

# 1 Exercice 1

Ecrire les algorithmes permettant :

1. Le calcul du nombre d'occurrences d'un élément donné dans un tableau.
2. Le calcul de la moyenne et du minimum des éléments d'un tableau.
3. De tester si un tableau est trié.
4. Le calcul du produit scalaire de deux vecteurs réels  $u$  et  $v$  de dimension  $n$ :  $u.v = \sum_{i=1}^n u_i * v_i$

Traduire les algorithmes précédents en langage C.

Algorithme à nbOccurrence

$T[0..100]$  d'entier (contient des valeurs)  
taille : entier // taille  $\leq 100$

$i$  : entier ; // compteur de parcours (indices) de la valeur 0 par exemple.

nbOcc : entier ;

// recherche de nombre d'occurrence d'une valeur

Recherche valeur : entier

Ecrire ("Donner la valeur recherchée :");

Lire (valeur);

taille  $\leftarrow 20$ ;

nbOcc  $\leftarrow 0$ ;

taille = 7      taille - 1  
valeur = 12      -6  
nbOcc = 1      1

i	0	1	2	3	4	5	6	...	99	100 élts possible
	12	0	14	0	13	20	10	...		

un tableau de 7 entiers  
je recherche le nombre d'occurrences

de la valeur 0 par exemple.

Pour i  $\leftarrow 0$  à taille-1 faire :

Si ( $T[i] = valeur$ ) alors :

nbOcc  $\leftarrow$  nbOcc + 1;

Fin Si

i  $\leftarrow$  i + 1;

Fin Pour

Ecrire (" nb occurrences ");

Ecrire (nbOcc);

fin

# 1 Exercice 1

Ecrire les algorithmes permettant :

1. Le calcul du nombre d'occurrences d'un élément donné dans un tableau.
2. Le calcul de la moyenne et du minimum des éléments d'un tableau.
3. De tester si un tableau est trié.
4. Le calcul du produit scalaire de deux vecteurs réels  $u$  et  $v$  de dimension  $n$  :  $u.v = \sum_{i=1}^{i=n} u_i * v_i$

taille  
moy  
min  
Som  
 $i$   
 $T$

Traduire les algorithmes précédents en langage C.

Algorithme : moyEtMin

$T[0..taille]$ ; réel; taille: entier;  $i$ : entier  
min, S: réel // somme des élts  
 $S \leftarrow 0$ ;  $taille \leftarrow 7$ ;  $min \leftarrow T[0]$ ;

pour  $i \in 0 \text{ à } Taille - 1$

$S \leftarrow S + T[i]$ ;

    Si ( $T[i] < min$ ) alors:

$min \leftarrow T[i]$ ;

    fin si

$i \leftarrow i + 1$ ;

fin pour

    Ecrire (" la moyenne est ")

100 élts possible

12	0	14	0	13	20	10	...	99
----	---	----	---	----	----	----	-----	----

Ecrire ( $S / Taille$ ) ;

Ecrire (" le minimum est : ") ;

Ecrire ( $min$ ) ;

Fin

# 1 Exercice 1

Ecrire les algorithmes permettant :

1. Le calcul du nombre d'occurrences d'un élément donné dans un tableau.
2. Le calcul de la moyenne et du minimum des éléments d'un tableau.
3. De tester si un tableau est trié. *par ordre croissant*
4. Le calcul du produit scalaire de deux vecteurs réels  $u$  et  $v$  de dimension  $n$ :  $u \cdot v = \sum_{i=1}^{i=n} u_i * v_i$

Traduire les algorithmes précédents en langage C.

Algorithm : RstTrié

$T[0 \dots 100]$ ;  $i$ : entier;  $\text{trie}$ : booléen;  $\text{taille}$ : entier

debut  
     $\text{taille} \leftarrow 100;$   
    pour  $i \leftarrow 0$  à  $\text{taille} - 2$   
        si  $T[i] \leq T[i + 1]$   
             $i \leftarrow i + 1;$

    sinon

$\text{trie} = \text{vrai}$   
        Fin Pour //  $i \leftarrow i + 1$

    fini

Fin Pour //

suite

$\text{trie} \leftarrow \text{vrai}$

$i < 0$

$i = 2$

$i = 1$

$i = 2$

$i = 3$

taille

$i$

trié  $\leftarrow$  vrai

$T_1$  trié

trié  $\leftarrow$  faux

$T_2$  non trié



$T_1$

1 15 18 100

$T_2$

5 6 7 4 8



$\downarrow$   
    si ( $\text{trie} = \text{vrai}$ ) alors  
        1 recrue (" le tableau est trié ");  
        sinon  
            1 recrue (" le tableau n'est pas  
            trié ");  
        fin si  
    fin

