

Examen

2ème année maths

(Durée : 1 heure 30 min)

Exercice 1 :

Quels sont les résultats des commandes suivantes?

In [1]:

```
X = [8 1; 2 7] ;  
idx = X>3  
X(idx)
```

idx =

```
1 0  
0 1
```

ans =

```
8  
7
```

In [2]:

```
A = [1 2 3] ;  
B = ones(2) ;  
C = [4 3 2 1] ;  
X = [A ; B ; C]
```

error: vertical dimensions mismatch (1x3 vs 2x2)

In [3]:

```
A = [1 2 3 ; 4 5 6] ;  
X = repmat(A,3,2)
```

X =

```
1 2 3 1 2 3  
4 5 6 4 5 6  
1 2 3 1 2 3  
4 5 6 4 5 6  
1 2 3 1 2 3  
4 5 6 4 5 6
```

In [4]:

```
M = [8 4; 1 7] ;  
M = kron(M,ones(2,3))
```

M =

```
8 8 8 4 4 4  
8 8 8 4 4 4  
1 1 1 7 7 7  
1 1 1 7 7 7
```

Exercice 2 :

In []:

Soit une matrice

In []:

```
A = 8 4 2 1  
27 9 3 1  
64 16 4 1  
125 25 5 1
```

In []:

1- comment construire rapidement la matrice A

In [5]:

```
A=vander([2:5])
```

A =

8	4	2	1
27	9	3	1
64	16	4	1
125	25	5	1

2

In []:

2- Ecrire la commande Matlab permettant d'extraire les éléments de la diagonale de A

In [7]:

```
diag(A)'
```

ans =

8 9 4 1

^

In []:

3- Ecrire la commande Matlab permettant de supprimer la première ligne de A

In [8]:

```
A(1,:)=[];
```

A

A =

27	9	3	1
64	16	4	1
125	25	5	1

^

In []:

4- Ecrire la commande Matlab permettant de supprimer la première colonne de A

In [9]:

```
A=vander([2:5]);
```

```
A(:,1)=[];
```

A

A =

4	2	1
9	3	1
16	4	1
25	5	1

^

Exercice 3 :

In []:

1- Convertir le nombre décimal $x = 8.625$ en virgule flottante suivant la norme IEEE-754 (32 bits)

In []:

conversion de $x = 8.625$ en binaire
la partie entière 8 vers 1000
la partie décimale 0.625 vers 0.101
donc la somme 8.625 vers 1000.101

1- bit de signe : 0 (nombre > 0)

2- la mantisse sur 23bits : $1000.101 = 1.0001010 * 2^3$ donc m = 0001 0100 0000 0000 0000 0000

3- exposant sur 8bits biaisé à 127 : $3 + 127 = 130 = 1000010$ donc e = 1000 0010

donc x est converti vers 0 1000010 000101000000000000000000

ou bien : 0100 0001 0000 1010 0000 0000 0000 0000

^

^

^

In []:

2- Ecrire un script Matlab permettant de convertir des chiffres à virgule flottante (simple) en binaire selon la norme IEEE-754.

In []:

```
function s = hex2bin(x,n)
if nargin<2
    s = dec2bin(hex2dec(x));
else
    s = dec2bin(hex2dec(x),n);
end
end
```

In []:

```
function y=ieee32_conv(x)
x=single(x);
hex = num2hex(x);
dec = hex2dec(hex');
bin = dec2bin(dec,4);
bitstr = reshape(bin',[1 32]);
s = bitstr(1);
e = bitstr(2:9);
f = bitstr(10:32);
y = [s,e,f];
```

1,2

In [1]:

```
ieee32_conv(8.625)
```

```
ans = 01000001000010100000000000000000
```

Exercice 4 :

In []:

1- Ecrire un script Matlab permettant de tracer le graphe de la fonction suivante (n'utilisant aucune boucle : for, while,...) :

$$f(x) = \begin{cases} 2x - 1 & \text{si } x > 0 \\ 1 & \text{si } x = 0 \\ 3x + 1 & \text{si } x < 0 \end{cases}$$

In []:

```
f=@(x)(x>0).*(2*x-1)+(x==0)*1+(x<0).*(3*x+1);
fplot(f, [-1 1])
```

1,2

In []:

2- Ecrire un script Matlab permettant de calculer symboliquement l'intégrale suivante :

$$\int_0^{\infty} \frac{dx}{x^4+1}$$

In [13]:

```
pkg load symbolic
syms x ;
int(1/(x^4+1),x,0,inf)
```

1,5

Symbolic pkg v3.0.0: Python communication link active, SymPy v1.11.1.
ans = (sym)

$$\frac{\sqrt{2} \pi}{4}$$

In []:

3- Ecrire un script Matlab permettant de résoudre symboliquement l'équation :

$$x^6 - x^3 + x^2 - 2x + 1 = 0$$

1,5

In [14]:

```
syms x ;
solve(x^6 - x^3 + x^2 - 2*x + 1)
```

```
ans = (sym) 1
```

In []: