

LIFAP1 – TD 2 : Algorithmes un peu moins simples

Objectifs : Approfondir les notions vues dans le TD précédent (boucles, conditions, structures de données, entrées / sorties, ...)

Conditionnelle à choix multiple (sélective)

SELON (sélecteur) FAIRE

Cas <liste de valeurs-1> : <suite d'action (s)-1>

[Cas <liste de valeur-2> : <suite d'action (s)-2>

.....]

[Autrement : <suite d'action (s)-n>]

FINSELON

Le sélecteur est une variable de type entier ou caractère

1. Écrire l'algorithme d'un programme permettant de simuler le fonctionnement d'une calculatrice simple (+, -, *, /). Dans cet exercice, l'utilisateur saisira les deux opérandes, l'opérateur et le programme lui affichera le résultat correspondant. Dans le cas d'une division, on vérifiera bien que le dénominateur est non nul !

Début

Nb1, Nb2 : reals

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)

Fin si

Autrement : Afficher ('Opération inexistante')

Fin Selon

Fin

2. Écrire un algorithme permettant de calculer la somme des n premières puissances de 2.
Exemple : valeur saisie : 6 → résultat : 63 (= 1 + 2 + 4 + 8 + 16 + 32).

Outil : 2^i s'écrira en algorithmique : puissance(2, i)

Début

Variables : n, somme, i : entier

Afficher('donnez la valeur de n')

Saisir(n)

somme ← 0

Pour i allant de 0 à n-1 par pas de 1 faire

 somme ← somme + puissance(2,i)

Fin Pour

Afficher (somme)

Fin

3. Écrire un algorithme permettant de calculer la somme des n premiers nombres impairs.

Exemple : valeur saisie : 6 → résultat : 36 (= 1 + 3 + 5 + 7 + 9 + 11)

Quel lien pouvez-vous établir entre la valeur obtenue et le nombre n ?

```
Début
Variables : n, somme, i : entier
Afficher('donnez la valeur de n')
Saisir(n)
somme ← 0
Pour i allant de 1 à 2*n par pas de 2 faire
    somme ← somme + i
Fin Pour
Afficher (somme)
Fin
```

Exemple pour n=5

Somme = 1+3+5+7+9 = 25 = 5²

Donc la somme des n premiers nombres impairs est égale au carré de n.

4. Écrire un algorithme permettant de lire 20 nombres entiers au clavier. Si le nombre x saisi est pair, on affiche la valeur (x / 2) sinon on affiche (3*x + 1). Attention, on ne mémorisera pas les 20 valeurs saisies.

Puisque l'on connaît le nombre de passages dans la boucle, on utilise la boucle **pour** :

```
Début
Variables : nbre, i : entier
Pour i allant de 1 à 20 par pas de 1 faire
    Afficher ('Entrez un nombre')
    Saisir(nbre)
    Si (nbre modulo 2) = 0      Alors   nbre ← nbre / 2
                               Sinon   nbre ← 3*nbre + 1
    FinSi
    Afficher(nbre)
FinPour
Fin
```

5. Afficher tous les nombres pairs compris entre 0 et 20 inclus

a. en utilisant une boucle **pour**

b. en utilisant une boucle **tant que**

a- en utilisant une boucle **pour**

```
Début
i : Entier
Pour i allant de 0 à 20 par pas de 2 faire
    Afficher(i, ' ')
Fin Pour
Fin
```

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

b- en utilisant une boucle **tant que**

```
Début
i : entier
i ← 0
Tant que (i ≤ 20)
    Afficher(i, ' ')
    i ← i + 2
Fin tant que
Fin
```

```
Début
i : entier
i ← 0
Tant que (i ≤ 10)
    Afficher(i *2, ' ')
    i ← i + 1
Fin tant que
Fin
```

6- Écrire un algorithme qui calcule la moyenne de n valeurs saisies par l'utilisateur, n étant choisi préalablement par l'utilisateur.

```
Début
Variables valeur, nbre, i, somme: entier

Afficher ('Donnez le nombre de valeurs')
Saisir(nbre)
somme ← 0
Pour i allant de 1 à nbre par pas de 1 faire
    Afficher ('Entrez une valeur')
    Saisir(valeur)
    somme ← somme + valeur
FinPour
Afficher ('La moyenne est : ', somme / nbre)
Fin
```