NUMERIC(7) NOT NULL,
date_cons DATE,
heure_cons NUMERIC(4,2),
tarif NUMERIC(6,2),
diagnostique VARCHAR(60),
annulee BOOLEAN ;
CONSTRAINT cons_numm_fk FOREIGN KEY
(num_matricule) REFERENCES medecin
(num_matricule),
CONSTRAINT cons_numss_fk FOREIGN KEY
(num_ss) REFERENCES patient (num_ss));

```

Relation Service : code_serv : identifiant d'un service, nom_serv : le nom du service (pédiatrie, cardiologie,...). chefServ : l'identifiant du medecin chef du service.

Relation Medecin ; num_matricule : identifiant d'un medecin, nom_med : le nom du medecin, prenom_med : le prenom d'un medecin, adr_med : adresse postale complète d'un medecin, spécialité : la spécialité du medecin (cardiologue, généraliste, pédiatre, ...), code_serv : le code du service ou travaille le medecin et cette attribut est obligatoire.

Relation Patient ; num_ss : le numero de sécurité sociale du patient, nom_pat : le nom du patient, prenom_pat : prénom du patient, adr_pat : adresse complète du patient, date_nais : la date de naissance du patient, sexe_pat : le sexe du patient ('F', 'M').

Relation Consultation : num_cons : identifiant d'une consultation, num_matricule : l'identifiant du medecin qui a effectué la consultation, num_ss : l'identifiant du patient, date_cons : la date de la consultation, heure_cons : l'heure de la consultation, tarif : le tarif de la consultation qui varie selon la spécialité, diagnostique : la conclusion du medecin après la consultation sur l'état de santé du patient, annulee : cet attribut prend la valeur TRUE quand la consultation est annulée.

Relation MedNonDispo : contient les périodes pendant lesquelles un medecin ne donne pas de consultation (congé,mission,...)

AVIS IMPORTANT AUX ETUDIANTS

1. Chacune des feuilles de votre copie doit comporter une étiquette code à barres placée à l'endroit indiqué «coller ici votre code à barres».
2. Une copie d'examen comporte une seule «feuille principale» et des «feuilles suites». Sur chacune de vos feuilles, le code à barres est obligatoire.
3. Cette feuille d'examen est strictement personnelle. Elle ne doit comporter aucun signe distinctif. Elle doit être écrite en noir et/ou bleu.
4. Le non respect de l'une de ces recommandations peut faire attribuer la note ZERO à l'épreuve.

NOTE

Coller ici votre
code à barre

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20

00	25	50	75

Examen Session Principale
Matière : PROGRAMMATION JAVA

Enseignant : S. Nouira/I. Ben Slimen Date : 10/ 01/2025

Filière / Classe : 2^{ème} ING INFO Durée : 1h30

Nbr. De pages : 8 Documents : non aut.

A part les 8 pages à rendre vous trouvez une annexe

Problème

L'objectif de ce problème est de réaliser en Java une partie d'un système informatique pour une chaîne africaine de supermarchés (Tunisia/Senegal/South Africa). Nous nous concentrerons sur deux types différents de produits : des livres et des paniers de fruits secs, ainsi que sur la partie du système informatique qui calcule le prix des produits après taxes pour tous les items vendus par cette chaîne, dans chacun des pays.

- 1) Créer une interface fonctionnelle appelée « Service » qui possède une méthode getPrice, retournant une valeur de type double qui représente le prix d'un service avant taxes.

Noms, prénoms et signature
de l'enseignant correcteur

Page 1 | 8



- 2) Compléter cette classe d'exception appelée « MarginNumberException » dans l'objectif est de générer un objet exception si un nombre réel noté nbr fourni en paramètre est hors d'un intervalle [Min, Max].

public class MarginNumberException { private double Min, Max, nbr ; // constructeur	void treat() {
/* méthode de traitement qui sert à affecter dans nbr la valeur Max (resp. Min) si nbr>Max (resp. si nbr<Min) et afficher un message sur la console informant de la manipulation*/ /*Exemple de message : The number inserted 5.6 is replaced by the minimum which is 30.0*/	

- 3) Un produit est défini par son prix de base obligatoirement entre 5.0 et 100.0, son pays de vente (énumération). Coder la classe Product et l'énumération adéquate. Il est à noter que cette classe doit disposer de la méthode getPrice, et qu'elle ne doit avoir que deux classes filles appelées « Book » et « NutsBasket ». N'oublier pas de coder les getters et les setters adéquats

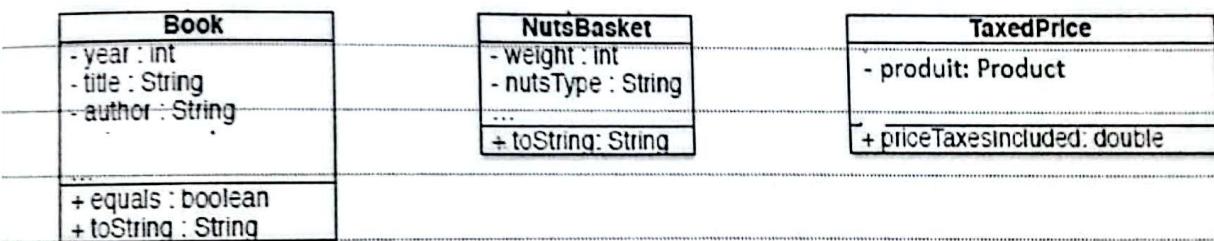


Fig1 : portion d'un digramme de classes (UML)

4) Un Livre est un produit caractérisé par son titre, son année d'édition (entier), son auteur.

