

### Exercice 7 :

1) Ecrire un script qui cherche le maximum d'un vecteur sans utiliser la fonction `max`. Pour se faire, initialiser une variable `maxi` avec le premier élément du vecteur, puis écrire une boucle `for` qui passera en revue chaque élément du vecteur et qui testera si celui-ci est supérieur à `maxi`, s'il est supérieur, on mettra à jour `maxi` avec l'élément courant du vecteur. Ainsi une fois tout le vecteur parcouru, `maxi` devrait contenir la valeur maximum du vecteur.

2) Modifier le script pour également sauvegarder la position du maximum dans le vecteur.

## 4. Les fonctions

Tout comme pour les scripts, il est possible de créer ses propres fonctions au sein d'un M-file. Contrairement aux scripts, les fonctions peuvent accepter des données en entrée et renvoyer des données en sortie. Chaque fonction a son propre espace en mémoire et ne voit pas les données du workspace. Seules les données qu'on aura spécifiées comme arguments d'entrée seront utilisables par la fonction.

On définit la fonction **func** de la manière suivante :

```
function [vars1, ..., varsn] = func (vare1,  
..., varem)  
    séquence d'instructions
```

Où

- ☞ `vars1, ..., varsn` sont les variables de sortie de la fonction;
- ☞ `vare1, ..., varem` sont les variables d'entrée de la fonction;
- ☞ Séquence d'instructions est le corps de la fonction.

Le fichier doit impérativement commencer par le mot-clé **function**. Suit entre crochets les variables de sortie de la fonction, le symbole `=`, le nom de la fonction et enfin les variables d'entrée entre parenthèses. Si la fonction ne possède qu'une seule variable de sortie, les crochets sont inutiles. Il est impératif que la fonction ayant pour nom **func** soit enregistrée dans un fichier de nom **func.m** sans quoi cette fonction ne sera pas « visible » par matlab.

### Exemple :

```
function [mu, minv, maxv] = myMean ( tab )  
% MYMEAN(TAB) calcul de moyenne sur  
% un tableau 1D  
% Pour un tableau 'tab' à 1 dimension  
% [mu, minv, maxv] = myMean( tab )  
% retourne la moyenne 'mu', le minimum  
% 'minv' le maximum 'maxv' des  
% éléments de 'tab'.  
mu = 0;  
for i = 1:length(tab)  
mu = mu + tab(i);  
end  
mu = mu / length(tab);  
minv = min(tab);  
maxv = max(tab);
```

### Exercice 8 :

La fonction `max` renvoie le maximum du vecteur donné en entrée, mais elle peut aussi renvoyer la position du maximum dans ce vecteur.

### Exercice 9 :

Ecrivez une fonction de conversion de degrés Fahrenheit vers des degrés Celsius. (Rappel :  $C = 5/9(F - 32)$ )

### Exercice 10 :

Ecrivez une fonction prenant 2 arguments d'entrée et 2 arguments de sortie qui permettra de déterminer la taille en cm et le poids en kg d'une personne à partir de sa taille en pouces (1 pouce = 2.54 cm) et son poids en pound (1 lb = 0.453 kg).

## 5. Représentation graphique des résultats

Il existe plusieurs possibilités pour représenter un ensemble de points  $(x(i), y(i))$ . Les plus utilisées sont énumérées ci-dessous :

- plot(x, y, 's')** tracé d'une courbe ou d'un nuage de points.
- bar(x, y, 's')** tracé sous forme d'un histogramme.
- stem(x, y, 's')** diagramme en bâtons.
- stairs(x, y)** tracé en escalier des valeurs discrètes.
- fplot** représente des fonctions.
- hist** trace des histogrammes.

's' est un paramètre facultatif constitué d'une chaîne de caractères qui spécifie le type de tracé (couleur, différents tracés en pointillés, symboles pour le tracé de points). Par d'habitude, le tracé est continu. Tapez help plot pour avoir la liste des valeurs possibles pour 's'.

### 5.1 Gestion de la fenêtre graphique

**hold on** les prochains tracés se superposent aux tracés déjà effectués.

**hold off** le contenu de la fenêtre graphique active sera effacé lors du prochain tracé.

**clf** efface le contenu de la fenêtre graphique active.

**figure(n)** affiche ou rend active la fenêtre graphique numéro **n**.

**close** ferme la fenêtre graphique active.

**close all** ferme toutes les fenêtres graphiques.

**subplot(n,m,p)** partage la fenêtre graphique active en  $m \times n$  espaces graphiques et sélectionne le  $p$ -ème.

### 5.2. Axes et légendes

**axis([xmin xmax ymin ymax])** pour définir les échelles des axes.

**grid** quadrillage du graphique.

**grid off**

**title('titre')** titre pour le graphique

**xlabel('titre')** légende pour l'axe des abscisses.

**ylabel('titre')** légende pour l'axe des ordonnées.

**legend('titre1', 'titre2', ...)** légende pour chaque courbe du graphique.

**text(x,y, 'texte')** texte explicatif à la position (x,y).

**gtext('texte')** texte positionné à l'aide de la souris.

#### Exemple :

```
clear all
close all
clc
i=1;
hold on
for x=-1:0.05:3
    y=((x+1)*sin(x))/(x^2+1);
    f(i)=y;
    i=i+1;
end
x=-1:0.05:3;
z=sin(x);
grid
plot(x,f);
plot(x,z,'r');
legend('f','z')
title('la fonction f');
```

```
xlabel('x');
ylabel('y');
hold off
```

