REPUBLIQUE DU CAMEROUN Paix – Travail – Patrie		ANNEE SCOLAIRE : 2025-2026
MINISTERE DES ENSEIGNEMENTS SECONDAIRES		CLASSE : Tle C&D
DELEGATION REGIONALE DU LITTORAL		Durée : 2h
DELEGATION DEPARTEMENTALE DU WOURI	Site web : www.groupesimo.org	Coef : 2
IMMATRICULATION CCG : 7IJ2GWFD3I1I1402I02	**EVALUATION N°3**	

Examinateur : Soreil TCHEPDA

EPREUVE D'INFORMATIQUE

Partie A : SYSTEMES INFORMATIQUES 5PTS

Un centre de santé intégré (CSI) vient d'installer un système informatique pour :

- L'enregistrement des patients,
- La gestion des consultations,
- L'édition des factures médicales.

Le matériel et les logiciels disponibles sont :

- Processeur : AMD Ryzen 5 – 3.6 GHz
- Disque dur : 256 Go SSD
- Mémoire RAM : 8 Go
- Système d'exploitation : Ubuntu 22.04 LTS
- Logiciel installes : Antivirus, LibreOffice
- Carte réseaux : Ethernet 10/100/1000 Mbps
- Ecran : 1600 x 900 pixels ; 45 cm x 28 cm

Après une panne électrique, plusieurs dossiers ont été perdus.

Travail demandé :

1) Définir un **système informatique**. **0,5pt**
2) Donner le **type de système informatique** utilisé au CSI. **0,5pt**
3) Citer le **système d'exploitation** installé et préciser son **type**. **0,5pt**
4) Donner **deux avantages** de l'utilisation d'un disque SSD. **1pt**
5) Donner la **résolution** et la **taille en pouce** de l'écran. **1pt**
6) Citer un **équipement matériel** permettant d'éviter la perte de données en cas de coupure électrique. **0,5pt**
7) Proposer **une mesure organisationnelle** et **une mesure technique** pour sécuriser les données du centre. **1pt**

Partie B : SYSTEME D'INFORMATION ET BASE DE DONNÉES 8PTS

Le CSI souhaite informatiser la gestion :

- Des patients,
- Des consultations médicales,
- Des médecins.

CBCG Terminale C&D

Règles de gestion :

- **RG1** : Chaque patient est identifié par un numéro unique.
- **RG2** : un patient peut effectuer plusieurs consultations.
- **RG3** : une consultation concerne un seul patient et un seul médecin.
- **RG4** : un médecin peut assurer plusieurs consultations.
- **RG5** : Chaque consultation est enregistrée par un agent administratif.

Exercice 1 : **(4,25 points)**

1. Définir les notions suivantes : **entité**, **association**. **1pt**
2. À partir des règles de gestions :
   a) Identifier les entités du système. **1pt**
   b) Identifier les associations entre ces entités. **1pt**
3. Construire le **MCD** en faisant apparaître les **entités**, les **associations** et les **cardinalités**. **1pt**
4. Déduire le MLD correspondant. **1pt**

Exercice 2 : **(3,75 points)**

On vous donne le schéma relationnel suivant :

- PATIENT(NumPatient, Nom, Age, Contact)
- MEDECIN(MatriculeMed, NomMed, Specialite)
- AGENT(MatriculeAgent, NomAgent)
- CONSULTATION(NumConsult, DateConsult, #NumPatient, #MatriculeMed, #MatriculeAgent)

1. Identifier la clé primaire de chaque table. **1pt**
2. Indiquer les clés étrangères. **0,75pt**
3. Ecrire la requête SQL permettant d'enregistrer la consultation suivante : **1pt**

CONS2026-008	2026-01-15	PAT032	MED004	AGT010

4. Expliquer le rôle de la requête suivante : **1pt**

```
SELECT NomMed, Specialite
FROM MEDECIN
WHERE Specialite = "Pédiatrie";
```

Partie B : ALGORITHME ET PROGRAMMATION **7PTS**

Un centre de formation souhaite analyser les **notes finales** de ses apprenants. Chaque apprenant est caractérisé par :

- Son **nom**,
- Sa **note finale** sur 100.

L'objectif est de concevoir un algorithme permettant de :

- Déterminer la **meilleure note**,
- Afficher les **noms des apprenants recalés** (note < 50).