On considère deux livres comme "égaux", s'ils ont le même titre et auteur, et ont été publiés la même année dans un même pays.

Créer la classe Book (voir figure 1) sachant qu'elle est héritable par n'importe quelle classe fille. Ne pas mettre les getters/setters des attributs (vous supposez déjà codés pour la suite)

Exemple d'affichage d'un objet de la classe Book :

"Java for dummies" by John Smith, sold for 85.0 plus taxes

5) NutsBasket (voir figure 1) possède 2 variables d'instance nutsType et le poids en grammes.

Dans le cas de la classe NutsBasket, le prix avant taxe est de 0.1 par gramme ajouté au prix de base. Créer la classe sachant qu'elle n'est pas héritable. Ne pas mettre les getters/setters des attributs (vous supposez déjà codés pour la suite).

Exemple d'affichage d'un objet de la classe NutsBasket:

basket of Pistachios costing 15.0 plus taxes

Annexe examen Java 2025

Interfaces fonctionnelles	Quelques méthodes statiques de la classe Collections Collection <T> sort(Collection <T> ,Comparator<T>)
Comparator <T> : int compare(T , T) Consumer <T> : void accept(T) Supplier <T> : T get() Predicate <T> : boolean test(T) Function <T1, T2> : T2 apply(T1) BiFunction <T1,T2,T3> : T3 apply(T1,T2) UnaryOperator <T> : T apply(T) BinaryOperator <T> : T apply(T,T) EventHandler<Event> handle(Event) ChangeListner<T> changed(ObservableValue<T> , T oldValue, T newValue)	Quelques méthodes de Stream <T> Stream <T> parallel() Stream <T> filter(Predicate<T>) Stream <T1> map(Function<T,T1>) forEach (Consumer<? Super T>) double collect(Collectors.summingDouble(référence au méthode))
Une propriété implémentant observable possède la méthode : addListener(ChangeListner<T>)	Exemples de propriétés : Slider : ValueProperty, ProgressIndicator : ProgressProperty, ToggleGroup : selectedToggleProperty
Quelques méthodes de List <T>	classe Thread implements Runnable add(int, T) set(int, T) ou set(T) T get(int) List < T > subList(int, int) int size()
Quelques méthodes de Collection<T>	Button setOnAction(EventHandler<ActionEvent>()); addEventHandler(ActionEvent.Action, (EventHandler<ActionEvent>()));
méthodes et membres statiques numériques	TextField + Label ToggleButton RadioButton getSelectedToggle()
int Integer.parseInt(String) double Double.parseDouble(String) String Integer.toString(int) String Double.toString(double)	setText(String) String getText()

Examen

Matière : Web Avancé

Enseignant(es)	: MK.JENDOUBI,C.ABIDI	Date	: 13/01/2025
Filière / Classe	: 2 INFO	Durée	: 1h30
Barème indicatif	: / 20	Documents	: aut. / non aut.
Nbre. de pages	: 4	Calculatrice	: aut. / non aut.

Exercice 1 (5 Points)

- 1) Dessiner le schéma de l'Architecture Web
- 2) Dessiner le schéma des interactions entre le fichier TypeScript et le fichier HTML

Exercice 2 (5 Points)

- 1) C'est quoi le modèle SPA?
- 2) Qu'est-ce qu'un "two-way data binding" en Angular ? Donne un exemple d'utilisation.
- 3) Quelles sont les caractéristiques du fichier TypeScript ?
- 4) Quelle est la différence entre "let" et "var" ?
- 5) Quelle est la différence entre ngOnInit() et le constructeur d'un composant Angular ?

Exercice 3 QCM (Une seule réponse est correcte) (5 Points)

- 1) Qu'est-ce qu'un "npm" ?
 - A) Network Performance Monitor
 - B) Network Package Manager
 - C) Node Package Manager
 - D) Node Performance Monitor

2) Qu'est-ce qu'un "module" Angular ?

- A) Un type de composant
- B) Un ensemble de services
- C) Une collection de composants, directives, pipes et services organisés
- D) Un fichier de configuration pour les routes

3) Quelle directive Angular permet de créer une boucle dans le template ?

- A) ngModel
- B) ngFor
- C) ngIf
- D) ngSwitch

4) Quel est le rôle de @Output() dans un composant Angular ?

- A) Permet de recevoir des événements depuis un composant enfant
- B) Permet de gérer les événements DOM
- C) Permet d'émettre des événements d'un composant enfant vers un composant parent
- D) Permet de lier des données entre les composants

5) "templateUrl" permet de :

- A) spécifier les feuilles de styles CSS associées à ce composant
- B) d'associer un fichier externe HTML contenant la structure de la vue du composant
- C) définir à l'intérieur du décorateur le code HTML représentant la vue du composant

D) définir à l'intérieur du décorateur le code CSS contenant la structure de la vue du composant

Exercice 4 (5 Points)
Compléter le code suivant :

app.component.ts :

