

Énoncé du projet 2018-2019

Consignes générales

- Le projet est **personnel**. Toute utilisation de ressources non personnelles (cours, TD, web, camarades, etc.) doit donner lieu à citation. Si vous discutez avec des camarades, ne les copiez pas.
 - Vos initiales devront figurer en début de chacun de vos noms de fichiers.
 - Vos nom et prénom devront apparaître en premier dans chacun des fichiers que vous rendrez.
 - Chaque début de fichier devra contenir un commentaire précisant ce qu'il contient par rapport aux attendus du projet.
 - Choisissez des noms explicites pour vos fichiers, vos fonctions, vos variables.
 - Pour chaque fonction écrite, vous proposerez un programme montrant les tests que vous avez effectués pour vérifier le bon fonctionnement de la fonction. Si le programme inclut une interaction avec des utilisateurs, décrivez brièvement en commentaire ou dans un fichier annexe les principaux tests effectués. Le correcteur devra pouvoir exécuter les tests sans modification des codes fournis.
 - Écrivez les spécifications des fonctions. Sans paraphraser le code, n'hésitez pas à ajouter tout commentaire que vous jugerez utile pour la compréhension de votre travail.
-

Lecture fortement conseillée : site de sensibilisation au plagiat de l'université Paul-Valéry Montpellier 3 (qu'est-ce que c'est ? Comment l'éviter ? Les sanctions en cas de plagiat...)

Critères principaux de notation

- Respect des consignes générales du projet.
- Respect des délais (les dépassements trop importants entraîneront la non-considération des fichiers remis).
- Fonctionnement des programmes (les programmes ne s'exécutant pas ne seront pas considérés).
- Respect des consignes générales du projet (spécifications/commentaires de fonctions, pas d'usage de l'instruction **return** dans le corps d'une instruction répétitive, développement en considérant que les expressions booléennes ne sont pas évaluées de manière paresseuse, ...). Comme au premier semestre, les fonctions sans spécifications seront lourdement sanctionnées.
- Choix des bons algorithmes, avec les bonnes complexités en temps.

La note de projet pourra aussi être déterminée en fonction d'informations complémentaires fournies soient de manière volontaires (mini-rapport, commentaires dans les fichiers, ...) ou oralement sous forme de mini-soutenance à la demande des enseignants.

L'**objectif général** du projet est la mise en place de programmes de traitement de données.

1 Choix des données

Choisissez sur internet un jeu de données ouvertes disponible sous forme d'un fichier au format csv d'au moins 100 enregistrements (lignes du CSV). Les enregistrements du jeu de données devront avoir au moins un attribut non numérique et au moins deux attributs numériques (Un attribut sera considéré numérique si cela a du sens de réaliser du calcul numérique dessus (par exemple un numéro d'étudiant n'est pas une donnée numérique)).

Envoi du choix

Pour validation de votre choix, envoyez un mail à gwenael.richomme@univ-montp3.fr avec :

- vos nom et prénom ;
- l'URL à laquelle le jeu de données est décrit et peut être téléchargé au format CSV ;
- dans le cas où l'URL ne permet pas de savoir directement que les données sont ouvertes, des informations permettant de vérifier que les données sont ouvertes ;
- le nom des 3 à 5 attributs des données avec lesquels vous travaillerez (dont au moins un non numérique et au moins deux numériques).

Date limite de définition de votre jeu de données : **mercredi 6 février 8h15**

Attention! Chaque étudiant devra travailler à partir d'un jeu de données différent de ceux des autres étudiants. La validation des jeux de données se fera selon le fonctionnement d'une file d'attente (algorithme FIFO, First In First Out, selon principe du premier arrivé premier servi). En conséquence, envoyez votre premier choix bien avant le 6 février.

2 Récupération et préparation des données

- En utilisant un tableur, préparez 4 fichiers au format CSV contenant respectivement 5, 10, 50 et 100 lignes du jeu de données avec uniquement les attributs retenus. C'est essentiellement avec ces fichiers que vous travaillerez pour tester vos programmes.

3 Lecture de fichier

1. Créez un fichier Python (fichier 1) contenant une déclaration de nouveau type de données (appelé `votre_type` dans la suite de ce sujet). Chaque ligne de votre jeu de données pourra être représentée par une valeur de votre type.
2. Ajoutez dans le fichier 1, une fonction permettant d'afficher une instance d'enregistrement de type `votre_type`.
3. Écrivez un programme de test (fichier 2) .
4. Dans un fichier dédié à la gestion de listes de valeurs de type `votre_type` (fichier 3), écrivez une fonction permettant d'afficher une liste de valeurs de type `votre_type`.
5. Écrivez un programme de test de la fonction écrite précédemment (fichier 4)
6. Dans le fichier 3, écrivez une fonction de lecture d'un fichier retournant une liste de valeurs de type `votre_type`.
7. Écrivez un programme de test permettant de tester la fonction précédente à partir des fichiers créés dans la partie 2.

