

RÉPUBLIQUE TUNISIENNE ----- MINISTÈRE DE L'ÉDUCATION	EXAMEN DU BACCALAURÉAT SESSION 2026	
	Épreuve Pratique : Algorithmique et Programmation	
	Section : Sciences de l'informatique	
	Coefficient de l'épreuve : 1	Durée : 1h 30 mn

Le sujet comporte 5 pages numérotées de 1/5 à 5/5

Important :

- 1) Toutes les ressources à utiliser se trouvent dans le dossier "Ressources" situé sur la racine du disque C.
- 2) Il est demandé au candidat :
 - De créer, dans le dossier Bac2026 situé sur la racine du disque C, un dossier de travail portant son numéro d'inscription (6 chiffres) et dans lequel il doit enregistrer, au fur et à mesure, tous les fichiers solution de ce sujet.
 - De copier, dans son dossier de travail, le fichier "Algo2.rar" situé dans "C:\Ressources", puis d'extraire son contenu, en utilisant le mot de passe "25ezaw", dans ce même dossier de travail.
 - De s'assurer que le contenu extrait est formé des fichiers "InterfRech.ui", "InterfRempl.ui", "Recherche.py" et "RempliM.py".
 - De créer tous les nouveaux fichiers dans son dossier de travail et de vérifier à la fin de l'épreuve que tous les fichiers s'y trouvent.

Dans un lycée, des observations précédentes ont montré que certaines zones de la cour pendant la récréation présentent une forte concentration d'élèves pouvant provoquer des tensions.

Pour améliorer la surveillance, un relevé du nombre d'élèves présent est modélisé sous forme d'une grille rectangulaire, où chaque case contient le nombre d'élèves présent à un emplacement donné, constituant ainsi une matrice d'entiers.

En analysant cette matrice, on cherche à repérer automatiquement certaines positions particulières pour mieux organiser la circulation des élèves et éviter les tensions et les problèmes. Ces positions sont appelées **points croissants**.

Un point croissant d'une matrice **M** de **L** lignes et **C** colonnes est tout élément **M[i,j]**, qui vérifie la relation suivante :

$$M[i,j] = M[i,j-1] + M[i-1,j] + M[i+1,j]$$

Avec :

- $1 \leq i \leq L-2$ (on exclue la première et la dernière ligne : ligne 0 et ligne L-1)
- $1 \leq j \leq C-1$ (on exclue la première colonne : colonne 0)

Exemple :

Soit la matrice **M** suivante :

		0	1	2	3	4
M	0	12	5	5	2	7
	1	4	13	3	2	18
	2	6	4	8	16	9
	3	1	11	10	6	2

Les points croissants existant dans la matrice **M** sont :

- **13** car $13 = 4 + 5 + 4$
- **18** car $18 = 2 + 7 + 9$
- **16** car $16 = 8 + 2 + 6$

On souhaite développer une application Python munie de deux interfaces graphiques permettant :

- de saisir les dimensions **L** et **C** et de remplir la matrice **M**,
- de rechercher et d'afficher les points croissants contenus dans la matrice **M** ainsi que leur nombre,
- de rechercher et d'afficher les zones sous-occupées dans la matrice **M** ainsi que leur nombre, sachant qu'une zone sous-occupée est une case de la matrice **M** dont la valeur est inférieure ou égale à un seuil **S**.

Travail demandé :

