



Modalités

- ✓ Durée : 2h.
- ✓ Vous pouvez utiliser une feuille de notes manuscrites (écrites à la main non photocopie) recto-verso de taille quelconque.
- ✓ Rendez l'annexe jointe si vous l'utilisez

I. Situation préliminaire

Il s'agit de décrire des recettes de cuisine.

Une **recette** de cuisine comprend un **titre**, une **liste d'étapes**, ainsi qu'un **nombre de parts** pour lesquels la quantité d'ingrédients de la recette convient. Par exemple, *pour 6 personnes*.

Un **ingrédient** est représenté par son **nom**, une **unité** et un **nombre de calories** associées à cette unité : par exemple *lait entier, litre et 586* (il y a 586 calories dans un litre de lait) ou *sucre, gramme et 3,87* (3,87 calories pour un gramme de sucre).

Une **étape** nécessite un ou plusieurs **ingrédients quantifiés**, c'est-à-dire des ingrédients associés chacun à une certaine **quantité** en fonction de leur unité : par exemple *1,5 litre de lait entier*, ou *125 grammes de sucre et 250 grammes de farine*.

Elle contient la **description textuelle d'une action** : par exemple *faire bouillir* (le lait) ou *mélanger* (la farine et le sucre).

Pour répondre aux questions de tout l'examen, vous disposez d'une classe outil¹ **LesQuantities** dont chaque instance représente une collection d'ingrédients quantifiés (Elle sera vue comme telle dans le diagramme). Cette classe possède entre autres les méthode suivantes :

- ✓ **IngredientQuantifie** `get(int i)` :
Restitue la référence à l'ingrédient quantifié de rang `i` dans la liste ou `null` si `i` est hors limites.
- ✓ **double** `getTaille()` :
Restitue le nombre d'ingrédients quantifiés dans la collection.

Questions

1. Faites figurer dans le premier diagramme de l'annexe les relations entre classes qui rendent compte de l'énoncé.
2. Ajoutez à ce diagramme les déclarations des attributs nécessaires (Les collections pourront être représentées par des tableaux).
3. En supposant que les accesseurs utiles soient déjà écrits, écrivez la méthode `getCa-`

¹ Vous n'avez pas à l'écrire.

`lories()` de la classe **IngredientQuantifie**. Pour chaque instance de cette classe, cette méthode restitue son nombre total de calories en fonction du nombre de calories de l'ingrédient et de la quantité qui lui sont associées/e/s. Par exemple, en fonction des valeurs vues plus haut, pour 1,5 litres de lait entier, il y a $1,5 \times 586 = 879$ calories, pour 125 grammes de sucre, il y en a 483,75.

4. Écrivez de même la méthode `getCalories()` pour une instance d'étape² et une instance de recette³.

5. Écrivez les méthodes `toString` de :

- ✓ **Ingredient**. Par exemple, pour du lait avec litre comme unité, cela peut donner :
 lait : 1 litre, 586.0 calories.
- ✓ **IngredientQuantifie**. Par exemple, pour 1,5 litre de lait, cela donne : **lait** : 1.5 litre. Dans ce cas, on ne conserve pas le nombre de calories qui est obtenu par ailleurs, grâce à `getCalories()`.
- ✓ **LesQuantifies**. Par exemple, pour 1,5 litre de lait et 125 grammes de sucre, cela donnerait :
 lait : 1.5 litre
 Sucre : 125.0 gramme

II. Situation complète

Certaines étapes, dites **complexes**, sont décomposables en une **liste de sous-étapes** : des **étapes simples** — qui sont identiques aux étapes de la situation préliminaire — et, éventuellement, d'autres **étapes complexes**. Une recette peut alors être vue comme comportant une seule étape complexe, elle-même décomposable en sous-étapes.

Questions

1. Faites figurer dans le deuxième diagramme de l'annexe les relations entre classes qui rendent compte de ces nouvelles modalités.
2. Ajoutez à ce diagramme les déclarations des attributs nécessaires pour rendre compte de toutes les informations de l'énoncé dans cette nouvelle forme.
3. Sachant que le nombre de calories utilisées dans une étape complexe est la somme des calories utilisées dans les étapes qui la composent, indiquez précisément ce qu'il faut ajouter/modifier à chaque classe pour que `getCalories()` restitue une valeur correcte pour chaque type d'étape et pour la recette complète.

