

Programmation 1

Sujet 3 : tableaux et fonctions

Notions à acquérir

A lot

1 Opérations sur les tableaux correctes ?

Donner une trace de cet algorithme, tout en déterminant les lignes incorrectes.

Algo `exoTableaux`

Variables

```
tab1, tab2 : tableau de 3 entier
tab3       : tableau de 3 booleen
```

Debut

```
tab1[0] <- 3 ; tab1[1] <- 2 ; tab1[2] <- 1
```

```
tab2 <- tab1
```

```
si tab1 < tab2 alors afficher ("tableau 1 petit") finSi
```

```
tab3[1] <- faux
```

```
tab3[tab1[1]] <- vrai
```

```
tab3[1] <- tab1[1] == tab2[2]
```

```
pour i dans 0..2 faire
```

```
    tab3[i] <- tab1[i] == tab2[i]
```

```
finPour
```

Fin `exoTableaux`

2 Traitements sur un tableau

1. Ecrire une fonction (procédure) `afficherTabEntiers` (`t` : tableau de taille entier) retourne void qui affiche les éléments du tableau en paramètre.
2. Ecrire une fonction `saisirTabEntiers` (`taille` : entier) retourne tableau de entier qui crée un tableau de `taille` entiers, fait saisir ses éléments par l'utilisateur et le retourne.

3. Ecrire une fonction `moyenne` (`t : tableau de taille entier`) retourne reel qui retourne la moyenne des éléments du tableau en paramètre.
4. En faisant appel aux fonctions précédentes, écrire un algorithme `traiterTabEntiers` qui :
 - (a) initialise un tableau `tab` de 100 entiers en le faisant remplir par l'utilisateur ;
 - (b) affiche les valeurs de `tab` ;
 - (c) calcule et affiche la moyenne des éléments de `tab` ;
 - (d) modifie `tab` en ajoutant 1 à chaque élément pair de `tab` ;
 - (e) affiche les éléments du tableau `tab` modifié.

3 Fréquence des chiffres dans un nombre

1. Ecrire une fonction `frequenceChiffres` qui prend un nombre entier n en paramètre et renvoie un tableau de 10 entiers. Chaque case de ce tableau indicé de 0 à 9 contient la fréquence (le nombre d'occurrences) du chiffre en indice dans le nombre n . Cela permet de savoir combien de 0 contient le nombre, combien de 1, etc.
Par exemple, si $n = 15121$, alors la fonction renvoie le tableau `[0,3,1,0,0,1,0,0,0,0]`.
Indication : diviser n par 10 itérativement.
2. Ecrire une fonction `chiffresTousDifferentes` qui renvoie `vrai` si les chiffres (de la représentation en base 10) d'un entier n en paramètre sont distincts 2 à 2 (tous différents), `faux` sinon. Cette fonction fera appel à la fonction précédente.
Par exemple, si $n = 15121$, alors la fonction renvoie `faux` car le chiffre 1 apparaît plusieurs fois.
3. Ecrire une fonction `aChiffresDistinctsBis` qui a exactement les mêmes spécifications que `aChiffresDistincts`, mais qui est plus efficace en temps et ne fait pas appel à `frequenceChiffres`.

4 Palindrome

Ecrire une fonction booléenne `estPalindrome` qui détermine si les caractères contenus dans un tableau de caractères constituent un palindrome.

Exemples : ELLE et KAYAK sont des palindromes. RESTER n'est pas un palindrome.

On écrira une deuxième fonction `testPalin` qui appelle `estPalindrome` pour la tester. En Java, on pourra utiliser `Ut.saisirCharArray()` qui demande de saisir une chaîne de caractères sur le terminal et retourne un tableau de caractères.

5 Pyramide simple

On souhaite afficher à l'écran une pyramide de hauteur donnée h constituée de lignes répétant un caractère donné c . Par exemple, si $h = 5$ et $c = '*'$, on obtient la pyramide suivante :

```

*
***
*****
*****
*****
*****

```

1. Ecrire d'abord une fonction `repeteCarac (nb : entier, car : caractère) retourne void` qui permet d'afficher `nb` caractères `car` à partir de la position courante du curseur sur le terminal.
2. En utilisant la fonction précédente, écrire une fonction :
`pyramideSimple (h : entier, c : caractère) retourne void`
qui permet d'afficher à l'écran une pyramide de hauteur `h` constituée de lignes répétant le caractère `c`.