- 1- En se basant sur la figure **Fig.1** de l'**annexe**, compléter l'interface graphique "**InterfRempl.ui**" par les éléments manquants (un **label** et un objet **TextEdit**).
- 2- Apporter les modifications nécessaires au fichier "**RempliM.py**" afin de réaliser les actions suivantes :
 - a. Développer le module **Valide(L, C)** qui permet de vérifier la validité de **L** et **C**, sachant que $2 \leq L \leq 100$ et $2 \leq C \leq 100$.
 - b. Développer le module **Remplissage** qui s'exécute suite au clic sur le bouton "**Enregistrement de la matrice**" et qui permet :
 - **de vérifier la validité** des dimensions **L** et **C** en faisant appel au module **Valide**. Un message d'erreur doit être affiché via une boîte de dialogue **QMessageBox** si l'une des contraintes sur **L** ou sur **C** n'est pas respectée (voir figure **Fig.2**).
 - **de vérifier la conformité** du contenu saisi dans la zone **TextEdit** :
 - Le contenu saisi doit contenir exactement **L** lignes.
 - Chaque ligne doit contenir exactement **C** entiers naturels, tels que chaque deux entiers consécutifs sont séparés par un seul espace.
 - Afficher un message d'erreur via une boîte de dialogue **QMessageBox** si l'une des conditions précédentes n'est pas respectée (voir figure **Fig.3**).
 - **d'enregistrer les données** dans un fichier texte nommé "**cour.txt**", si toutes les conditions sont validées, selon le format suivant :
 - La première ligne du fichier contient les valeurs de **L** et de **C**, séparées par un espace.
 - Les lignes suivantes contiennent le contenu saisi dans la zone **TextEdit**, sachant que chaque ligne de cette zone correspond à une ligne du fichier.
 - c. Compléter la partie exploitation de l'interface graphique "**InterfRempl.ui**" par les noms adéquats afin :
 - **de charger** l'interface graphique depuis le fichier "**InterfRempl.ui**",
 - **d'appeler** le module nommé **Remplissage** suite au clic sur le bouton intitulé "**Enregistrement de la matrice**".
- 3- En se basant sur la figure **Fig.4** de l'**annexe**, compléter l'interface graphique "**InterfRech.ui**" par les éléments manquants (un objet **LineEdit** et un **bouton**).
- 4- Apporter les modifications nécessaires au fichier "**Recherche.py**" afin de réaliser les actions suivantes :
 - a. Développer le module **RechAffPC** qui s'exécute suite au clic sur le bouton "**Recherche et affichage des points croissants**" et qui permet :

- **de remplir** la matrice **M** de **L** lignes et **C** colonnes à partir du fichier texte "**cour.txt**". On rappelle que la première ligne du fichier texte contient les valeurs de **L** et de **C**, séparées par un espace, et que les lignes suivantes contiennent le contenu de la matrice,
 - **de déterminer et d'afficher**, dans la zone **TextEdit**, les valeurs et les indices des points croissants de la matrice **M** ainsi que leur nombre total. Chaque valeur d'un point croissant est affichée sur une ligne suivie de son indice de ligne, puis son indice de colonne, séparés par un espace (voir figure **Fig.5**),
 - **d'enregistrer** les valeurs et les indices des points croissants dans un fichier texte nommé "**PointsCroissant.txt**". Chaque ligne du fichier doit contenir la valeur du point croissant, suivie de son indice de ligne dans la matrice, puis de son indice de colonne, séparés par un espace.
- b.** Développer le module **RechAffZones** qui s'exécute suite au clic sur le bouton "**Recherche et affichage des zones sous occupées**" et qui permet :
- **de récupérer** la valeur du seuil **S** à partir de la zone **LineEdit**. Le seuil **S** doit être un entier positif inférieur ou égal à **5**. En cas de non-respect de cette contrainte, afficher un message d'erreur via une boîte de dialogue **QMessageBox** (voir figure **Fig.6**),
 - **de déterminer et d'afficher**, dans la zone **TextEdit**, les valeurs et les indices des zones sous-occupées de la matrice **M**, ainsi que leur nombre total (voir figure **Fig.7**).
N.B : On rappelle qu'une zone sous-occupée est une case de **M** dont la valeur est inférieure ou égale au seuil **S**.
- c.** Développer le module **Effacer** qui s'exécute suite au clic sur le bouton "**Effacer**" et qui permet d'effacer le contenu de la zone **TextEdit**.
- d.** Compléter la partie exploitation de l'interface graphique "**InterfRech.ui** " par les noms adéquats afin :
- **de charger** l'interface graphique depuis le fichier "**InterfRech.ui** ",
 - **d'appeler** le module nommé **RechAffPC** qui s'exécute suite au clic sur le bouton intitulé " **Recherche et affichage des points croissants**",
 - **d'appeler** le module nommé **RechAffZones** qui s'exécute suite au clic sur le bouton intitulé " **Recherche et affichage des zones sous occupées** ",
 - **d'appeler** le module nommé **Effacer** qui s'exécute suite au clic sur le bouton intitulé "**Effacer**".

