Corrigé de la Fiche de TD Récursivité

Exercice 1

a) Déroulez les procédures récursives suivantes pour k=6 :

Procédure test (↓k : entier)		Procédure essa	Procédure essai (↓k : entier)	
Début		Début		
Si (k≥0) alors	test (k-1);	Si (k≥0) alors	Écrire (k);	
	Écrire (k);		essai (k-1);	
fsi; Fin;		fsi ; Fin ;		

Déroulement :

Procédure Test:

La descente : 1er appel test (6) \rightarrow appel test (5) et empiler (6) \rightarrow appel test (4) et empiler (5) \rightarrow appel test(3) et empiler (4) \rightarrow appel test (2) et empiler (3) \rightarrow appel test (1) et empiler (2) \rightarrow appel test(0) et empiler (1) \rightarrow arrêt

Maintenant on dépile et on affiche le contenu de la pile : 1 2 3 4 5 6

Procédure essai

1er appel essai (6) → afficher (6) et appel essai (5) → afficher (5) et appel essai (4) → afficher (4) et appel essai (3) → afficher (3) et appel essai (2) → afficher (2) et appel essai (1) → afficher (1) et appel essai (0) → arrêt

Onaffiche: 654321

b) L'affichage est croisant pour test car la récursivité est non terminale. Par contre il est décroissant dans essai car la recursivité est terminale

```
c) tester (19) → tester (9), empiler (1) → tester (4); empiler (1); → tester (2); empiler (0) → tester (1); empiler (0) → tester (0); empiler (1)

Dépiler et afficher: 10011 = 19 en binaire

tester (13) → tester (6), empiler (1) → tester (3); empiler (0); → tester (1); empiler (1) → tester (0); empiler (1)

Dépiler et afficher: 1101 = 13 an binaire
```

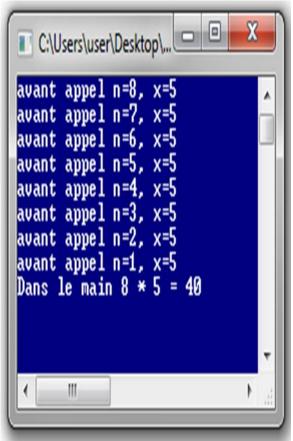
La procédure tester est non terminale

d) Déroulez la fonction récursive suivante et dites ce qu'elle fait

```
fonction produit(n:entier, x :entier):sinonentierproduit \leftarrow 0;Débutfsi, finsi (n > 0) alorsDébutecrire("avant appel", n,x);Débutproduit \leftarrow produit(n - 1, x) + x;n = 8, x = 5;ecrire ("apres appel :", n,x);}écrire (n, '*', x, '=',produit(n, x));
```

fin.

1^{er} appel Produit (8,5);



Elle fait le produit de n*x.

L'instruction ecrire ("apres appel:", n,x); dans la fonction produit n'est jamais exécutée.

Exercice 2

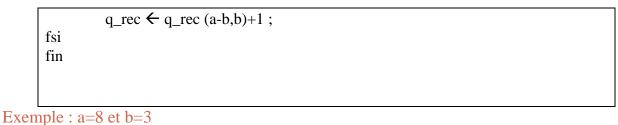
a) Écrire une fonction itérative qui renvoie le reste de la division euclidienne d'un entier a par un entier b en utilisant les soustractions successives.

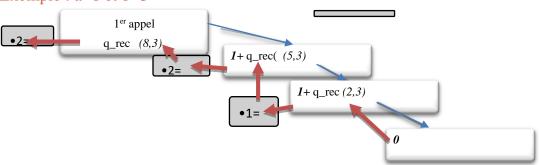
```
Fonction q_it (a,b :entier) :entier

Début
S :entier;
S \leftarrow 0;
Tque a \leq b faire
a \leftarrow a - b;
s \leftarrow s + 1;
ftque
q_it \leftarrow s;
fin
```

b) Donner la fonction récursive correspondante.

```
Fonction q_rec (a,b :entier) :entier
Début
Si a<b alors q_rec ←0;
Sinon
```





Il s'agit d'une fonction non terminale, donc il y a une descente et une remontée.

La descente nous permettra de faire les appels (flèches en bleues).

Une fois avoir atteint le point d'arrêt, on remonte pour effectuer les calculs (flèches en rouges).

Pour notre exemple, le resultat=2.

Exercice 3

```
\begin{tabular}{ll} \textbf{Fonction} & binomial(n:entier, p:entier): entier\\ Début & \\ \textbf{Si} & (p=0 \ ou \ p=n) \ alors \ Retourner \ 1 \ ;\\ \textbf{Sinon} & \\ & Retourner \ binomial(n-1, p) + binomial(n-1, p-1) \ ;\\ \textbf{FinSi} & \vdots & \\ \hline Fin & \\ \end{tabular}
```

Exercice 4

/*version 1 La forme itérative*/

Fonction premierc(n :entier) :entier

Debut

Variables i, S :entier; S←O; Pour i de 1 à n faire

Sinon Si (n mod 2 = 0) Alors

Tour_Mehdi(n-1)

Sinon

Fsi:

```
S \leftarrow S + (i*i);
Fin pour
Retourner S:
Fin
Algorithme principal
Variables A, Som: entier;
Debut
Ecrire ('Donnez la valeur de A :')
Som \leftarrow premierc(A); /* appel de la fonction*/
Ecrire ('la somme est :',Som);
Fin
/*version2 La forme récursive */
Fonction premiercrecu(n :entier) :entier
Debut
Si n=1
Alors retourner 1 /* critère d'arrêt */
Sinon si n > 1
      Alors retourner (n*n)+ premiercrecu(n-1)
Finsi
Fin
Algorithme principal
Variables A, Som: entier;
Debut
Ecrire ('Donnez la valeur de A :')
Som← premiercrecu (A); /* appel de la fonction récursive par le programme principal */
Ecrire ('la somme est :', Som);
Fin
Exercice 5
Procédure Tour Mehdi (↓n: entier)
Debut Si (n=0)
Alors Ecrire("Salim a gagné!");
Sinon Tour_Salim(n-1); Fsi; Fin.
Procédure Tour Salim ( \ln : entier)
Debut Si (n=0) Alors Ecrire("Mehdi a gagné!");
```

Tour_Mehdi(n-2);

Fsi; Fin

• Les deux procédures s'appellent mutuellement → il s'agit de procédures Récursives Croisées (indirecte)

Exercice Supplémentaire:

Fonction Truc (n : Entier) : Entier Début X : Entier ; Si (n<10) alors Truc ← n*n Sinon X ← n mod 10 ;

Truc \leftarrow X*X+ Truc (n div 10)

Fsi; Fin.

- 1-Que fait la fonction Truc?
- 2- Quelle est la nature de la récursivité.

Correction

- la fonction Truc calcule la somme des carrés des chiffres d'un entier n donné. Ex : N= 142 alors Truc= 1²+ 4²+2² .
- La nature de cette récursivité est non terminale car il ya des traitements à faire dans la phase de remontée (calcul de X*X) donc l'appel récursif ne termine pas la fonction.