

# Digital Campus 2019/2020 – Algorithmie de la Data

## Algorithmie simple

### Exercice 0 :

Donnez la valeur des variables A, B et C après chaque exécution.

Début

A, B, C : Entier

A ← 3 # A = 3; B = NULL; C = NULL

B ← 10 # A = 3; B = 10; C = NULL

C ← A + B # A = 3; B = 10; C = 13

B ← A \* C # A = 3 ; B = 39; C = 13

A ← C + 4 # A = 17; B = 39; C = 13

Fin

### Exercice 1 :

Écrire un algorithme qui demande un nombre à l'utilisateur puis affiche le carré de ce nombre.

Une solution possible :

Début

valeur, carre : entier

Afficher('Donnez la valeur dont vous voulez calculer le carré')

Saisir(valeur)

carre ← valeur \* valeur

Afficher('Le carré de', valeur, 'est', carre)

Fin

### Exercice 2 :

Afficher les nombres pairs compris entre 0 et 20 :

1. En utilisant une boucle Pour

Une solution possible :

Début

i : entier

Pour i allant de 0 à 20 par pas de 2 Faire

Afficher(i)

FinPour

Fin

Une autre solution possible :

```
Début
  i : Entier
  Pour i allant de 0 à 10 par pas de 1 Faire
    Afficher(i*2)
  FinPour
Fin
```

## 2. En utilisant une boucle TantQue

Une solution possible :

```
Début
  i : entier
  i ← 0
  TantQue i ≤ 20 Faire
    Afficher(i)
  i ← i + 2
  FinTantQue
Fin
```

Une autre solution possible :

```
Début
  i : entier
  i ← 0
  Tant que (i ≤ 10) Faire
    Afficher(i *2)
    i ← i + 1
  FinTantQue
Fin
```

## Exercice 3 :

Calculer la somme des n premiers nombres impairs. n est donné par l'utilisateur.

Par exemple, pour n=5 on calculera la somme :  $1 + 3 + 5 + 7 + 9 = 25$

Une solution possible :

```

Début
  n, i, somme: Entier
  somme ← 0
  Saisir(n)
  Pour i allant de 1 à 2*n par pas de 2 Faire
    somme ← somme + i
  FinPour
  Afficher(somme)
Fin

```

Une autre solution possible :

```

Début
  n, i, somme: Entier
  somme ← 0
  Saisir(n)
  i ← 0
  TantQue i <= n Faire
    somme ← somme + i*2 + 1
    i ← i + 1
  FinTantQue
  Afficher(somme)
Fin

```

#### Exercice 4 :

Afficher les n premiers entiers, n est demandé à l'utilisateur et doit être compris entre 1 et 100.

Une solution possible :

```

Debut
  n, i : entier
  Afficher('Donnez un nombre entre 1 et 100')
  Saisir(n)
  Si n < 1 OU n > 100 Alors
    Afficher(n, 'doit être compris entre 1 et 100')
  Sinon
    Pour i allant de 1 à n+1 Faire
      Afficher(i)
    FinPour
  FinSi
Fin

```

Une autre solution possible :

```

Debut
  n, i : entier
  Afficher('Donnez un nombre entre 1 et 100')
  Saisir(n)
  Si n >= 1 ET n <= 100 Alors
    i ← 0
    TantQue i <= n Faire
      Afficher(i)
      i ← i + 1
    FinTantQue
  Sinon
    Afficher(n, 'doit être compris entre 1 et 100')
  FinSi
Fin

```

### Exercice 5 :

Écrire l'algorithme simulant le fonctionnement d'une calculatrice avec les opérations (+, -, \*, /)

Une solution possible :

```

Début
  Nb1, Nb2 : reels
  Op : caractère
  Afficher('Donnez le premier nombre :')
  Saisir(Nb1)
  Afficher('Donnez le deuxième nombre :')
  Saisir(Nb2)
  Afficher('Donnez le symbole de l'opération :')
  Saisir(Op)
  Selon Op
    '+' : Afficher (Nb1 + Nb2 )
    '-' : Afficher (Nb1 - Nb2 )
    '*' : Afficher (Nb1 * Nb2)
    '/' : Si (Nb2 = 0) Alors
      Afficher ('Opération impossible')
    Sinon
      Afficher( Nb1 / Nb2)
    FinSi
  Autrement : Afficher ('Opération inexistante')
  FinSelon
Fin

```