

Feuille 5 : mise en pratique du cours sur les tranches

Table des matières

Exercice 1

Exercice 2

Exercice 3

Exercice 4



Pour tous les algorithmes sur les tranches, vous devez prévoir qu'une tranche peut être vide...

Exercice 1

On veut décomposer une tranche d'entiers en deux tranches : la première contiendra ses éléments pairs, la seconde ses éléments impairs. Le programme principal initialisera une tranche d'entiers, la décomposera et affichera ces deux tranches.

Si, par exemple, la tranche initiale est initialisée avec les valeurs 22, 3, 9, 42, 66, 100, -11, le programme devra afficher :

```
pairs : [22 42 66 100]
impairs : [3 9 -11]
```

TXT



Le format `%v` de `fmt.Printf` permet de produire ces affichages

Exercice 2

On veut inverser une tranche d'entiers. Le programme principal initialisera une tranche d'entiers, l'inversera et affichera le résultat.

Si, par exemple, la tranche initiale est initialisée avec les valeurs 22, 3, 9, 42, 66, 100, -11, le programme devra afficher :

```
La tranche inversée vaut [-11 100 66 42 9 3 22]
```

TXT

Proposez deux versions de l'inversion :

- une qui inverse la tranche "sur place" (procédure)
- une qui produit une copie de la tranche inversée (fonction)

Exercice 3

On veut utiliser les tranches pour représenter des ensembles. Dans une telle tranche, il ne pourra donc pas y avoir d'éléments en double...

Le programme principal initialisera deux "tranches ensembles", produira et affichera leur union et leur intersection (qui doivent donc également être des "tranches ensembles")

Si, par exemple, les deux tranches ensembles sont initialisées à [12, 1, 3, 10] et [3, 10, 28], le programme devra afficher :

Union de [12 1 3 10] et [3 10 28] = [12 1 3 10 28]
Intersection de [12 1 3 10] et [3 28 10] = [3 10]

TXT



Vous pouvez également calculer la différence de deux "tranches ensembles" : il s'agit des éléments qui sont dans la première mais pas dans la seconde. Avec notre exemple, on obtiendrait [12 1]

Exercice 4

On veut fusionner deux tranches **triées**.

Le programme principal initialisera deux tranches **triées**, puis produira et affichera leur fusion.

Si, par exemple, les deux tranches triées sont initialisées à [10, 20, 30] et [1, 2, 3, 4, 25], le programme devra afficher :

Fusion de [10 20 30] et [1 2 3 4 25] = [1 2 3 4 10 20 25 30]

TXT

Last updated 2023-02-06 21:24:11 +0100