



Maturité gymnasiale

Session 2025

EXAMEN DE L'OPTION COMPLÉMENTAIRE INFORMATIQUE

Informations et consignes :

- | | |
|--|---|
| x Temps à disposition : 3 heures. | x Vous devez écrire proprement <u>au stylo ou à l'encre</u> . |
| x Seul document autorisé : aide-mémoire Python 3 en annexe | La présentation est prise en compte pour la note. |
| x Calculatrice non programmable autorisée. | x Utilisez <u>une feuille par problème</u> . |
| x Le nombre de points est indiqué pour chacun des problèmes. Il y a 100 points au total. | x Écrivez votre <u>nom sur chaque feuille</u> . |
| | x Faites une <u>marge</u> de 2 cm à gauche. |
| | x Rendez <u>tous vos documents</u> , y compris la donnée. |
-

Problème 1 : machine de Turing (10 + 15 = 25 points)

Concevez une machine de Turing qui vérifie si une chaîne binaire donnée est un *palindrome* (une chaîne identique dans les deux sens de lecture, comme « 101 » ou « 0110 »).

- a) Expliquez (en pseudo-code) comment vous vérifieriez qu'une chaîne est un palindrome avec une machine de Turing.

Remarque : vous avez le droit de modifier la chaîne de caractères écrite sur le ruban pendant le processus de vérification.

- b) Écrivez la table d'actions.

Problème 2 : structure de données (15 + 5 + 5 = 25 points)

On utilise un tas où l'élément le plus grand est à la racine.

- a) Entrez les nombres suivants, dans cet ordre, dans un tas. Dessinez l'évolution du tas, sous forme d'un arbre binaire, étape par étape (dessinez toutes les étapes).

5, 1, 7, 6, 4, 8, 2, 3

- b) On retire l'élément se trouvant à la racine du tas. Dessinez les modifications du tas induites par cette opération, étape par étape.
- c) Comment peut-on implémenter un arbre binaire avec un tableau ? En particulier, comment trouver les fils et le père ?

Problème 3 : algorithmique et Python (5+10+10 = 25 points)

Le *tri comptage* (*counting sort* en anglais), appelé aussi *tri casier*, est un algorithme de tri qui s'applique sur des valeurs entières.

On considère que l'index des tableaux commence à 0. Le signe = est utilisé pour les affectations. Le tableau `tab` est le tableau à trier, et est passé en paramètre de la fonction `triParComptage`. La variable `borneSupérieure` est la valeur entière maximale présente dans `tab`.

La fonction `triParComptage` utilise des variables intermédiaires :

- `tabComptage`, est un tableau contenant N éléments, N étant la valeur maximale dans `tab`.
- `i` et `j` sont des variables de type entier, servant à parcourir les tableaux `tab` et `tabComptage`.

```
fonction triParComptage(tab) :  
  
// étape 1  
borneSupérieure = 0  
Pour tout k de tab  
    si k > borneSupérieure alors  
        borneSupérieure = k  
    finSi  
finPour  
  
// étape 2  
Pour i = 0 à borneSupérieure faire  
    tabComptage[i] = 0  
finPour  
  
// étape 3  
Pour tout k de tab faire  
    tabComptage[k] = tabComptage[k] + 1  
finPour  
  
// étape 4  
N = taille de tabComptage  
cpt = 0  
Pour i = 0 à N faire  
    Pour j = 0 à tabComptage[i] faire  
        tabTrié[cpt] = i  
        cpt = cpt + 1  
    finPour  
    écrire(tabTrié) // pour la question b)  
finPour  
  
retourne tabTrié
```

- Commentez succinctement les 3 premières étapes de l'algorithme ci-dessus et plus en détails l'étape 4.
- Appliquez cet algorithme pour trier la liste [2, 4, 5, 5, 6, 6, 5, 1, 2]. Écrivez le résultat de chacune des 4 étapes.
- Traduisez cet algorithme en Python 3.

Problème 4 : base de données (2+3+3+5+5+7 = 25 points)

La base de données d'un hôtel contient trois tables : « Chambre », « Client » et « Reservation ». Elles sont définies comme suit :

Chambre (Num_Chambre, Prix, Nbr_Lits, Nbr_Pers, Confort, TV)

Client (Id_Client, Nom, Prenom, Adresse)

Reservation (Id_Client, Num_Chambre, Date_Arr, Date_Dep)

<i>Chambre</i>					
Num_Chambre	Prix	Nbr_Lits	Nbr_Pers	Confort	TV
11	80	1	2	WC	Non
12	100	2	2	Douche	Non
13	180	3	3	Bain	Oui
...	...	...	...	...	...

<i>Client</i>			
Id_Client	Nom	Prenom	Adresse
1001	Bon	Jean	Bayonne
1002	Fonfec	Sophie	Genève
...	...	...	...

<i>Reservation</i>			
Id_Client	Num_Chambre	Date_Arr	Date_Dep
1001	11	09/07/2025	21/07/2025
1002	14	31/07/2025	Null
...	...	...	...

Exprimez les requêtes suivantes en SQL. On veut connaître :

1. Les numéros des chambres avec TV.
2. La capacité théorique d'accueil de l'hôtel.
3. Le nombre de chambres dont le prix se situe entre 85 et 120 CHF (inclus).
4. Les noms et prénoms des clients n'ayant pas fixé la date de départ.
5. L'adresse de tous les clients ayant réservé la chambre 11.
6. Le nombre de lits dans la chambre réservée par Jean Bon.