

# Parcours complet [tb07] – Exercice résolu

Karine Zampieri, Stéphane Rivière

Unisciel  algoprogram  Version 19 mai 2018

## Table des matières

<b>1</b>	<b>Parcours complet / pgcomplet</b>	<b>2</b>
1.1	Fonction saisirTab (saisie d'un tableau) . . . . .	2
1.2	Fonction sommeTab (somme d'un tableau) . . . . .	3
1.3	Programme de test . . . . .	4
<b>2</b>	<b>Références générales</b>	<b>5</b>

## C - Parcours complet (Solution)



**Mots-Clés** Tableau unidimensionnel ■

**Utilise** Définitions et notations, Tableaux et paramètres, Parcours de tableaux ■

**Difficulté** ●○○ (30 min) ■

# 1 Parcours complet / pgcomplet



## Objectif

Cet exercice calcule la somme des éléments d'un tableau d'entiers.

Exemple :

```
[45, 54, 1, -56, 22, 134, 49, 12, 90, -27]
==> Somme des éléments est 324
```



## Définitions C

```
enum { TMAX = ... };
typedef int ITableau[TMAX];
```

## 1.1 Fonction saisirTab (saisie d'un tableau)



Écrivez le **profil** d'une fonction `saisirTab(t)` qui effectue la saisie du nombre de valeurs dans un entier, puis saisit les valeurs dans un `ITableau t` et renvoie le nombre de valeurs saisies.

### Solution Paramètres

Sortants : Un `ITableau t`

Résultat de la fonction : Un entier



### Saisie du nombre de valeurs

Le nombre de valeurs doit être compris entre 1 (inutile d'appeler cette fonction pour saisir un tableau vide) et `TMAX` (taille maximale d'un `ITableau`). On utilise donc une répétitive `TantQue` ou `Répéter` selon que l'on souhaite ou non avertir l'utilisateur d'une erreur de saisie.



Écrivez la saisie contrainte du nombre de valeurs entre 1 et `TMAX`.  
Affichez l'invite (où `[x]` désigne le contenu de `x`) :

```
Nombre de valeurs dans [1..[TMAX]]?
```



### Saisie des valeurs

Soit `n` le nombre de valeurs à saisir. Il faut répéter `n` fois la saisie d'une valeur (parcours complet). Par conséquent, le remplissage du tableau sera piloté par une boucle `Pour`.



Écrivez la saisie des valeurs entières dans `t`.  
Affichez une invite spécifiant le rang de l'élément à saisir :

```
t[..]?
```



Validez votre fonction avec la solution.

### Solution C @[UtilsTB.c]

```
int saisirTab(Tableau t)
{
    printf("Nombre de valeurs dans [1..%d]? ", TMAX);
    int n;
    scanf("%d", &n);
    while(!(1<=n && n<=TMAX))
    {
        printf("ERREUR dans [1..%d]? ", TMAX);
        scanf("%d", &n);
    }
    int ix;
    for (ix=0; ix<n; ++ix)
    {
        printf("t[%d]? ", ix);
        scanf("%d", &t[ix]);
    }
}
```

## 1.2 Fonction sommeTab (somme d'un tableau)



Écrivez le **profil** d'une fonction `sommeTab(t,n)` qui renvoie la somme des `n` premières valeurs d'un `ITableau t`.

### Solution Paramètres

**Entrants** : Un `ITableau t` et un entier `n`

**Résultat de la fonction** : Un entier (somme d'entiers)



Comment initialiser la somme à retourner ?

### Solution simple

A zéro comme dans le cas de la somme d'une séquence de valeurs.



Écrivez le corps de la fonction.

### Solution simple

On déclare une variable `somme` (le résultat) et on l'initialise à zéro. Ensuite on parcourt `t` par la boucle `Pour` (compteur `ix`) et on ajoute la valeur de l'élément `t[ix]` dans `somme`.



Validez votre fonction avec la solution.

**Solution C** @[UtilsTBOpers.c]

```
int sommeTab(const Tableau t,int n)
{
    int somme = 0;
    int k;
    for (k = 0 ; k < n ; ++k)
    {
        somme += t[k];
    }
    return somme;
}
```

### 1.3 Programme de test



Écrivez un programme qui définit la constante `TMAX=20`, demande et stocke des entiers dans un `Tableau` puis en calcule et affiche la somme des valeurs.



Testez. Exemple d'exécution :

```
Nombre d'éléments dans [1..20]? 10
t[0]? 45
t[1]? 54
t[2]? 1
t[3]? -56
t[4]? 22
t[5]? 134
t[6]? 49
t[7]? 12
t[8]? 90
t[9]? -27
==> Somme des éléments est 324
```



Validez votre programme avec la solution.

**Solution C** @[pgcomplet.c]

```
#include <stdio.h>

enum { TMAX=20 };
typedef int Tableau[TMAX];

#include "UtilsTB.c"
#include "UtilsTBOpers.c"

int main(void)
{
    Tableau tab;
    int nelems = saisirTab(tab);
    printf("==> Somme des elements est %d\n",sommeTab(tab,nelems));
}
```

## **2 Références générales**

Comprend ■