

Quelques commandes de base sur Matlab

Gestion de la mémoire

`clear all` libère l'espace mémoire associé à toutes les variables,
`clear nom_var` libère l'espace mémoire associé à la variable `nom_var`,

Manipulation d'objets

`a=zeros(10,2) ;` déclare une matrice A de dimension 10x2 et la remplit de 0,
`x= 0 :0.01 :10 ;` génère un vecteur dont les composantes varient de 0 à 10 par pas de 0.01,
`A(5,2)=2 ;` affecte la valeur 2 à la composante (5,2) de la matrice A,
`y=x'` ; la transposée de la matrice x est stockée dans la matrice y ;
`c=a+b ;` la somme des matrices a et b est stockée dans la matrice c ;
`b=A*x ;` le produit matriciel de A par x est stocké dans la matrice b ;
`c=A.*x ;` les produits des composantes de A et de x sont stockés dans la matrice c ;
`x=2^(1/3);` stocke la valeur $2^{1/3}$ dans la variable x ;
`x=a.^4;` stocke la puissance 4^{ème} de chaque composante de la matrice a dans la matrice x ;
`figure ;` déclare une figure,
`figure(3);` rend active la figure 3,
`hold on;` autorise la superposition des graphes dans une même figure ;
`hold off;` interdit la superposition des graphes dans une même figure ;

Structures algorithmiques de base

`for i=1 :2 :16`
...
`end` réalise la suite d'opérations ... pour toutes les valeurs d'indice i entre 1 et 16 avec un pas de 2 ;

`while (condition)`

```
...  
end ;           réalise la suite d'opérations ... tant que (condition) est vrai ;  
  
if (condition)  
    ... 1 ...  
else  
    ... 2 ...  
end             réalise la suite d'opérations ...1... si (condition) est vrai, la suite  
                d'opérations ...2... dans le cas contraire,
```

Fonctions de base

```
size(x) ;      renvoie la taille de l'objet x ,  
  
sin(), exp(), abs(),  
exp(), sqrt(), log(),  
min(), max(), ...    les fonctions mathématiques classiques,  
  
plot(x,y) ;     trace le vecteur y en fonction du vecteur x,  
imagesc(A) ;    trace le contenu de la matrice A sous forme d'image,  
surf(A) ;       trace le contenu de la matrice A sous forme d'une surface 3D,  
colorbar ;      affiche l'échelle de couleurs utilisée pour le tracé,
```

Manipulation des fichiers

```
fid=fopen('nom_fichier') ;  
        Ouvre le fichier nom_fichier et stocke son adresse dans le pointeur fid,  
  
a=fscanf(fid, '%f\t%f\n') ;  
        Lit et stocke les données contenues dans le fichier dont l'adresse est  
        contenue dans le pointeur fid dans la matrice a. Les données sont lues  
        comme des réels (%f) contenues sur 2 colonnes séparées par une  
        tabulation (\t),  
  
a=load('nom_fichier') ;  
        Lit et stocke les données contenues dans le fichier 'nom_fichier'  
        dans la matrice a ;  
  
save data.txt -ascii -tabs z ;  
        Écrit dans le fichier data.txt les données contenues dans la matrice z.  
        Les données sont stockées au format libre ascii et les colonnes sont  
        séparées par une tabulation ;
```

`fclose(fid)` ; ferme le fichier dont l'adresse est contenue dans le pointeur `fid`.

Création de fonctions

Définition de la fonction `mon_abs(x)` qui renvoie un vecteur dont les composantes sont les valeurs absolues des composantes d'un vecteur `x` : stocker les commandes suivantes dans le fichier `mon_abs.m` :

```
function mon_abs=mon_abs(x)
    mon_abs = sign(x).*x ;
return ;
```

Remarques générales

- matlab fait la distinction entre minuscules et majuscules,
- pas de blanc dans les noms de variables,
- utilisez des noms de variables explicites,
- les indices des vecteurs varient de 1 à n (l'indice 0 n'existe pas),
- l'aide de Matlab est TRES bien faite. A utiliser sans modération.