```
import { Component } from '@angular/core';

interface Task {
  name: string;
  completed: boolean;
}

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  1) // Compléter ici pour initialiser la variable newTask
  tasks: Task[] = [
    { name: 'Faire les courses', completed: false },
    { name: 'Nettoyer la maison', completed: false },
    { name: 'Répondre aux emails', completed: false }
  ];

  addTask() {
    if (this.newTask) {
      this.tasks.push({ name: this.newTask, completed: false });
    2) // Compléter ici pour vider la variable newTask après l'ajout
    }
  }

  markAsCompleted(task: Task) {
    3) // Compléter ici pour marquer une tâche comme terminée
  }
}
```

app.component.html:

```
<div>
  <label for="newTask">Nouvelle tâche :</label>
  <input id="newTask" 4)="newTask" placeholder="Nom de la tâche"> // Compléter ici pour lier la variable newTask au champ de saisie dans le template.
  <button (click)="addTask()">Ajouter</button>
</div>

<ul>
  <li 5)>// Compléter ici pour afficher la liste des tâches dans un élément <ul>.
    <span [ngClass]="{{ 'completed': task.completed }}>{{ task.name }}</span>
    <button (click)="markAsCompleted(task)" [disabled]="task.completed">Terminer</button>
  </li>
</ul>
```

Bonne chance!

Examen

Matière : Web Avancé

Enseignant(es)	: MK.JENDOUBI,C.ABIDI	Date	: 13/01/2025
Filière / Classe	: 2 INFO	Durée	: 1h30
Barème indicatif	: / 20	Documents	: aut. / non aut.
Nbre. de pages	: 4	Calculatrice	: -aut. / non aut.

Exercice 1 (5 Points)

- 1) Dessiner le schéma de l'Architecture Web
- 2) Dessiner le schéma des interactions entre le fichier TypeScript et le fichier HTML

Exercice 2 (5 Points)

- 1) C'est quoi le modèle SPA?
- 2) Qu'est-ce qu'un "two-way data binding" en Angular ? Donne un exemple d'utilisation.
- 3) Quelles sont les caractéristiques du fichier TypeScript ?
- 4) Quelle est la différence entre "let" et "var" ?
- 5) Quelle est la différence entre ngOnInit() et le constructeur d'un composant Angular ?

Exercice 3 QCM (Une seule réponse est correcte) (5 Points)

- 1) Qu'est-ce qu'un "npm" ?

- A) Network Performance Monitor
- B) Network Package Manager
- C) Node Package Manager
- D) Node Performance Monitor

2) Qu'est-ce qu'un "module" Angular ?

- A) Un type de composant
- B) Un ensemble de services
- C) Une collection de composants, directives, pipes et services organisés
- D) Un fichier de configuration pour les routes

3) Quelle directive Angular permet de créer une boucle dans le template ?

- A) ngModel
- B) ngFor
- C) ngIf
- D) ngSwitch

4) Quel est le rôle de @Output() dans un composant Angular ?

- A) Permet de recevoir des événements depuis un composant enfant
- B) Permet de gérer les événements DOM
- C) Permet d'émettre des événements d'un composant enfant vers un composant parent
- D) Permet de lier des données entre les composants

5) "templateUrl" permet de :

- A) spécifier les feuilles de styles CSS associées à ce composant
- B) d'associer un fichier externe HTML contenant la structure de la vue du composant
- C) définir à l'intérieur du décorateur le code HTML représentant la vue du composant

D) définir à l'intérieur du décorateur le code CSS contenant la structure de la vue du composant

Exercice 4 (5 Points)

Compléter le code suivant :