Travail demandé :

CBCG Terminale C&D

1. Définir un **algorithme**. **0,5pt**
2. Déclarer deux tableaux pour stocker les **noms** et les **notes** de **25 apprenants** du centre. **1pt**
3. Préciser le type de chaque donnée utilisée. **0,5pt**
4. Compléter (sur la feuille de composition) l'algorithme incomplet ci-après, qui permet de saisir le nom et la note de chacun des 25 apprenants. **1pt**

```
Début
        Pour i allant de …à … Faire
                ……
                ……
                ……
                ……
        FinPour
Fin
```

5. On considère l'algorithme suivant. Les tableaux Nom[1..25] et Note[1..25] contiennent respectivement les noms et les notes de 25 apprenants.

```
1   Debut
2       Max ← Note[1]
3       Pour i ← 2 À 25 Faire
4           Si (Note[i] > Max) Alors
5             Max ← Note[i]
6           FINSI
7       FinPOUR
8       Ecrire("La meilleure note est : ", Max)
9       Ecrire("Liste des apprenants recalés :")
10      Pour i ← 1 À 25 Faire
11          Si (Note[i] < 50) Alors
12             Ecrire Nom[i]
13          FinSi
14      FinPour
15  Fin
```

   a) Expliquer le rôle de l'instruction : Max ← Note[1]. **0,5pt**
   b) Relever dans cet algorithme : **1pt**
      - ❖ Le bloc d'instructions permettant de rechercher la meilleure note.
      - ❖ Le bloc d'instructions permettant d'afficher les apprenants recalés.
6. Citer un **IDE** adapté à la programmation en **C**. **0,5pt**
7. Indiquer la **bibliothèque C** nécessaire pour **exécuter** cet algorithme. **0,5pt**
8. Traduire les trois lignes suivantes en langage.
   - ❖ Pour i allant de 2 à 25 Faire **0,5pt**
   - ❖ Si (Note[i] > Max) Alors **0,5pt**
   - ❖ Max ← Note[i] **0,5pt**

CBCG Terminale C&D

CORRECTION DE L'ÉVALUATION N°3 - INFORMATIQUE

Classe : Terminale C&D | Année Scolaire : 2025-2026

PARTIE A : SYSTÈMES INFORMATIQUES (5 PTS)

1. **Définition d'un système informatique :**
   Un système informatique est un ensemble intégré composé de matériel (hardware), de logiciels (software), de données et de procédures, conçu pour collecter, traiter, stocker et diffuser l'information afin d'aider à la prise de décision ou au contrôle dans une organisation. (0,5 pt)
2. **Type de système informatique au CSI :**
   Il s'agit d'un **système informatique centralisé** (ou monoposte/autonome), car les ressources sont dédiées à une seule machine pour les tâches locales décrites. (0,5 pt)
3. **Système d'exploitation :**
   - Nom : **Ubuntu 22.04 LTS**.
   - Type : Système d'exploitation **Libre** (Open Source) basé sur Linux (GNU/Linux). (0,5 pt)
4. **Deux avantages du disque SSD :** (1 pt)
   - Vitesse de lecture/écriture très élevée (démarrage rapide).
   - Résistance aux chocs (pas de pièces mécaniques mobiles).
   - Silence de fonctionnement et faible consommation énergétique.
5. **Résolution et taille de l'écran :** (1 pt)
   - **Résolution :** 1600×900 pixels.
   - **Taille en pouces :** Calcul de la diagonale en cm : $D_{cm} = \sqrt{45^2 + 28^2} = \sqrt{2025 + 784} = \sqrt{2809} = 53$ cm.
     Conversion en pouces (1 pouce $= 2,54$ cm) : $D_{pouces} = \frac{53}{2,54} \approx 20,86$.
     La taille est d'environ **21 pouces**.
6. **Équipement contre les coupures électriques :**
   Un **Onduleur** (UPS). (0,5 pt)
7. **Mesures de sécurité :** (1 pt)
   - **Mesure technique :** Mise en place de sauvegardes automatiques régulières (backup) ou installation d'un pare-feu.
   - **Mesure organisationnelle :** Formation du personnel aux bonnes pratiques (mots de passe complexes) ou rédaction d'une charte informatique.

PARTIE B : SYSTÈME D'INFORMATION ET BASE DE DONNÉES (8 PTS)

Exercice 1 : Modélisation (4,25 pts)

1. **Définitions :** (1 pt)
   - **Entité :** Objet réel ou abstrait possédant une existence propre et décrit par des attributs (ex : Patient, Médecin).
   - **Association :** Lien sémantique reliant deux ou plusieurs entités, décrivant une interaction entre elles (ex : Consulter).
2. **Analyse des règles de gestion :**

   [label=)]**Entités identifiées :** (1 pt)

   (a)
   - PATIENT
   - MEDECIN
   - AGENT
   - CONSULTATION (Considérée ici comme une entité associative ou événementielle car elle possède ses propres données et liens multiples).