Nous avons vu que les quantités de chaque ingrédient dans les étapes d'une recette dépend du nombre de parts indiqué (par exemple, quantifiés pour 4 personnes). On considère désormais ce nombre de parts comme une valeur de référence affiché par défaut. Mais, si l'utilisateur choisit une autre valeur, il faut que le nombre de parts, les quantités réelles des ingrédients des différentes étapes ainsi que les calories associées soient modifiées en conséquence.

4. Sans écrire de programme, indiquez quoi modifier dans les classes pour respecter ces nouvelles spécifications.

² Le nombre de calories des ingrédients de l'étape.

³ Le nombre de calories total de l'ensemble des ingrédients de la recette (la somme des calories des étapes).

Annexe à rendre

I. Situation préliminaire

Etape

IngredientQuantifie

Recette

Ingredient

II. Situation complète

Recette

Etape

IngredientQuantifie

EtapeComplexe

EtapeSimple

Ingredient

N° d'anonymat :



Modalités

- ✓ Durée : 2h.
- ✓ Vous pouvez utiliser une feuille de notes manuscrites (écrites à la main non photocopie) recto-verso de taille quelconque.
- ✓ Rendez l'annexe jointe si vous l'utilisez

I. Situation préliminaire

Il s'agit de décrire des recettes de cuisine.

Une **recette** de cuisine comprend un **titre**, une **liste d'étapes**, ainsi qu'un **nombre de parts** pour lesquels la quantité d'ingrédients de la recette convient. Par exemple, *pour 6 personnes*.

Un **ingrédient** est représenté par son **nom**, une **unité** et un **nombre de calories** associées à cette unité : par exemple *lait entier, litre* et *586* (il y a 586 calories dans un litre de lait) ou *sucre, gramme* et *3,87* (3,87 calories pour un gramme de sucre).

Une **étape** nécessite un ou plusieurs **ingrédients quantifiés**, c'est-à-dire des ingrédients associés chacun à une certaine **quantité** en fonction de leur unité : par exemple *1,5 litre de lait entier*, ou *125 grammes de sucre* et *250 grammes de farine*.

Elle contient la **description textuelle d'une action** : par exemple *faire bouillir* (le lait) ou *mélanger* (la farine et le sucre).

Pour répondre aux questions de tout l'examen, vous disposez d'une classe outil¹ `LesQuantifiés` dont chaque instance représente une collection d'ingrédients quantifiés (Elle sera vue comme telle dans le diagramme). Cette classe possède entre autres les méthodes suivantes :

- ✓ `IngredientQuantifie get(int i)` :
Restitue la référence à l'ingrédient quantifié de rang *i* dans la liste ou *null* si *i* est hors limites.
- ✓ `double getTaille()` :
Restitue le nombre d'ingrédients quantifiés dans la collection.

Questions

1. Faites figurer dans le premier diagramme de l'annexe les relations entre classes qui rendent compte de l'énoncé.
2. Ajoutez à ce diagramme les déclarations des attributs nécessaires (Les collections pourront être représentées par des tableaux).
3. En supposant que les accesseurs utiles soient déjà écrits, écrivez la méthode `getCa-`

¹ Vous n'avez pas à l'écrire.

lories() de la classe `IngredientQuantifie`. Pour chaque instance de cette classe, cette méthode restitue son nombre total de calories en fonction du nombre de calories de l'ingrédient et de la quantité qui lui sont associé/e/s. Par exemple, en fonction des valeurs vues plus haut, pour 1,5 litres de lait entier, il y a $1,5 \times 586 = 879$ calories, pour 125 grammes de sucre, il y en a 483,75.

4. Écrivez de même la méthode `getCalories()` pour une instance d'étape² et une instance de recette³.
5. Écrivez les méthodes `toString` de :
 - ✓ `Ingredient`. Par exemple, pour du lait avec litre comme unité, cela peut donner :
Lait : 1 litre, 586.0 calories.
 - ✓ `IngredientQuantifie`. Par exemple, pour 1,5 litre de lait, cela donne : Lait : 1.5 litre. Dans ce cas, on ne conserve pas le nombre de calories qui est obtenu par ailleurs, grâce à `getCalories()`.
 - ✓ `LesQuantifies`. Par exemple, pour 1,5 litre de lait et 125 grammes de sucre, cela donnerait :
Lait : 1.5 litre
Sucre : 125.0 gramme

II. Situation complète

Certaines étapes, dites complexes, sont décomposables en une liste de sous-étapes : des étapes simples — qui sont identiques aux étapes de la situation préliminaire — et, éventuellement, d'autres étapes complexes. Une recette peut alors être vue comme comportant une seule étape complexe, elle-même décomposable en sous-étapes.