```
app.component.ts :  
  
import { Component } from '@angular/core';  
  
interface Task {  
  name: string;  
  completed: boolean;  
}  
  
@Component({  
  selector: 'app-root',  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})  
export class AppComponent {  
  1) // Compléter ici pour initialiser la variable newTask  
  tasks: Task[] = [  
    { name: 'Faire les courses', completed: false },  
    { name: 'Nettoyer la maison', completed: false },  
    { name: 'Répondre aux emails', completed: false }  
  ];  
  
  addTask() {  
    if (this.newTask) {  
      this.tasks.push({ name: this.newTask, completed: false });  
    2) // Compléter ici pour vider la variable newTask après l'ajout  
    }  
  }  
  
  markAsCompleted(task: Task) {  
    3) // Compléter ici pour marquer une tâche comme terminée  
  }  
}
```

app.component.html:

```
<div>
  <label for="newTask">Nouvelle tâche :</label>
  <input id="newTask" 4)="newTask" placeholder="Nom de la tâche"> // Compléter ici pour lier la variable newTask au champ de saisie dans le template.
  <button (click)="addTask()">Ajouter</button>
</div>

<ul>
  <li 5)>// Compléter ici pour afficher la liste des tâches dans un élément <ul>.
    <span [ngClass]="{{ 'completed': task.completed }}>{{ task.name }}</span>
    <button (click)="markAsCompleted(task)" [disabled]="task.completed">Terminer</button>
  </li>
</ul>
```

Bonne chance!

Recherche Opérationnelle et Optimisation

Enseignant(s)	: S. Ben Messaoud	Date	: 13/01/2025
Filière / Classe	: 2ème ING Info	Durée	: 1h30
Barème indicatif	: 4 / 5 / 11	Documents	: aut. / non aut.
Nbre. de pages	: 02	Calculatrice	: aut. / non aut.

NB : Utiliser les notations du cours et reporter sur votre copie toutes les itérations exécutées.

Exercice 1 :

Résoudre par l'algorithme de simplex, le programme linéaire suivant :

$$\begin{aligned} \text{Min } Z = & -2x_1 - 6x_2 \\ \left\{ \begin{array}{l} 4x_1 - 2x_2 \leq 20 \\ -2x_1 + 4x_2 \leq 16 \\ 2x_1 + x_2 \leq 14 \\ x_1, x_2 \geq 0 \end{array} \right. \end{aligned}$$

Exercice 2:

Un fabricant d'appareils électroménagers vient de racheter l'usine de l'un de ses concurrents qu'il a rapidement reconvertie pour la production de sa marque. Les deux usines dont il dispose maintenant n'ont pas les mêmes caractéristiques. Le coût d'une heure de fonctionnement de l'usine A est de 350 D, et celui de l'usine B est de 300 D. Ce fabricant doit satisfaire exactement une commande de 1000 cuisinières et de 1200 lave-linge. Par contre, il veut produire au moins 500 réfrigérateurs. Le tableau ci-dessous indique les temps requis (en minutes) pour produire chaque appareil dans chacune des deux usines.

	Cuisinière	Réfrigérateur	Lave-linge
Usine A	120 minutes	240 minutes	180 minutes
Usine B	180 minutes	180 minutes	120 minutes

Le nombre d'heures disponibles est de 5400 heures dans l'usine A et de 4800 heures dans l'usine B pour la période de production considérée. Le fabricant désire savoir combien d'appareils de chaque type doit-il fabriquer dans chaque usine s'il veut minimiser les coûts de production.

Formuler un programme linéaire correspondant à ce problème.

Exercice 3:

L'exécution d'un projet exige la réalisation de 10 tâches A, B, C, D, E, F, G, H, I et J dont les durées (en jours) et les contraintes de dépendance sont données par le tableau suivant :

Tâche	Durée (jours)	Antériorités
A	5	-
B	4	-
C	7	A
D	7	A, B
E	10	C, D
F	6	B, D
G	3	F
H	5	F
I	5	E, G
J	2	G, H

1. Donner le graphe MPM de ce projet.
2. Déterminer le calendrier d'exécution des tâches (dates de début au plus tôt et au plus tard).
3. En déduire la durée minimale du projet et le(s) chemin(s) critique(s).
4. Indiquer la marge libre de chaque tâche non critique.
5. On supposera que la durée d'exécution de la tâche J est retardée de deux jours, donner dans ce cas les conséquences sur le déroulement de ce projet du point de vue durée minimale et chemin(s) critique(s).
6. Donner le graphe PERT correspondant à ce projet.

Année universitaire 2024-2025
EXAMEN SESSION PRINCIPALE
Matière : Comptabilité d'Entreprise

