

Regroupement juin 2023

Programmation fonctionnelle

Épreuve 1

Dans les différents exercices, vous pouvez, si besoin est, écrire des sous-fonctions pour vous simplifier le travail.

Soient deux listes plates, la première étant une liste plate de places boursières, la seconde une liste contenant des matières premières, construire récursivement une liste comme suit (chaque bourse a toutes les matières) :

liste_bourses : (bourse1 bourse2... bourseN)

liste_matières : (matière1 matière2... matièreN)

=> ((bourse1 (matière1) (matière2) ... (matièreN)) ... (bourse2 (matière1) (matière2)... (matièreN)))

On appellera cette liste : Cours-bourses. Cette partie de l'exercice est très simple pour ceux qui ont révisé, ce qui fait que c'est la suite qui compte vraiment.

Écrire une fonction avec comme argument Cours-bourses qui interroge l'utilisateur (avec un (read)) pour obtenir le prix de chaque matière dans chaque place boursière et qui construit la liste suivante :

=> ((bourse1 (matière1 . prix) (matière2 . prix)...) etc.

La fonction parcourt toutes les places boursières et toutes les matières, et, pour chaque couple place boursière – matière première, pose la question :

"Quel est le prix de <blé par exemple> à <Frankfort par exemple> ? (écrivez un nombre)"

Puis effectue un read de la réponse, vérifie que c'est bien un nombre, et insère (matière1 . prix) dans la nouvelle liste.

Astuce : pour poser la question, il faut utiliser la fonction (print <message>) ; vous pouvez imprimer avec cinq fois la fonction print (pour "Quel est le prix de", puis pour <matière première en cours>, puis pour "à", puis pour <place boursière en cours>, puis pour " ? (écrivez un nombre)"), ou alors construire la chaîne avec string-concat et l'imprimer avec un seul print. Faites un (terpri) à la fin pour aller à la ligne. Pour récupérer la réponse, il suffit de la fonction (read).

Conseil : écrivez une sous-fonction juste pour imprimer la question et récupérer la réponse.

Épreuve 2

Écrire une variante de la fonction finale de l'épreuve 1, qui modifie Cours-bourses au lieu de construire une nouvelle liste.