# 1 Exercice 1

Ecrire les algorithmes permettant :

1. Le calcul du nombre d'occurrences d'un élément donné dans un tableau.
2. Le calcul de la moyenne et du minimum des éléments d'un tableau.
3. De tester si un tableau est trié. *par ordre croissant*
4. Le calcul du produit scalaire de deux vecteurs réels  $u$  et  $v$  de dimension  $n$ :  $u \cdot v = \sum_{i=1}^{i=n} u_i * v_i$

Traduire les algorithmes précédents en langage C.

Algorithm : R<sub>et</sub>Trié

$T[0 \dots 100]$ ;  $i$ : entier;  $\text{trie}$ : booléen;  $\text{taille}$ : entier

debut

$\text{taille} \leftarrow 100;$

pour  $i \leftarrow 0$  à  $\text{taille} - 2$

si  $T[i] > T[i + 1]$

$\text{trie} \leftarrow \text{faux};$

fin si fin pour;

$i \leftarrow i + 1;$

fin Pour

suite

$\text{trie} \leftarrow \text{vrai}$

$i \leftarrow 0$

$i \leftarrow 2$

$\text{vrai}$

$i \leftarrow 1$

$i \leftarrow 2$

$\text{vrai}$

$i \leftarrow 3$

$\text{vrai}$

$i \leftarrow 4$

$\text{vrai}$

$i \leftarrow 5$

$\text{vrai}$

$i \leftarrow 6$

$\text{vrai}$

$i \leftarrow 7$

$i \leftarrow 8$

$i \leftarrow 9$

$i \leftarrow 10$

$i \leftarrow 11$

$i \leftarrow 12$

$i \leftarrow 13$

$i \leftarrow 14$

$i \leftarrow 15$

$i \leftarrow 16$

$i \leftarrow 17$

$i \leftarrow 18$

$i \leftarrow 19$

$i \leftarrow 20$

$i \leftarrow 21$

$i \leftarrow 22$

$i \leftarrow 23$

$i \leftarrow 24$

$i \leftarrow 25$

$i \leftarrow 26$

$i \leftarrow 27$

$i \leftarrow 28$

$i \leftarrow 29$

$i \leftarrow 30$

$i \leftarrow 31$

$i \leftarrow 32$

$i \leftarrow 33$

$i \leftarrow 34$

$i \leftarrow 35$

$i \leftarrow 36$

$i \leftarrow 37$

$i \leftarrow 38$

$i \leftarrow 39$

$i \leftarrow 40$

$i \leftarrow 41$

$i \leftarrow 42$

$i \leftarrow 43$

$i \leftarrow 44$

$i \leftarrow 45$

$i \leftarrow 46$

$i \leftarrow 47$

$i \leftarrow 48$

$i \leftarrow 49$

$i \leftarrow 50$

$i \leftarrow 51$

$i \leftarrow 52$

$i \leftarrow 53$

$i \leftarrow 54$

$i \leftarrow 55$

$i \leftarrow 56$

$i \leftarrow 57$

$i \leftarrow 58$

$i \leftarrow 59$

$i \leftarrow 60$

$i \leftarrow 61$

$i \leftarrow 62$

$i \leftarrow 63$

$i \leftarrow 64$

$i \leftarrow 65$

$i \leftarrow 66$

$i \leftarrow 67$

$i \leftarrow 68$

$i \leftarrow 69$

$i \leftarrow 70$

$i \leftarrow 71$

$i \leftarrow 72$

$i \leftarrow 73$

$i \leftarrow 74$

$i \leftarrow 75$

$i \leftarrow 76$

$i \leftarrow 77$

$i \leftarrow 78$

$i \leftarrow 79$

$i \leftarrow 80$

$i \leftarrow 81$

$i \leftarrow 82$

$i \leftarrow 83$

$i \leftarrow 84$

$i \leftarrow 85$

$i \leftarrow 86$

$i \leftarrow 87$

$i \leftarrow 88$

$i \leftarrow 89$

$i \leftarrow 90$