Enseignant(es) : Mme Turki
Filière / Classe : 2^{ème} Info

Date : 06/01/2025
Durée : 1h30
Documents : aut. / non aut.
Calculatrice : aut. / nonaut.

Nbre. de pages : 1

Les éléments du bilan de l'entreprise « CABLETUNISIE » de fabrication de câbles électriques au 31/12/N se présentent comme suit :

élément	Montant	élément	Montant
Capital social	200 000	Banque	10 300
Etat,TVA déductible	43 000	Etat,TVA collectée	57 000
Constructions	19 500	Fournisseurs d'Exploitation	20 340
Créditeurs Divers	14 560	Equipement de Bureau	40 550
Clients Avance et Acompte	11 950	Stock de Marchandises	22 600

L'entreprise est assujettie à la TVA au taux de 20%

La méthode de l'inventaire permanent est utilisée dans cet exercice

Durant le mois de janvier N+1 elle a effectué les opérations suivantes :

04/01 : « CABLETUNISIE » commande suivant bon n°2378 des marchandises d'une valeur brute de 90 000 D, elle envoie le chèque N°56 pour 15 000 D au fournisseur « ABC ». La marchandise et la facture N°AST3311 correspondantes à la commande N°2378 parviennent 7jours après .Cette facture comporte deux rabais successifs de 3% et de 1% ainsi qu'une ristourne de 4%, une remise de 1% On note par ailleurs que tout règlement sous quinzaine bénéficie d'une escompte de règlement de 3%.Le fournisseur « ABC » a effectué le transport en espèces par un transporteur et l'a facturé à « CABLETUNISIE » pour 500 DHT (TVA 10%)

22/01 : « CABLETUNISIE » vend à la société « MT CABLE » une marchandise d'une valeur de 200 000DHT remise 1% Escompte 3% rabais 4%

27/01 : « CABLETUNISIE » vend de la marchandise à son client « DESERTA » d'une valeur brute de 60 000D ,une remise de 7 000D, un escompte de 5 000D, TVA 20%, Facture N°AZ547
« DESERTA » a avancé le 19/01 la somme de 10 000 D à « CABLETUNISIE » chèque N°123

TRAVAIL A FAIRE : 1-Etablir le bilan de clôture de l'entreprise « CABLETUNISIE » au 31/12/N

2-Etablir la facture N°AST3311 du 04/01/N+1

3-Enregistrer au journal cette facture :

a-Chez l'acheteur

b-Chez le vendeur

4-Enregistrer l'opération du 22/01/N+1

a-chez l'acheteur

b-chez le vendeur

5-Etablir la facture N°AZ547

a-Enregistrer chez l'acheteur et chez le vendeur

b- En supposant que le cout de transport de ces marchandises est de 700DHT(TVA 10%) procéder aux écritures nécessaires chez l'acheteur et chez le vendeur dans chacun des cas suivants :

- a- Transport effectué par les propres moyens du vendeur et facturé au client
- b- Transport effectué par le client et non facturé par le fournisseur
- c- Transport effectué par un transporteur externe et facturé au vendeur qui a donné des espèces. Ce dernier l'a fait supporter au client
- d- Transport effectué par les propres moyens du vendeur et n'a pas été facturé au client
- e- Transport a été effectué par un transporteur externe et facturé au client

6-Etablir le 31/01/N+1 le solde du compte Stock de marchandises de la société « CABLETUNISE » en enregistrant tous les mouvements du mois de janvier N+1 de stock de marchandises dans le compte en T

AVIS IMPORTANT AUX ETUDIANTS

- Chacune des feuilles de votre copie doit comporter une étiquette code à barres placée à l'endroit indiqué « coller ici votre code à barres ».
- Une copie d'examen comporte une seule « feuille principale » et des « feuilles suites ». Sur chacune de vos feuilles, le code à barres est obligatoire.
- Cette feuille d'examen est strictement personnelle. Elle ne doit comporter aucun signe distinctif. Elle doit être écrite en noir et/ou bleu.
- Le non respect de l'une de ces recommandations peut faire attribuer la note ZERO à l'épreuve.

Coller ici votre
code à barre

NOTE

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20

00	25	50	75

Examen de la Session Principale
Matière : Systèmes d'Exploitation

Enseignantes : K. Bedoui & R. Tlili **Date** : 06/01/2022

Filière/Classe : Ingénieur Informatique/2eme **Durée** : 1h30

Documents : non autorisés **Calculatrice non programmable** : autorisée

Exercice 1 : Question du cours **4 points** **15 min**

1. Cocher les bonnes cases :