Questions

1. Faites figurer dans le deuxième diagramme de l'annexe les relations entre classes qui rendent compte de ces nouvelles modalités.
2. Ajoutez à ce diagramme les déclarations des attributs nécessaires pour rendre compte de toutes les informations de l'énoncé dans cette nouvelle forme.
3. Sachant que le nombre de calories utilisées dans une étape complexe est la somme des calories utilisées dans les étapes qui la composent, indiquez précisément ce qu'il faut ajouter/modifier à chaque classe pour que `getCalories()` restitue une valeur correcte pour chaque type d'étape et pour la recette complète.

Nous avons vu que les quantités de chaque ingrédient dans les étapes d'une recette dépend du nombre de parts indiqué (par exemple, quantités pour 4 personnes). On considère désormais ce nombre de parts comme une valeur de référence affichée par défaut. Mais, si l'utilisateur choisit une autre valeur, il faut que le nombre de parts, les quantités réelles des ingrédients des différentes étapes ainsi que les calories associées soient modifiées en conséquence.

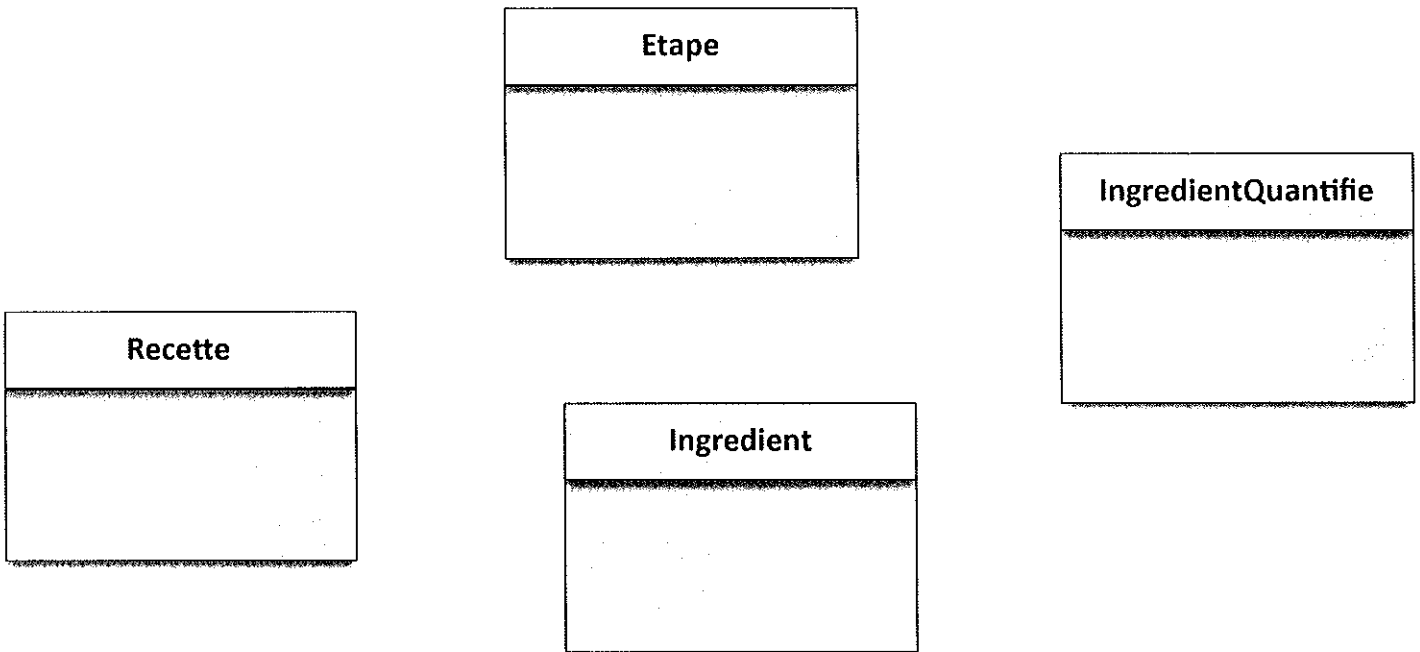
4. Sans écrire de programme, indiquez quoi modifier dans les classes pour respecter ces nouvelles spécifications.

² Le nombre de calories des ingrédients de l'étape.

³ Le nombre de calories total de l'ensemble des ingrédients de la recette (la somme des calories des étapes).

Annexe à rendre

I. Situation préliminaire



II. Situation complète