$i \leftarrow 91$

$i \leftarrow 92$

$i \leftarrow 93$

$i \leftarrow 94$

$i \leftarrow 95$

$i \leftarrow 96$

$i \leftarrow 97$

$i \leftarrow 98$

$i \leftarrow 99$

$i \leftarrow 100$

$i \leftarrow 101$

$i \leftarrow 102$

$i \leftarrow 103$

$i \leftarrow 104$

$i \leftarrow 105$

$i \leftarrow 106$

$i \leftarrow 107$

$i \leftarrow 108$

$i \leftarrow 109$

$i \leftarrow 110$

$i \leftarrow 111$

$i \leftarrow 112$

$i \leftarrow 113$

$i \leftarrow 114$

$i \leftarrow 115$

$i \leftarrow 116$

$i \leftarrow 117$

$i \leftarrow 118$

$i \leftarrow 119$

$i \leftarrow 120$

$i \leftarrow 121$

$i \leftarrow 122$

$i \leftarrow 123$

$i \leftarrow 124$

$i \leftarrow 125$

$i \leftarrow 126$

$i \leftarrow 127$

$i \leftarrow 128$

$i \leftarrow 129$

$i \leftarrow 130$

$i \leftarrow 131$

$i \leftarrow 132$

$i \leftarrow 133$

$i \leftarrow 134$

$i \leftarrow 135$

$i \leftarrow 136$

$i \leftarrow 137$

$i \leftarrow 138$

$i \leftarrow 139$

$i \leftarrow 140$

$i \leftarrow 141$

$i \leftarrow 142$

$i \leftarrow 143$

$i \leftarrow 144$

$i \leftarrow 145$

$i \leftarrow 146$

$i \leftarrow 147$

$i \leftarrow 148$

$i \leftarrow 149$

$i \leftarrow 150$

$i \leftarrow 151$

$i \leftarrow 152$

$i \leftarrow 153$

$i \leftarrow 154$

$i \leftarrow 155$

$i \leftarrow 156$

$i \leftarrow 157$

$i \leftarrow 158$

$i \leftarrow 159$

$i \leftarrow 160$

$i \leftarrow 161$

$i \leftarrow 162$

$i \leftarrow 163$

$i \leftarrow 164$

$i \leftarrow 165$

$i \leftarrow 166$

$i \leftarrow 167$

$i \leftarrow 168$

$i \leftarrow 169$

$i \leftarrow 170$

$i \leftarrow 171$

$i \leftarrow 172$

$i \leftarrow 173$

$i \leftarrow 174$

$i \leftarrow 175$

$i \leftarrow 176$

$i \leftarrow 177$

$i \leftarrow 178$

$i \leftarrow 179$

$i \leftarrow 180$

$i \leftarrow 181$

$i \leftarrow 182$

$i \leftarrow 183$

$i \leftarrow 184$

$i \leftarrow 185$

$i \leftarrow 186$

$i \leftarrow 187$

$i \leftarrow 188$

$i \leftarrow 189$

$i \leftarrow 190$

$i \leftarrow 191$

$i \leftarrow 192$

$i \leftarrow 193$

$i \leftarrow 194$

$i \leftarrow 195$

$i \leftarrow 196$

$i \leftarrow 197$

$i \leftarrow 198$

$i \leftarrow 199$

$i \leftarrow 200$

# 1 Exercice 1

Ecrire les algorithmes permettant :

1. Le calcul du nombre d'occurrences d'un élément donné dans un tableau.
2. Le calcul de la moyenne et du minimum des éléments d'un tableau.
3. De tester si un tableau est trié. *par ordre croissant*
4. Le calcul du produit scalaire de deux vecteurs réels  $u$  et  $v$  de dimension  $n$ :  $u \cdot v = \sum_{i=1}^{i=n} u_i * v_i$

Traduire les algorithmes précédents en langage C.

