

mathématique et permet d'éviter de se demander, à chaque fois que l'on utilise une donnée, si elle est toujours là ou si elle a été consommée ou transformée par une autre fonction. En revanche, quand on travaille avec de grosses données, on ne peut pas les copier sans cesse. Par exemple, quand on met à jour une base de données, il faut accepter la disparition de l'ancien état de cette base de données.

### Exercice 6.6

Programmer une fonction qui supprime un élément d'une liste, en copiant son argument. Même question en modifiant son argument.

### Exercice 6.7 (Le crible d'Érathostène)

On considère des listes de couples formés d'un entier et d'un booléen indiquant si cet entier est rayé ou non.

1. Écrire une fonction qui raye tous les éléments multiples non triviaux d'un entier  $n$  dans une liste  $l$ .
2. Écrire une fonction qui produit la liste des entiers  $2 \dots 1000$  non rayés.
3. Écrire une fonction qui produit la liste des nombres premiers inférieurs à 1000.

### Exercice 6.8

Si  $T$  est un type muni d'une relation d'ordre, typiquement les entiers, on dit qu'une liste d'éléments de  $T$  est *ordonnée* si chaque élément de cette liste est inférieur au suivant.

1. Écrire une fonction qui teste si une liste est ordonnée.
2. Écrire deux fonctions qui insèrent et suppriment un élément dans une liste ordonnée.

### Exercice 6.9

On représente un polynôme par une liste de couples formés d'un degré et d'un coefficient, fonctionnelle et ordonnée par degrés croissants. Écrire une fonction qui calcule la somme de deux polynômes.