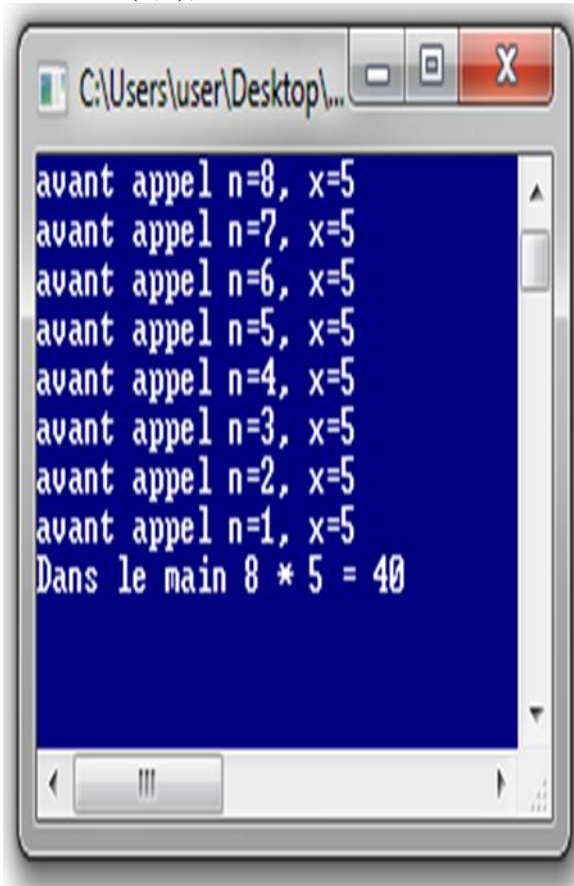




| fin.

1<sup>er</sup> appel Produit (8,5);



Elle fait le produit de  $n \cdot x$ .

L'instruction écrire ("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 ← 0 ;
Tque a ≤ b faire
a ← a - b ;
s ← s + 1 ;
ftque
q_it ← s ;
fin
```

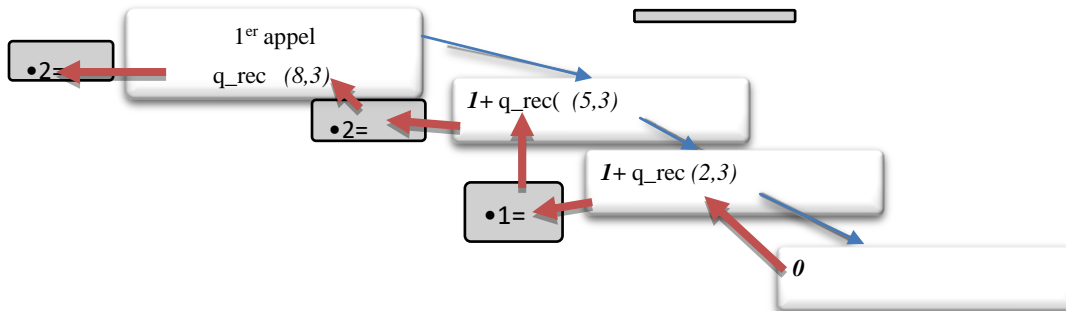
- b) Donner la fonction récursive correspondante.

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

```

q_rec ← q_rec (a-b,b)+1 ;
fsi
fin
    
```

Exemple : a=8 et b=3



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

```

Fonction binomial(n : entier, p : entier) : entier
Début
    Si (p = 0 ou p = n) alors Retourner 1 ;
    Sinon
        Retourner binomial(n - 1, p) + binomial(n - 1, p - 1) ;
    FinSi ;
Fin
    
```

### Exercice 4

*/\*version 1 La forme itérative\*/*

Fonction premierc(n :entier) :entier

Debut

Variables i, S :entier;

S←0;

Pour i de 1 à n faire

```
S←S+(i*i) ;  
Fin pour  
Retourner S ;  
Fin
```

Algorithme principal

```
Variables A, Som : entier ;  
Debut  
Ecrire ('Donnez la valeur de A :')  
Som← 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  
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 ( ↓n : entier)  
Debut Si (n=0) Alors Ecrire("Mehdi a gagné!");  
Sinon Si (n mod 2 = 0) Alors Tour_Mehdi(n-2);  
Sinon Tour_Mehdi(n-1)  
Fsi;
```

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 ← 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=  $1^2 + 4^2 + 2^2$  .
- 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.