Algorithme : RstTrié

$T[0..100]$ ;  $i$ : entier;  $\text{trie}$ : booléen;

debut

$\text{taille} \leftarrow 100$ ;

$\text{trie} \leftarrow \text{vrai}; i \leftarrow 0$ ;

tant que ( $i \leq \text{taille}-2$ ) et ( $\text{trie} = \text{vrai}$ ) faire  $\text{vrai}$

    si ( $T[i] > T[i+1]$ ) alors

$\text{trie} \leftarrow \text{faux}$ ;

    fin si

$i \leftarrow i+1$ ;

fin Tant que

*suite*

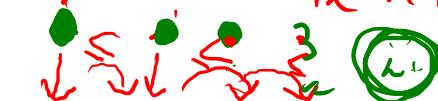
taille

$i$

$\text{trie} \leftarrow \text{vrai}$   
 $\text{trie} \leftarrow \text{faux}$

$T_1$  trié

$T_2$  non trié

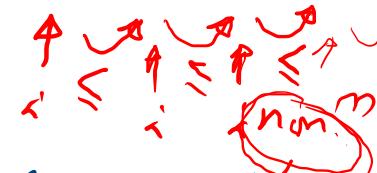


$T_1$

1 15 18 100

$T_2$

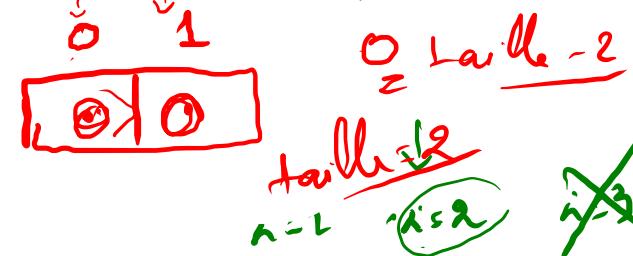
5 6 7 4 8



7

[1][30]

31



$\text{taille} \leftarrow$   
 $i = 2$

$\text{vrai} \quad \text{vrai}$

1 1 1

$i = 1 \quad i = 2 \quad i = 3$

↓

Si ( $\text{trie} = \text{vrai}$ ) alors

    recrue (" le tableau est trié");

    sinon

        recrue (" le tableau n'est pas trié");

    fin si

Fin

# 1 Exercice 1

Ecrire les algorithmes permettant :

1. Le calcul du nombre d'occurrences d'un élément donné dans un tableau.
2. Le calcul de la moyenne et du minimum des éléments d'un tableau.
3. De tester si un tableau est trié.
4. Le calcul du produit scalaire de deux vecteurs réels  $u$  et  $v$  de dimension  $n$  :  $u \cdot v = \sum_{i=1}^{i=n} u_i * v_i$

Traduire les algorithmes précédents en langage C.

*Algortithme : prod*

Si  $n$ : entier;  $U[0..n-1]$ : réel;  $V[0..n-1]$ : réel;  $s$ : entier;

Debut

```

    n ← 5; s ← 0;
    pour i ← 0 à n-1 faire
        s ← s + (U[i] * V[i]);
        i ← i + 1;
    fin pour
    Ecrire(s);
    fin
  
```

$$\vec{v} = (v_1, v_2, v_3) \sim v[0..2]$$

$$\vec{u} = (u_1, u_2, u_3) \sim u[0..2]$$

$$\vec{u} \cdot \vec{v} = u_1 v_1 + u_2 v_2 + u_3 v_3$$

$v[0..n-1]$  dimension  $n$   
 $u[0..n-1]$

$$\begin{array}{cccc}
 & 0 & 1 & 2 & 3 \\
 \downarrow & \downarrow & \downarrow & \downarrow \\
 u & -1 & 0 & -2 & 0 \\
 v & 1 & 1 & 0 & -1
 \end{array}$$

$$\begin{aligned}
 i=0 & \quad s = -1 \\
 i=1 & \quad s' = -1 + 0 \cdot 1 = -1 \\
 i=2 & \quad s = -1 + -2 \cdot 0 = -1 \\
 i=3 & \quad s' = -1 + 0 \cdot -1 = -1 \\
 \vec{u} \cdot \vec{v} & = \boxed{s = -1}
 \end{aligned}$$

| fin