Annexe :

Fig.1

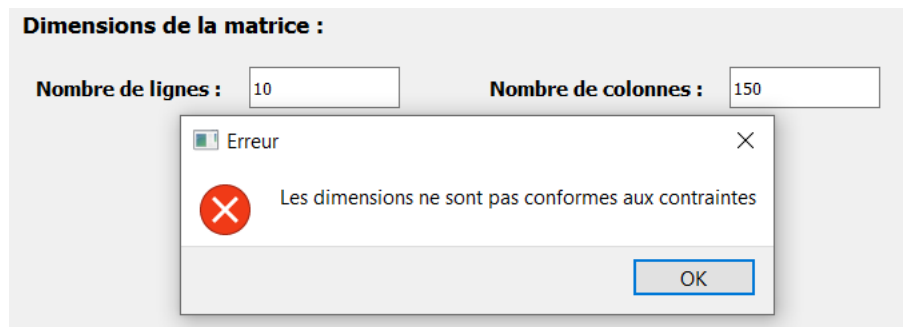


Fig.2

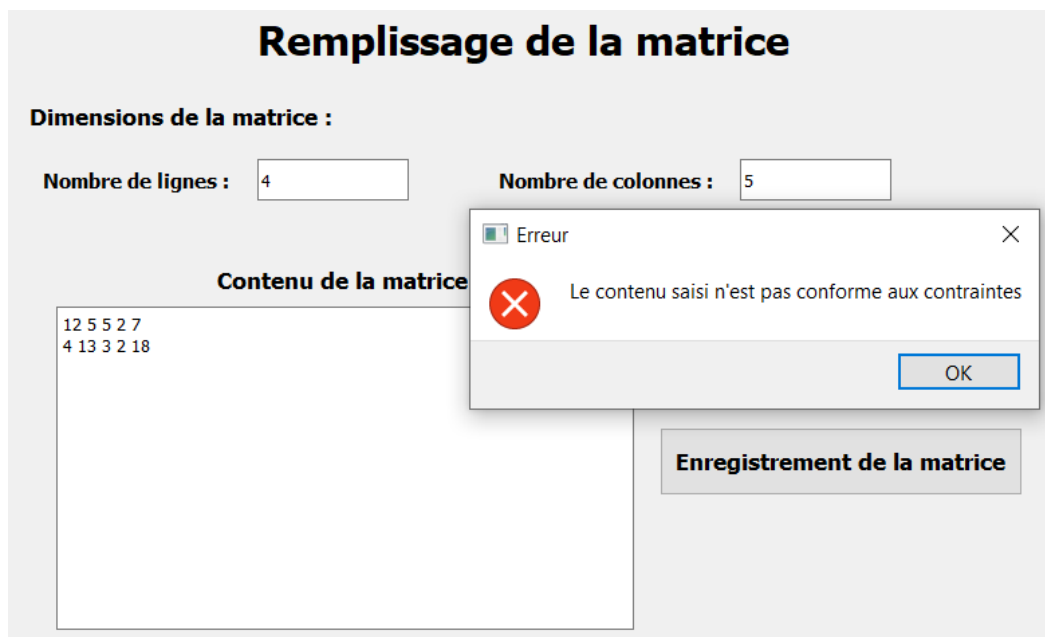


Fig.3

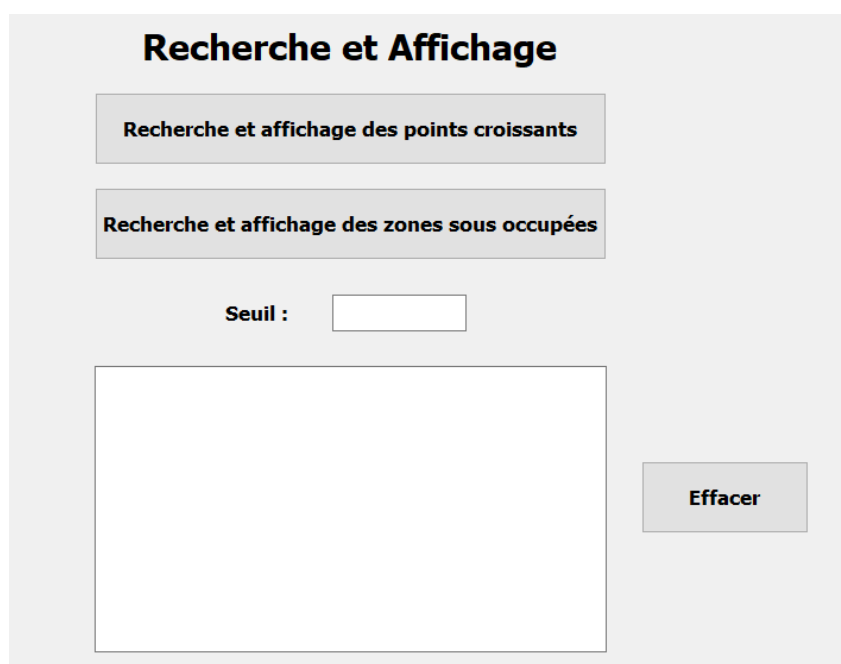


Fig.4



Fig.5

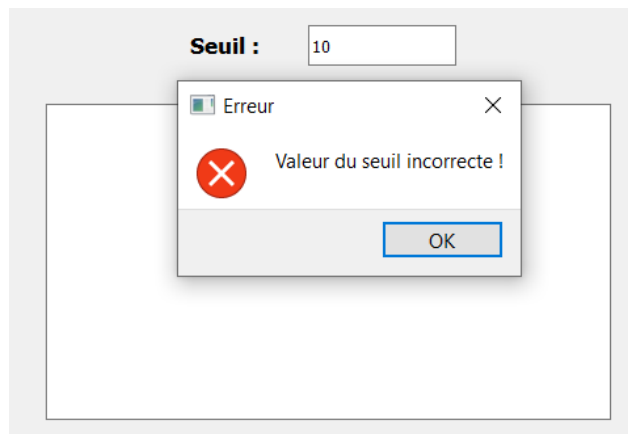


Fig.6

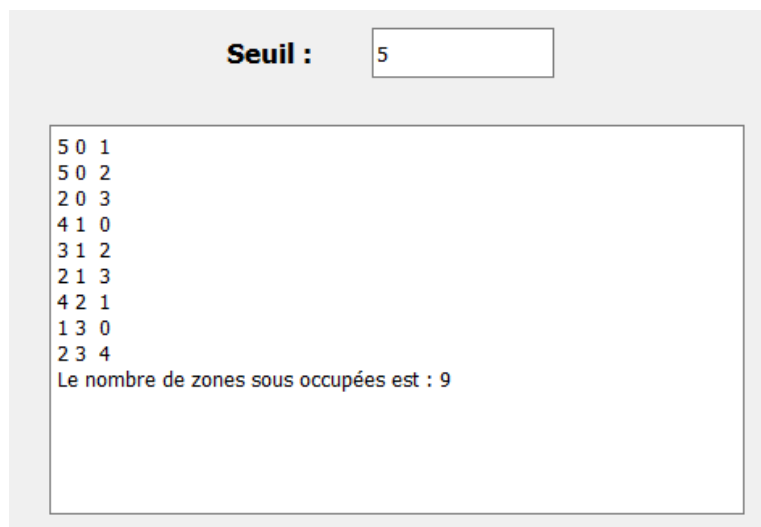


Fig.7

Grille d'évaluation :

	Traitement	Nombre de points
1	Complétion de l'interface graphique "InterfRempl.ui"	1
2	Modification dans "RempliM.py"	8,25
3	Complétion de l'interface graphique "InterfRech.ui"	1
4	Modification dans " Recherche.py "	9,75