

TD3: Collection, Dictionnaire

PC0, L2 Informatique UBO

9 février 2022

But : Collection, Ensemble, Bag, Dictionnaire

1 Un ensemble d'entiers, sac d'entiers

1.1 Ensemble

On définit une classe représentant un ensemble d'entiers (notée **intSet**). Dans cette classe, les éléments ne peuvent être présents qu'une seule fois. On ne peut pas avoir plusieurs occurrences d'un même élément. Cette classe est définie par une variable d'instance (attribut de données) noté *vals* qui est une liste. On définit les opérations suivantes sur cette classe :

1. insertion d'un objet (méthode **insert**) : si l'objet ne se trouve pas dans l'ensemble, il y est rajouté
2. appartenance (méthode **member**) : on teste l'appartenance de l'objet à l'ensemble
3. suppression d'un objet de l'ensemble (méthode **remove**), si l'objet n'est pas présent une erreur est générée
4. affichage textuel de la liste des objets appartenant à l'ensemble. Les éléments de l'ensemble sont affichés dans l'ordre croissant de leur valeur.

Tester les différentes opérations de votre classe avec des exemples.

1.2 Bag

On définit une classe comme un sac (Bag) où on peut conserver pour chaque élément son nombre d'occurrences. Cette information est stockée sous forme de dictionnaire en interne. Cette classe a deux variables d'instance : la collection des éléments et le dictionnaire des doublons qui prend en clé l'élément et en valeur son nombre d'occurrences. Définir la classe **intBag** qui est un Bag d'entiers. On définit les opérations suivantes sur cette classe :

1. insertion d'un objet (méthode **insert**) : l'objet est rajouté et le dictionnaire des occurrences est mis à jour
2. appartenance (méthode **member**) : on teste l'appartenance de l'objet au sac
3. suppression d'un objet de l'ensemble (méthode **remove**), si l'objet apparaît avec plusieurs occurrences, on met à jour le dictionnaire des occurrences sinon l'objet est supprimé
4. affichage textuel de la liste des objets appartenant au sac en affichant aussi le dictionnaire des occurrences. Les éléments de la liste sont affichés dans l'ordre croissant de leur valeur.

Tester les différentes opérations de votre classe avec des exemples.

2 Groupes, notes, histogrammes

On définit un ensemble d'étudiants appartenant à une filière représentés par leur nom (défini par une chaîne de caractères). Ces étudiants sont répartis de manière homogène dans trois groupes (notés groupe1, groupe2, groupe3) en respectant l'ordre alphabétique.

1. Définir une collection d'étudiants représentés par leur nom qui est donné sous forme de chaînes de caractères.
2. Trier cette collection dans l'ordre alphabétique des noms.
3. Trouver la taille de cette collection
4. Ecrire une fonction qui prend en paramètre un nombre i et retourne le nom du groupe ('groupe*i*'). Cette fonction permettra de définir le nom des 3 groupes (groupe1, groupe2, groupe3).
5. Définir le dictionnaire des groupes qui prend en clé le nom des groupes (nom défini par la fonction de la question précédente) et en valeur une collection (vide pour l'instant) des étudiants correspondant.
6. Casser la collection d'étudiants en groupes, et remplir le dictionnaire des groupes.
7. On possède trois tableaux de notes (de 0 à 10) correspondant aux notes des étudiants données dans l'ordre alphabétique. Chacun des tableaux correspond à une évaluation différente. Définir à partir de ces trois tableaux le dictionnaire des notes qui prend en clé le nom de l'étudiant et en valeur le tableau comportant les trois notes correspondant aux différentes évaluations. Quel est l'étudiant de la filière ayant la meilleure moyenne pour l'ensemble de l'évaluation ?
8. Dresser à partir de ce dernier dictionnaire un histogramme des notes pour l'ensemble de la filière ou par groupe et pour l'ensemble des évaluations ou pour chacune des évaluations. Quelle est la note la plus fréquente pour l'ensemble des évaluations ? On pourra passer par la classe `intBag` pour répondre à cette question.

Pour afficher un histogramme, on peut utiliser le code suivant :

```
from matplotlib import pyplot
x = [1, 2, 2, 3, 4, 4, 4, 4, 5, 5]
pyplot.hist(x, range = (0, 5), bins = 5, color = 'yellow',
edgecolor = 'red')
pyplot.xlabel('valeurs')
pyplot.ylabel('nombres')
pyplot.title('Exemple d\' histogramme simple')
```

Pour plus de détails sur l'affichage graphique, vous pouvez consulter la référence :

<http://www.python-simple.com/python-matplotlib/histogram.php>