Solution	Caractéristiques	Matérielle	Logicielle à base de variable			Avec attente active	Avec attente passive
			Simple	Structure	Classe		
TSL							
Peterson							
Sémaphore							
Moniteur							



2. Cocher les propositions correctes :

La MMU est :

- un Module de gestion de l'exclusion MUuelle
- une structure de donnée employée pour la gestion de la mémoire vive
- un dispositif électronique utilisé pour la gestion de la RAM
- une unité de transformation des adresses des blocs du disque dur et de la RAM
- une unité de transformation des adresses (logiques/virtuelles)

Le NRU est :

- un SGF relatif à WindowsNT
- un dispositif électronique qui se base sur deux bits R et M
- un algorithme de remplacement de cadres
- le numéro majeur du disque dur
- un algorithme de placement de pages qui se base sur l'utilisation des deux bits R et M

Le TSL est :

- un algorithme d'ordonnancement des processus sous Linux (selon le Shortest Time)
- une instruction qui verrouille le bus d'adresse durant son exécution
- une instruction assembleur atomique
- une solution au problème de synchronisation avec attente passive
- une instruction qui met le flag à 1 et charge le contenu du flag dans le registre

3. Quelle est la différence entre une stratégie de placement et une stratégie de remplacement de page. Citer trois exemples de chaque stratégie.

Exercice 2 : Processus et RAM

10 points

30 min

Soit une mémoire paginée où chaque adresse virtuelle comporte un numéro de page sur 4 bits et un déplacement dans la page sur 12 bits. La mémoire centrale est de taille 12288 \varnothing sans compter la partie système.

1. Quelle est la taille d'une page en K \varnothing ? Justifier.

2. Quelle est la taille maximale d'un processus ? Justifier.

3. Quels est le nombre de cadres en mémoire centrale ? Justifier.

Soient trois processus A, B et C s'exécutant dans ce système selon les hypothèses suivantes :

- La mémoire centrale comporte des cadres initialement libres.
- L'utilisation de la technique de pagination.
- La pagination est à la demande (la page n'est chargée en RAM que si elle est demandée).
- Le remplacement de pages se base sur l'algorithme **clock**.
- Si un processus termine son exécution alors ses pages sont retirées du système.
- L'ordonnancement sur le processeur se fait selon la stratégie RR avec $Q=2$ UT (Unité de Temps).

Les détails des processus sont donnés dans le tableau suivant :

Processus	A	B	C
Date d'arrivée	0	1	3
Durée d'exécution	4	6	4
Taille réelle en KØ	15	31	24

4. Compléter le tableau suivant :

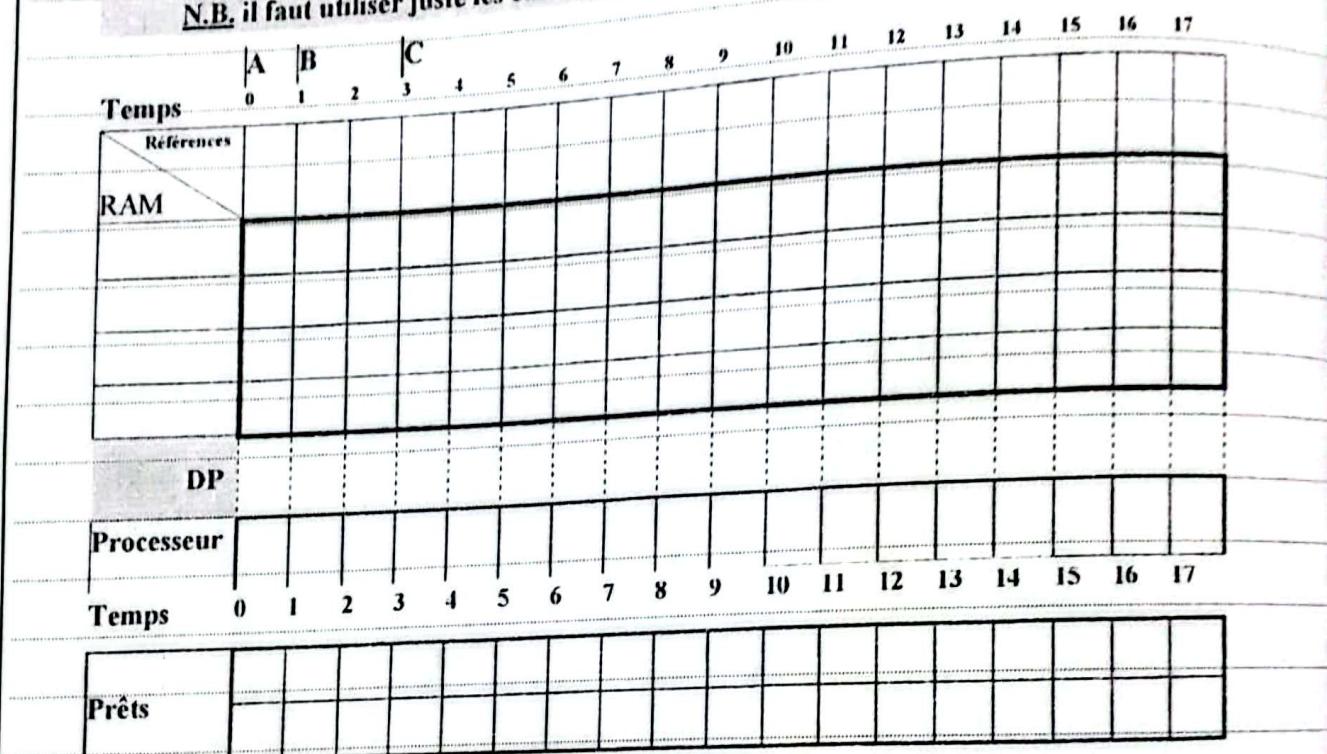
Processus	Nombre de pages	Taille physique	Fragmentation interne
A			
B			
C			

Nous supposons que chaque processus fait référence à sa première page ensuite à sa deuxième page et ce d'une manière répétitive si sa durée le permet encore. Sachant que chaque processus reste sur la première page 1 UT, alors que sur la deuxième page il reste 2 UT.

5. Déterminer la liste de références pour chaque processus et indiquer le type d'accès.

Processus	Liste de références aux pages
A	
B	
C	

6. Dresser l'évolution de l'état de la mémoire centrale, des processus et du processeur :
- N.B. il faut utiliser juste les cadres nécessaires déjà calculés dans la question 3.



7. Déterminer le nombre de défauts de pages :

8. Compléter le tableau suivant :

Temps de Réponse Moyen (TRM)	Temps d'Attente Moyen (TAM)	Nombre de commutations de contexte



20 min

4 points

Exercice 3 : Synchronisation

Pour garantir son bon déroulement, un club de natation applique le règlement suivant :

- A l'entrée, chaque nageur doit acquérir un panier s'il y en a.
- Ensuite, il cherche une cabine pour porter son maillot de bain et mettre ses vêtements dans le panier.
- Si le nageur termine de se changer, alors il libère la cabine mais garde le panier avec lui.
- Lorsqu'il finit de nager, il doit retrouver une cabine pour se rhabiller.
- Enfin, lorsqu'il termine de se rhabiller, il libère la cabine et ensuite le panier.

Le club met à la disposition de ses nageurs N paniers et M cabines.
Chaque nageur effectue, donc, les opérations principales suivantes :

