

2.7 Fonctions appliquées sur matrices

On peut appliquer des fonctions sur des matrices pour un traitement terme à terme: *sqrt*, *sin*, *acos*, *tanh*, *exp*, *abs*...

Quelques fonction utiles:

- *max()*
- *min()*
- *mean()*
- *sort()*
- *sum()*

2.8 Langage MATLAB

2.8.1 Commentaires

% Devant la ligne à commentariser. Utilisez et abusez!

```
clc, clear, close all;           %Clear Command Window
%ligne commenté
```

2.8.2 Opérations de comparaison et opérateurs logiques

Pas de booléen mais la constante *FAUX* et *VRAI* est représentée par 0 et 1

- == égalité
- ~ négation
- ~= différent
- & ET
- | OU

```
%ex:
A = [4 5 6 0; 1 2 0 1; 3 2 1 1];
                                     %ignorer warning
A==2                                %#ok<EQEFF>
```

```
ans = 3x4 logical array
    0    0    0    0
    0    1    0    0
    0    1    0    0
```

2.8.3 Variables

Tout nom possibles mais sans accent ni espace. Variable *i* et *j* peuvent être utilisées pour les nombres complexes

```
z = 3 + 2*i
```

```
z =
3.0000 + 2.0000i
```

2.8.4 Structures de contrôle

Les parenthèses autour des conditions sont facultatives

- *if* cond...
- *else* cond...
- *end* cond...

Boucles:

- *for* var = déb:pas:fin
- ...
- *end*
- *while* cond
- ...
- *end*

```
%Exercice: Créer une matrice 4x4 H telle que H(i,j) = i+j si i~=j et H(i,j)
%= 0 sinon
```

```
H = zeros(4); %Preallouer variable

for i = 1:4
    for j = 1:4
        if i ~= j
            H(i, j) = i + j; %H(i, j) est déjà zero sinon
        end
    end
end
```

```
H
```

```
H = 4x4
```

0	3	4	5
3	0	5	6
4	5	0	7
5	6	7	0

2.9 Quelques fonctions utiles

- *rem(n, m)* m, n 2 entiers Reste de la division euclidienne de n par m
- *round(x)* x nombre relatif L'arrondi le plus proche
- *ceil(x)* L'arrondi par excès (entier sup.)
- *floor(x)* L'arrondi par défaut (entier inf)
- *find(u)* Renvoie la liste des indices des éléments non nuls de u

%exemple

```
x = [-1.2 0 3.1 6.2 -3.3 -2.1];
y = find(x < 0)
```

```
y = 1×3
     1     5     6
```

%Remplacer tous les 0 de A par 1

```
A = [4 5 6 0; 1 2 0 1; 3 2 1 1];
A(find(A==0)) = 1      %option 1
```

```
A = 3×4
     4     5     6     1
     1     2     1     1
     3     2     1     1
```

```
A = [4 5 6 0; 1 2 0 1; 3 2 1 1];
A(A == 0) = 1;      %option 2
A
```

```
A = 3×4
     4     5     6     1
     1     2     1     1
     3     2     1     1
```

3 Scripts et fonctions

Ce sont des fichiers avec l'extension *.m*. Les scripts sont une suite d'instructions à exécuter. Les fonctions retournent des paramètres selon les arguments d'entrée.

3.1 Script

- Ouvrir l'éditeur MATLAB (NEW SCRIPT)

- Enregistrer le fichier *.m* dans le repertoire de travail

Règles

- 1ère ligne est 1 commentaire décrivant le script brièvement
- *clear*; vide le workspace
- *close all*; fer,e les figures
- *clc*; nettoie la fenêtre de commandes