   (b) **Associations identifiées :** (1 pt)
   - Entre PATIENT et CONSULTATION : *Effectuer*
   - Entre MEDECIN et CONSULTATION : *Assurer*
   - Entre AGENT et CONSULTATION : *Enregistrer*
3. **Modèle Conceptuel de Données (MCD) :** (1 pt)

   Représentation textuelle des cardinalités :
   - Un PATIENT effectue **(1,n)** CONSULTATIONS. Une CONSULTATION concerne **(1,1)** PATIENT.
   - Un MEDECIN assure **(1,n)** CONSULTATIONS. Une CONSULTATION est assurée par **(1,1)** MEDECIN.
   - Un AGENT enregistre **(1,n)** CONSULTATIONS. Une CONSULTATION est enregistrée par **(1,1)** AGENT.
4. **Modèle Logique de Données (MLD) :** (1 pt)

   Les clés primaires sont soulignées, les clés étrangères sont précédées de #.
   - PATIENT (NumPatient, Nom, Age, Contact)
   - MEDECIN (MatriculeMed, NomMed, Specialite)
   - AGENT (MatriculeAgent, NomAgent)
   - CONSULTATION (NumConsult, DateConsult, #NumPatient, #MatriculeMed, #MatriculeAgent)

Exercice 2 : Exploitation du schéma relationnel (3,75 pts)

Schéma :
- PATIENT(NumPatient, Nom, Age, Contact)
- MEDECIN(MatriculeMed, NomMed, Specialite)
- AGENT(MatriculeAgent, NomAgent)
- CONSULTATION(NumConsult, DateConsult, #NumPatient, #MatriculeMed, #MatriculeAgent)

1. **Clés primaires :** (1 pt)
   - PATIENT : `NumPatient`
   - MEDECIN : `MatriculeMed`

— AGENT : `MatriculeAgent`
— CONSULTATION : `NumConsult`

2. **Clés étrangères :** (0,75 pt)
Dans la table **CONSULTATION** :
— `NumPatient` (référence PATIENT)
— `MatriculeMed` (référence MEDECIN)
— `MatriculeAgent` (référence AGENT)

3. **Requête SQL d'insertion :** (1 pt)

```
INSERT INTO CONSULTATION (NumConsult, DateConsult, NumPatient, MatriculeM
VALUES ('CONS2026-008', '2026-01-15', 'PAT032', 'MED004', 'AGT010');
```

4. **Rôle de la requête SELECT :** (1 pt)
`SELECT NomMed, Specialite FROM MEDECIN WHERE Specialite="Pédiatrie";`
Cette requête permet d'**afficher le nom et la spécialité** de tous les médecins qui sont spécialistes en **Pédiatrie**.

PARTIE C : ALGORITHME ET PROGRAMMATION (7 PTS)

1. **Définition d'un algorithme :**
Suite finie et ordonnée d'instructions élémentaires permettant de résoudre un problème donné. (0,5 pt)

2. **Déclaration des tableaux :** (1 pt)

```
Tableau Nom[1..25] : Cha ne
Tableau Note[1..25] : R el
```

3. **Types des données :** (0,5 pt)
— `Nom` : Chaîne de caractères
— `Note` : Réel (ou Entier)
— `i` : Entier (compteur)
— `Max` : Réel

4. **Algorithme de saisie complété :** (1 pt)

```
D but
    Pour i allant de 1    25 Faire
        Ecrire("Entrez le nom de l'apprenant␣",␣i)
␣␣␣␣␣␣␣␣Lire(Nom[i])
␣␣␣␣␣␣␣␣Ecrire("Entrez␣la␣note␣de␣l'apprenant ", i)
        Lire(Note[i])
    FinPour
Fin
```

5. **Analyse de l'algorithme donné :**
[label=)]**Rôle de** `Max ← Note[1]` **:** (0,5 pt)
Initialiser la variable `Max` avec la première note du tableau pour servir de référence lors des comparaisons suivantes. **Blocs d'instructions :** (1 pt)

(b) — *Recherche de la meilleure note :*

```
Max <- Note[1]
Pour i <- 2    25 Faire
    Si(Note[i] > Max) Alors
        Max <- Note[i]
    FINSI
FinPOUR
```

— *Affichage des recalés :*

```
Pour i <- 1    25 Faire
    Si(Note[i] < 50) Alors
        Ecrire Nom[i]
    FinSi
FinPour
```

6. **IDE pour le langage C :**
   Code : :Blocks, Dev-C++, Visual Studio Code, CLion. (0,5 pt)
7. **Bibliothèque C nécessaire :**
   `stdio.h` (Standard Input Output). (0,5 pt)
8. **Traduction en langage C :** (1,5 pt)

```c
// Pour i allant de 2    25 Faire
for (i = 2; i <= 25; i++) {

    // Si(Note[i] > Max) Alors
    if (Note[i] > Max) {

        // Max <- Note[i]
        Max = Note[i];
    }
}
```