Indication. Vous pourrez vous inspirer des programmes fournis en exemple lors du premier cours. Consultez également la documentation sur la gestion de fichiers fournie sur Moodle.

4 Remise 1

Une tâche Moodle vous permettra de déposer au plus tard le **vendredi 15 février, 13h** une archive compressée contenant l'ensemble des fichiers créés en réponse aux parties 2 et 3 (les fichiers pourront être organisés dans une arborescence).

5 Écriture de fichier (partie bonus)

- Dans le fichier 3, écrivez une fonction permettant d'enregistrer une liste de valeurs type `votre_type` dans un fichier au format CSV.
- Écrivez un programme testant cette fonction (fichier 5).

6 Calcul d'indicateurs statistiques

- Choisissez un attribut numérique. Écrivez des fonctions permettant, de calculer pour une liste de valeurs de type `votre_type` le minimum, le maximum, la moyenne et l'écart type de cet attribut.
- Pour chacune des fonctions précédentes, écrivez un programme de test (tests sur plusieurs cas dont des cas extrêmes; n'hésitez pas à créer des jeux de tests spécifiques sur lesquels vous êtes capable de vérifier les calculs).
- Écrivez un programme permettant d'afficher les indicateurs statistiques calculés par les fonctions précédentes sur les jeux de données de la partie 2.

7 Remise 2

Une tâche Moodle vous permettra de déposer au plus tard le **vendredi 1 mars 13h** une archive compressée contenant l'ensemble des fichiers créés en réponse à l'ensemble des parties 2 à 6 (les fichiers pourront être organisés dans une arborescence – les modifications éventuelles dans des fichiers déjà rendus devront être signalées et expliquées).

8 Tris et applications

- Mettre en place un tri par sélection du maximum (vu en TD 1) : les données seront triées selon les valeurs (croissantes) de l'attribut numérique utilisé dans la partie 6 (dans la suite, par *liste triée*, nous entendons liste triée selon le critère précédent).
- Écrivez une fonction permettant de calculer, en temps constant, la médiane des valeurs d'une liste supposée triée.
- Écrivez une fonction permettant, en temps linéaire, de calculer les modalités de l'attribut numérique des valeurs d'une liste supposée triée.
- Écrivez une fonction permettant, en temps linéaire, de calculer les modes (valeurs prises le plus de fois) de l'attribut numérique des valeurs d'une liste supposée triée.
- Pour chacune des fonctions précédentes, écrivez un programme de test (tests sur plusieurs cas dont des cas extrêmes).
- Complétez le programme écrit dans la partie 6 en y ajoutant la médiane et les modes.

9 Remise 3

Une tâche Moodle vous permettra de déposer au plus tard le **vendredi 15 mars 13h** une archive compressée contenant l'ensemble des fichiers créés en réponse à l'ensemble des parties parties 2 à 6 (les fichiers pourront être organisés dans une arborescence – les modifications éventuelles dans des fichiers déjà rendus devront être signalées et expliquées).

10 Recherche dichotomique

- Écrivez une fonction, de complexité logarithmique, permettant de connaître un indice d'une occurrence d'une valeur comme attribut numérique (le même que dans la partie 6) des éléments d'une liste supposée triée.
- Écrivez une fonction, de complexité logarithmique, permettant de connaître le plus petit indice d'une occurrence d'une valeur comme attribut numérique (le même que dans la partie 6) des éléments d'une liste supposée triée.
- Écrivez une fonction, de complexité logarithmique, permettant de connaître le plus grand indice d'une occurrence d'une valeur comme attribut numérique (le même que dans la partie 6) des éléments d'une liste supposée triée.
- En utilisant des fonctions parmi les précédentes, écrivez une fonction permettant de déterminer, en temps logarithmique, le nombre d'occurrences d'une valeur comme attribut numérique (le même que dans la partie 6) des éléments d'une liste supposée triée.
- Écrivez un programme permettant de tester les fonctions précédentes.

11 Gestion de données (bonus)

En vous appuyant sur les fonctions écrites dans les parties précédentes, écrivez un programme proposant les traitements suivants : lecture et écriture de fichier, affichage, tri, insertion, recherche, suppression, calculs d'indicateurs statistiques, ...

12 Remise 4

Une tâche Moodle vous permettra de déposer au plus tard le **vendredi 29 mars 13h** une archive compressée contenant l'ensemble des fichiers créés en réponse à l'ensemble des parties parties 2 à 6 (les fichiers pourront être organisés dans une arborescence – les modifications éventuelles dans des fichiers déjà rendus devront être signalées et expliquées).

13 Barème de la note de projet

Rappel : la note de projet compte pour 4 points sur la note finale de première évaluation (chaque examen compte pour 8 points). Elle est essentiellement basée sur les remises du projet mais peut prendre en compte des éléments rendus en TD à la demande de l'enseignant.

- Chacune des 4 remises sera notée sur 1 point.
- Les bonus de projets seront au maximum de 0,2 point.
- Un bonus, d'au maximum 1 point, sera calculé à partir des remises correctes faites de devoirs à la maison proposés dans le cadre des TDs.