6 Pyramide élaborée

Affichons maintenant une pyramide un peu plus élaborée que la précédente.

1. Ecrire une fonction `afficheNombresCroissants (nb1, nb2 : entier) retourne void` qui permet d'afficher à l'écran sur une même ligne les chiffres représentant les unités des nombres allant de `nb1` à `nb2` en ordre croissant si $nb1 \leq nb2$, et ne fait rien sinon.
Par exemple, si `nb1 = 88` et `nb2 = 91`, la fonction affiche 8 9 0 1.
2. Ecrire une procédure `afficheNombresDecroissants (nb1, nb2 : entier) retourne void` qui permet d'afficher à l'écran sur une même ligne les chiffres représentant les unités des nombres allant de `nb1` à `nb2` en ordre décroissant si $nb1 \leq nb2$, et ne fait rien sinon.
Par exemple, si `nb1 = 88` et `nb2 = 91`, la fonction affiche 1 0 9 8.
3. En utilisant les fonctions précédentes (voire celles de l'exercice précédent), écrire une fonction `pyramide (n : entier) retourne void` qui permet de représenter à l'écran la pyramide suivante si $n = 10$:

```

      1
    2 3 2
  3 4 5 4 3
4 5 6 7 6 5 4
5 6 7 8 9 8 7 6 5
6 7 8 9 0 1 0 9 8 7 6
7 8 9 0 1 2 3 2 1 0 9 8 7
8 9 0 1 2 3 4 5 4 3 2 1 0 9 8
9 0 1 2 3 4 5 6 7 6 5 4 3 2 1 0 9
0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0

```

7 Maximum de trois nombres

1. Ecrire la fonction `max2 (a, b : entier) retourne entier` qui calcule et retourne la plus grande des deux valeurs `a` et `b`.
2. En utilisant la fonction précédente, écrire la fonction `max3 (a,b,c : entier) retourne entier` qui calcule et retourne la plus grande des trois valeurs `a`, `b` et `c`.

8 Nombre de chiffres composant un nombre

1. Ecrire la fonction `nbChiffres (n : entier) retourne entier`, qui calcule et retourne le nombre de chiffres que comporte l'écriture, en base 10, de l'entier naturel n .

Indication : diviser n par 10 itérativement.

2. Ecrire la fonction `nbChiffresDuCarre (n : entier) retourne entier`, qui calcule et retourne le nombre de chiffres que comporte l'écriture du nombre n^2 .

9 Nombre de triangles rectangles avec côtés entiers

Ecrire une fonction qui calcule le nombre de triangles rectangles dont les côtés sont des nombres entiers et dont le périmètre est inférieur à un nombre n passé en paramètre.

10 Nombres syracusiens

Un nombre n est syracusien si, en répétant l'opération suivante, on obtient l'entier 1 au bout d'un nombre fini d'étapes :

- si n est pair, il est remplacé par sa moitié ;
- si n est impair, il est remplacé par $3n + 1$.

1. Parmi les entiers de 1 à 10, lesquels sont syracusiens ?
2. Ecrire, si c'est possible, une fonction qui, étant donné un entier strictement positif n , détermine s'il est syracusien.
3. Ecrire une fonction qui, étant donnés deux entiers strictement positifs n et `nbMaxOp`, retourne `vrai` si on obtient 1 à partir de n en répétant au plus `nbMaxOp` fois l'opération ci-dessus, et `faux` sinon.

Que peut-on dire de n si le résultat retourné est vrai ? Que peut-on dire de n si le résultat retourné est faux ?

11 Calcul approché de e

Ecrire une fonction qui calcule une valeur approchée du nombre e en utilisant un *développement en fraction continue*. Cette fonction admet un nombre strictement positif n en paramètre et retourne la fraction obtenue en remplaçant le n^{eme} dénominateur " $2 + \dots$ " par " 2 ".

$$e = 1 + \frac{1}{1 - \frac{1}{2 + \frac{1}{3 - \frac{1}{2 + \frac{1}{5 - \frac{1}{2 + \dots}}}}}}$$