TP N° 5-Fonctions et polynômes

Exercice 1

1) Définir les fonctions suivantes en matlab avec trois méthodes :

$f_1(x) = 2x^2+1$
$f_2(x) = x+2e^{-x}-3$
$f_3(x) = 1-xe^{-x}$
$f_4(x) = (1/40)x^5-(1/16)x^4-(1/3)x^3+x^2$

2) Evaluer les fonctions f_1, f_2, f_3, f_4, *sin* et *cos* sur des exemples.

3) Tracer la courbe de chacune des fonctions f_1, f_2, f_3, f_4, *sin* et *cos* entre deux valeurs. (Il ne faut pas utiliser la fonction *plot* vue dans le chapitre 6 du cours).

4) Trouver des zéros des fonctions f_1, f_2, f_3, f_4, *sin* et *cos* auteur des valeurs et dans des intervalles donnés.

5) Trouver des minimums des fonctions f_1, f_2, f_3, f_4, *sin* et *cos* dans des intervalles donnés.

6) Est-ce que la fonction matlab utilisée dans la question précédente retourne toujours le minimum global ?

Exercice 2

1) Définir en matlab les polynômes suivants:

$P_1(x) = x^3+3x^2-6x-5$
$P_2(x) = x^2-3x+2$
$P_3(x) = -x^4 - x$
$P_4(x) = (1+i)x^2+(2-3i)x+3$

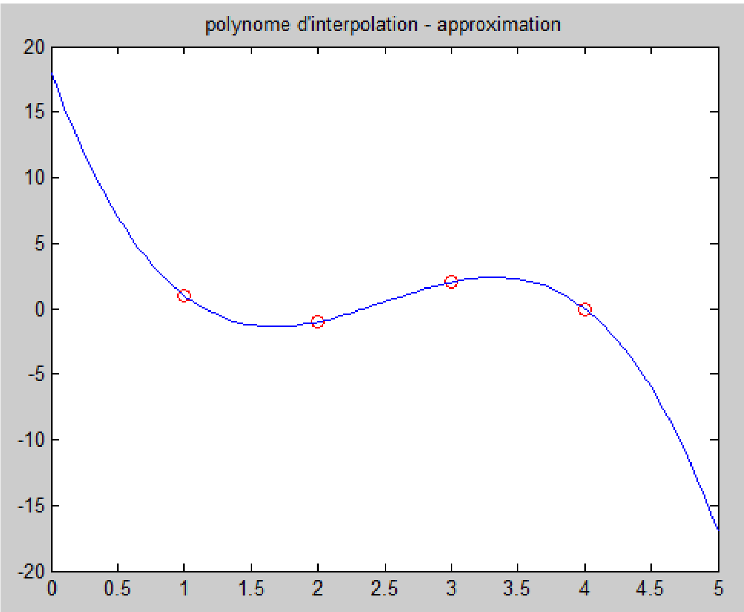
2) Evaluer ces polynômes en un ensemble de valeurs.

3) Tracer les courbes de ces polynômes.

4) Calculer les racines de ces polynômes puis montrer comment les retrouver des polynômes à partir des racines.

5) Monter avec des exemples comment calculer la somme, le produit et la division des polynômes, ainsi que la dérivée et l'intégral d'un polynôme.

6) Calculer le polynôme de degré 3 passant par les 4 points suivant s: (1, 1), (2,-1), (3,2), (4,0), puis tracer la courbe ci-dessous.



TP N° 7-Solution

Exercice 1 :

1)

Méthode 1: fonctions inlines

```
f1 = inline('2.*x.^2 + 1', 'x')
f2 = inline('x+2.*exp(-x)-3', 'x')
f3 = inline('1-x.*exp(-x)', 'x')
f4 = inline('1./40.*x.^5-1./16.*x.^4-1./3.*x.^3+x.^2', 'x')
```

Méthode 2: fonctions anonymes

```
f1 = @(x) 2.*x.^2 + 1
f2 = @(x) x+2.*exp(-x)-3
f3 = @(x) 1-x.*exp(-x)
f4 = @(x) 1./40.*x.^5-1./16.*x.^4-1./3.*x.^3+x.^2
```

Méthode 3: fonctions-M

Chaque fonction est définie dans un fichier-M.

```
function res = f1(x)
  res = 2.*x.^2 + 1;
return
function res = f2(x)
  res = x+2.*exp(-x)-3;
return
function res = f3(x)
  res = 1-x.*exp(-x);
return
function res = f4(x)
  res = 1./40.*x.^5-1./16.*x.^4-1./3.*x.^3+x.^2
return
```

2)

a) Si la fonction est définie en utilisant la $1^{\text{ière}}$ ou la $2^{\text{ième}}$ méthode :

```
feval(f1, 0), feval(f1, 1), ou bien
f1(0), f1(1)
```

b) Si la fonction est préfinie ou définie en utilisant la $3^{\text{ième}}$ méthode :

```
feval(@f1, 0), feval(@f1, [1 2 3]), feval(@f1, 1), feval(@sin, 0) ou bien
feval('f1', 0), feval('f1', 1), feval('sin', 0) ou bien
f1(0), f1(1), sin(1)
```

3)

Si les fonctions sont définies en utilisant la $1^{\text{ière}}$ ou la $2^{\text{ième}}$ méthode
`fplot(f1, [0 5])`
`fplot(f2, [-1 5])`
`fplot(f3, [0 10])`
`fplot(f4, [-1 4])`
	Si les fonctions sont définies en utilisant la $3^{\text{ième}}$ méthode
	`fplot(@f1, [0 5])`
	`fplot(@f2, [-1 5])`
	`fplot(@f3, [0 10])`
	`fplot(@f4, [-1 4])`
	`fplot(@sin, [-pi pi])`
	`fplot('cos', [-pi pi])`

4)

fplot(@f1, [0 5]) fzero(@f1,1) fzero(@f1,[-1 1])	fplot(@f2, [-1 5]) fzero(@f2,0.5) fzero(@f2,3) fzero(@f2,[-1 1]) fzero(@f2,[1 3]) fzero(@f2,[-1 3])%erreur	fplot(@f3, [0 10]) fplot(@f4, [-1 4]) fzero(@f4,0) fzero(@f4,[-0.5 0.5]) %erreur

```
fplot(@sin, [0 2*pi])
fzero(@sin,0)
fzero(@sin,3)
fzero(@sin,[3 4])
```

Exercice 2:

1)

```
P1 = [1 3 -6 -5] ;
%P1 = [1 3 -6 -5]' ;
P2 = [1 -3 2];
P3 = [-1 0 0 -1 0] ;
P4 = [1+i 2-3i 3];
```

2)

```
polyval(P1,2)
polyval(P1,1:5)
polyval(P1,[-1 0 1])
polyval(P1,[-1 0 1])
```

3)

```
x = linspace(-3, 3, 100);
y = polyval(P1,x);
plot(x,y);
grid on

x = -100:0.1:100 ;
y = polyval(P2,x);
plot(x,y);
grid on

x = -2:0.01:2 ;
y = polyval(P3,x);
plot(x,y);
grid on
```

4)

```
r = roots(P1)
poly(r)
```

6)

```
>> x1 = [1 2 3 4];     % Définir les 4
>> y1 = [1 -1 2 0];    % points
>> plot(x1,y1,'ro');  % Afficher les 4 points
>> %Calculer le polynôme d'interpolation
>> p = polyfit(x1, y1,3)
>> hold on;
>> x = linspace(0, 5, 100);
>> y = polyval(p,x);
>> plot(x, y); %Représentation graphique
>> title('polynome d''interpolation - approximation');
```