```
Processus Nageur
{
    <se_changer>
    <nager>
    <se_rhabiller>
}
```

1. Déterminer les ressources partagées entre les nageurs ?

2. Modifier le code suivant afin de synchroniser les nageurs et initialiser les sémaphores.

```
semaphore sp, sc ;
sem_init(sp,.....) ;           // initialiser ce sémaphore
sem_init(sc,.....) ;           // initialiser ce sémaphore
```

```
Processus Nageur
{
    <se_changer>

    <nager>

    <se_rhabiller>

}
```

3. Préciser l'utilité de chaque sémaphore :

- `sp` permet de

- `sc` permet de

4. En déduire le nombre maximal de nageurs que peut contenir la piscine à un instant donné.

Exercice 4 : Synchronisation pure

2 points

10 min

Voici le graphe de précédence de 4 processus P1, P2, P3 et P4 :

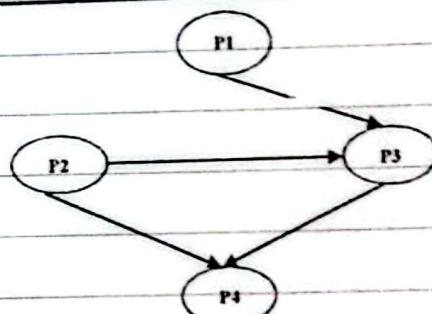
Proposer une solution, à l'aide de sémaphores, pour synchroniser ces processus tels présentés dans le graphe.

Chaque procédure `Ai()` représente l'action du processus `Pi`.

1. Compléter la déclaration

`semaphore
sem_init`

2. Compléter le code de chaque processus :



<code>P1()</code>	<code>P2()</code>	<code>P3()</code>	<code>P4()</code>
{	{	{	{
<code>A1();</code>	<code>A1();</code>	<code>A1();</code>	<code>A1();</code>
}	}	}	}

AVIS IMPORTANT AUX ETUDIANTS

- Chacune des feuilles de votre copie doit comporter une étiquette code à barres placée à l'endroit indiqué «coller ici votre code à barres».
- Une copie d'examen comporte une seule «feuille principale» et des «feuilles suites». Sur chacune de vos feuilles, le code à barres est obligatoire.
- Cette feuille d'examen est strictement personnelle. Elle ne doit comporter aucun signe distinctif. Elle doit être écrite en noir et/ou bleu.
- Le non respect de l'une de ces recommandations peut faire attribuer la note ZERO à l'épreuve.

NOTE

Coller ici votre code à barre

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20

00	25	50	75

Examen Session Principale

Matière : Technical English

Enseignant(es)	: Mme Slimane	Date	: 06 / 01 /2025
Filière / Classe	: 2 ^{ème} Génie Informatique	Durée	: 1h30
Barème indicatif	: 7/13	Document	: aut. / non aut.
Nbre. de pages	: 3	Calculatrice	: aut. / non aut.

Reading comprehension questions (7 pts)

1. What is the primary purpose of the text? (1)

- a. To describe the history and evolution of cloud computing.
- b. To explain the technical architecture of cloud computing.
- c. To criticize the challenges of cloud computing.
- d. To compare cloud computing with traditional computing.

2) According to the text, what prediction did John McCarthy make in the 1960s? (0.5)

3/ What does the term "scalability" mean in the context of cloud computing? (0.5)

Noms, prénoms et signature
de l'enseignant correcteur



4) How does the "pay-as-you-go" pricing model differ from traditional IT infrastructure investments?(1 pt)

.....
5) Why might hybrid cloud models become more popular in the future?(1pt)

6) What role might artificial intelligence play in the future of cloud computing... (2)

Language (13pts)

1) How can you define these abbreviations: (2)

AWS

CU

-G1-

CD

2) Complete these phrasal verbs with a preposition (1.5)

1) We can tap this pain point to help us market the product.

2) Thank you for pointing this issue to me . I was not aware of it ..

3) I took his duties as the technical lead of that project .

4) He camewith this idea all by himself.

5) Hover your camera the timetable

6) This app was written in **C++**.

The text

Cloud computing has revolutionized the way individuals and businesses store, process, and access data. At its core, cloud computing refers to the delivery of computing services—such as servers, storage, databases, networking, software, and analytics—over the internet, commonly known as "the cloud."

The concept of cloud computing dates back to the 1960s when computer scientist John McCarthy predicted that computing would one day become a utility, much like electricity or water. However, it was not until the early 2000s that cloud computing began to take shape in its modern form. Amazon played a pivotal role in this evolution with the launch of Amazon Web Services (AWS) in 2006, offering on-demand storage and computing power.

One of the most significant advantages of cloud computing is scalability. Users can easily scale their resources up or down based on demand, ensuring cost-efficiency. Additionally, the cloud enables global accessibility, allowing teams from different parts of the world to collaborate seamlessly. Another critical feature is the "pay-as-you-go" pricing model, where users only pay for the resources they consume.

Cloud services are typically divided into three categories: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

- **IaaS** provides virtualized computing resources over the internet. Examples include Amazon EC2 and Google Compute Engine.
- **PaaS** offers a platform for developers to build, test, and deploy applications. Examples include Microsoft Azure and Heroku.
- **SaaS** delivers software applications over the internet, such as Google Workspace and Dropbox.

Despite its numerous benefits, cloud computing is not without challenges. Data security and privacy remain major concerns, as sensitive information is stored offsite and potentially vulnerable to cyberattacks. Additionally, the dependence on a stable internet connection can be a limitation in areas with poor connectivity.

Looking ahead, the future of cloud computing appears promising. Emerging trends include edge computing, which processes data closer to its source to reduce latency, and the growing integration of artificial intelligence (AI) to optimize cloud services. Moreover, hybrid cloud models that combine private and public cloud resources are expected to gain popularity, offering greater flexibility and control.

3) Put the words between brackets in the right form (4.5pts)

Programming languages (1) (evolve) over time to meet the needs of developers and the industries they serve. Early languages, such as Assembly, (2) (require) programmers to write instructions directly for the hardware, which (3) (be) both time-consuming and error-prone. Modern languages like Python and JavaScript (4) (design) to simplify coding by providing more abstract and user-friendly syntax. While Python (5) (popularize) for its versatility and readability, JavaScript (6) (dominate) web development due to its ability to (7) (create) dynamic and interactive interfaces. Understanding how (8) (choose) the right language for a project is a critical skill that programmers (9) (develop) through experience and practice.

Fill in the blanks with the correct words from the box below. Each word can be used only once. (5pts)

Words: algorithm, neural networks, automation, machine learning, data, robotics, intelligence, ethical, innovation, prediction

1. The process of training a system to improve its performance using past experiences is called
2. In AI, are computational models inspired by the human brain.
3. Companies use AI to analyze large amounts of and extract valuable insights.
4. The development of has significantly transformed industries by reducing the need for manual labor.
5. Writing an efficient is essential for solving complex problems in AI systems.
6. One of the main debates in AI focuses on issues, such as privacy and bias.
7. The use of AI in weather forecasting has improved the accuracy of models.
8. AI and have led to significant advancements in the field of healthcare.
9. refers to the ability of machines to perform tasks that typically require human cognition.
10. is a branch of engineering focused on designing and building intelligent machines.