

## ENONCE DES EXERCICES

### Exercice 11.1

Écrivez une fonction qui renvoie la somme de cinq nombres fournis en argument.

---

### Exercice 11.2

Écrivez une fonction qui renvoie le nombre de voyelles contenues dans une chaîne de caractères passée en argument. Au passage, notez qu'une fonction a tout à fait le droit d'appeler une autre fonction.

---

### Exercice 11.3

Réécrivez la fonction Trouve, vue précédemment, à l'aide des fonctions Mid et Len (comme quoi, Trouve, à la différence de Mid et Len, n'est pas une fonction indispensable dans un langage).

## CORRIGES DES EXERCICES

### Exercice 11.1

Voilà un début en douceur...

**Fonction** Sum(a, b, c, d, e)

**Renvoyer** a + b + c + d + e

**FinFonction**

---

### Exercice 11.2

**Fonction** NbVoyelles(Mot en Caractère)

**Variables** i, nb **en Numérique**

**Pour** i ← 1 à Len(Mot)

**Si** Trouve("aeiouy", Mid(Mot, i, 1)) <> 0 **Alors**

nb ← nb + 1

**FinSi**

**i suivant**

**Renvoyer** nb

**FinFonction**

---

### Exercice 11.3

**Fonction** Trouve(a, b)

**Variable** i **en Numérique**

**Début**

i ← 1

**TantQue** i < Len(a) - Len(b) et b <> Mid(a, i, Len(b))

i ← i + 1

**FinTantQue**

**Si** b <> Mid(a, i, Len(b)) **Alors**

**Renvoyer** 0

**Sinon**

**Renvoyer** i

**FinFonction**

## **Fonction ChoixDuMot**

Quelques explications : on lit intégralement le fichier contenant la liste des mots. Au fur et à mesure, on range ces mots dans le tableau Liste, qui est redimensionné à chaque tour de boucle. Un tirage aléatoire intervient alors, qui permet de renvoyer un des mots au hasard.

**Fonction** ChoixDuMot()

**Tableau** Liste() **en** Caractère

**Variables** Nbmots, Choisi **en** Numérique

**Ouvrir** "Dico.txt" sur 1 **en** Lecture

Nbmots ← -1

**Tantque** Non EOF(1)

Nbmots ← Nbmots + 1

**Redim** Liste(Nbmots)

**LireFichier** 1, Liste(Nbmots)

**FinTantQue**

**Fermer** 1

Choisi ← Ent(Alea() \* Nbmots)

**Renvoyer** Liste(Choisi)

**FinFonction**

S

---

## **Fonction PartieFinie**

On commence par vérifier le nombre de mauvaises réponses, motif de défaite. Ensuite, on regarde si la partie est gagnée, traitement qui s'apparente à une gestion de Flag : il suffit que l'une des lettres du mot à deviner n'ait pas été trouvée pour que la partie ne soit pas gagnée. La fonction aura besoin, comme arguments, du tableau Verif, de son nombre d'éléments et du nombre actuel de mauvaises réponses.

**Fonction** PartieFinie(t() **en** Booleen, n, x **en** Numérique)

**Variables** i, issue **en** Numérique

**Si** x = 10 **Alors**

**Renvoyer** 2

**Sinon**

Issue ← 1

**Pour**  $i \leftarrow 0$  à  $n$

**Si Non**  $t(i)$  **Alors**

Issue  $\leftarrow 0$

**FinSi**

**i suivant**

**Renvoyer** Issue

**FinSi**

**FinFonction**

---

### **Procédure AffichageMot**

Une même boucle nous permet de considérer une par une les lettres du mot à trouver (variable  $m$ ), et de savoir si ces lettres ont été identifiées ou non.

**Procédure** AffichageMot( $m$  en Caractère par Valeur,  $t()$  en Booléen par Valeur)

**Variable** Aff en Caractere

**Variable**  $i$  en Numerique

Aff  $\leftarrow ""$

**Pour**  $i \leftarrow 0$  à  $\text{len}(m) - 1$

**Si Non**  $t(i)$  **Alors**

Aff  $\leftarrow$  Aff & "-"

**Sinon**

Aff  $\leftarrow$  Aff & Mid(mot,  $i + 1$ , 1)

**FinSi**

**i suivant**

**Ecrire** Aff

**FinProcédure**

**Remarque** : cette procédure aurait également pu être écrite sous la forme d'une fonction, qui aurait renvoyé vers la procédure principale la chaîne de caractères Aff. L'écriture à l'écran de cette chaîne Aff aurait alors été faite par la procédure principale. Voilà donc une situation où on peut assez indifféremment opter pour une sous-procédure ou pour une fonction.

## Procédure SaisieLettre

On vérifie que le signe entré (paramètre b) est bien une seule lettre, qui ne figure pas dans les propositions précédemment effectuées (paramètre a)

**Procédure SaisieLettre(a, b en Caractère par Référence)**

**Variable Correct en Booleen**

**Variable Alpha en Caractere**

**Début**

Correct  $\leftarrow$  Faux

Alpha  $\leftarrow$  "ABCDEFGHIJKLMNOPQRSTUVWXYZ"

**TantQue** Non Correct

**Ecrire** "Entrez la lettre proposée : "

**Lire** b

**Si** Trouve(alpha, b) = 0 Ou len(b)  $\neq$  1 **Alors**

**Ecrire** "Ce n'est pas une lettre !"

**SinonSi** Trouve(a, b)  $\neq$  0 **Alors**

**Ecrire** "Lettre déjà proposée !"

**Sinon**

Correct  $\leftarrow$  Vrai

a  $\leftarrow$  a & b

**FinSi**

**FinTantQue**

**Fin Procédure**

---

## Procédure VerifLettre