
Quick 3

Exercise 1:(10 pt)

Pour représenter une facture contenant plusieurs articles, on définit un type **facture** en CAML comme suit:

```
type facture = float list;;
```

1. Comment représenter la facture **f1** contenant un article à 1,50 €, un article à 42 € et deux articles à 39,90 € ?
2. Réaliser une fonction **tva**: `float -> float` qui calcule la TVA (on suppose un taux constant de 20%) pour un article.
Exemple: **tva** (1.0) = 0.2
3. Réaliser une fonction **tva_facture**: `facture -> float list` qui calcule la liste des TVA des entrées de la facture. On utilisera la fonction **tva**.
Exemple: **tva_facture** ([1.0; 2.0]) = [0.2; 0.4]
4. Réaliser une fonction **somme**: `float list -> float` qui calcule la somme des valeurs d'une liste. Exemple: **somme** ([1.0; 2.0]) = 3.0
5. Réaliser une fonction **total**: `facture -> float` qui calcule la somme de tous les prix plus la TVA.
Exemple: **total** ([1.0; 2.0]) = 3.6

Exercise 2:(10pt)

On représente une courbe 2D par une liste de points (couple de réels). Le but de cet exercice est d'implémenter des fonctions de bases pour les courbes.

1. Définir le type **point** et le type **courbe**
2. Donner le profil de la fonction **ajoutEnFin** qui ajoute en fin de courbe un point donné. Implémentez cette fonction.
3. Implémentez la fonction **supOrigine** qui supprime tous les points de coordonnées (0,0) d'une courbe
4. Généraliser **supOrigine** à **supPoint** qui prend en paramètre un point et une courbe et qui supprime toutes les occurrences de ce point dans la courbe.
5. Généraliser **supOrigine** à **supElem** qui supprime n'importe quelle occurrence d'un élément dans une alpha liste.