

Fonctions prédéfinies

Une fonction prédéfinie est un "microprogramme" autonome auquel il est possible de demander un traitement (un calcul en général) par passage d'un ou de plusieurs paramètres. Une fois le calcul assuré, la fonction restitue une valeur de retour (la réponse) au demandeur (séquence de code ayant fait appel à la fonction). Cette notion vous est sûrement familière car vous avez par exemple utilisé des fonctions natives sous Microsoft Excel (SOMME, RECHERCHEV...).

Nous verrons plus tard dans cette présentation de l'algorithmique (et aussi surtout sous JavaScript) comment développer ses propres fonctions.

L'objectif n'est pas ici d'énumérer la liste des fonctions courantes en algorithmique (elle peut d'ailleurs varier d'un auteur à un autre) mais plutôt de vous montrer comment elles peuvent s'utiliser. Une présentation au travers d'exemples ou d'exercices va être privilégiée.

1. Exercice n°4 : Affichage de la longueur d'un nom

Sujet

Écrire un algorithme permettant la saisie au clavier d'un nom pour afficher le nombre de caractères

Complément de cours : Une fonction **Longueur**(variable_chaine) prédéfinie sera utilisée pour déterminer le nombre de caractères du mot saisi.

Corrigés

Début

```
Co Déclarations Fco  
Car NOM  
  
Co Saisie du nom au clavier Fco  
Ecrire("Nom : ")  
NOM <- Lire  
  
Co Affichage du résultat Fco  
Ecrire(NOM, " contient " , Longueur(NOM), " caractère(s)")
```

Fin

ou encore :

Début

```
Co Déclarations Fco  
Car NOM  
Ent NB_CAR  
  
Co Saisie du nom au clavier Fco  
Ecrire("Nom : ")  
NOM <- Lire  
  
Co Affichage du résultat Fco  
NB_CAR <- Longueur(NOM)  
Ecrire(NOM, " contient " , NB_CAR, " caractère(s)")
```

Fin

2. Exercice n°5 : Détermination des initiales

Sujet

Écrire un algorithme qui permettra l'extraction des initiales d'une personne dont le prénom et le nom seront saisis au clavier (exemple CV pour Christian VIGOUROUX)

Complément de cours : une fonction **Sous_chaine**(CHAINE, POSITION_DEBUT, [POSITION_FIN]) prédéfinie sera utilisée.



Un paramètre de fonction qui n'est pas systématiquement nécessaire est noté conventionnellement entre crochets.

Dans le cas de la fonction **Sous_chaine**, si le troisième paramètre est absent, l'extraction se fera à partir du caractère en position POSITION_DEBUT jusqu'à la fin de la chaîne de caractères.

Dans de nombreux langages de programmation, ce troisième paramètre (ici POSITION_FIN) est en réalité un nombre de caractères à extraire à partir de la position POSITION_DEBUT).

Ci-après, vous trouverez des exemples concrets de mise en œuvre de la fonction **Sous_chaine** ainsi que de fonctions complémentaires (**Gauche**, **Droite**, **Concaténer**...).

```
Car NOM, PRENOM, INITIALES
NOM <- "VIGOUROUX"
Sous_chaine(NOM, 1, 3) vaut "VIG"
Gauche(NOM, 4) vaut "VIGO"
Sous_chaine(NOM, 3, 8) vaut "GOUROU"
Sous_chaine(NOM, 3) vaut "GOUROUX"
Sous_chaine(NOM, Longueur(NOM), Longueur(NOM)) vaut "X"
Sous_chaine(NOM, Longueur(NOM)) vaut "X"
Sous_chaine(NOM, Longueur(NOM)-1, Longueur(NOM)) vaut "UX"
Droite(NOM, 2) vaut "UX"
Gauche(PRENOM, 1) & Gauche(NOM, 1) vaut "CV"
PRENOM <- "Christian"
INITIALES <- Sous_chaine(PRENOM, 1, 1) & Sous_chaine(NOM, 1, 1) vaut "CV"
INITIALES <- Concaténer(Sous_chaine(NOM, 1, 1), Sous_chaine(PRENOM, 1, 1)) vaut "VC"
```



L'esperluette (&) sert à concaténer deux éléments de type caractère.

Corrigé

Début

```
Co Déclarations Fco
Car PRENOM, NOM, INITIALE_PRENOM, INITIALE_NOM

Co Saisie du prénom Fco
Ecrire("Prénom : ")
PRENOM <- Lire

Co Saisie du nom Fco
Ecrire("Nom : ")
NOM <- Lire

Co Détermination des initiales Fco
```

```

INITIALE_PRENOM <- Sous_chaine(PRENOM, 1, 1)
INITIALE_NOM <- Sous_chaine (NOM, 1, 1)

Co Affichage des initiales Fco
Ecrire(PRENOM, " ", NOM, " a pour initiales : ", INITIALE_PRENOM, INITIALE_NOM)

```

Fin

ou encore :

Début

```

Co Déclarations Fco
Car PRENOM, NOM, INITIALES

Co Saisie du prénom Fco
Ecrire("Prénom : ")
PRENOM <- Lire

Co Saisie du nom Fco
Ecrire("Nom : ")
NOM <- Lire

Co Affichage des initiales Fco
INITIALES <- Gauche(PRENOM, 1) & Gauche(NOM, 1)
Ecrire(PRENOM, " ", NOM, " a pour initiales : ", INITIALES)

```

Fin