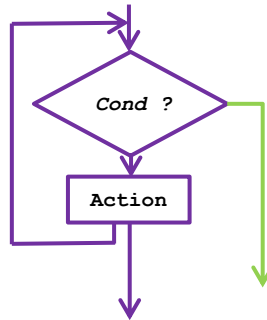




Section Date	CPI1
Enseignant Matière	M. MALDONADO ALGORITMIQUE

INSTRUCTIONS REPETITIVES



1. Boucle *Pour ... Faire*

Présentation

- Nombre de répétitions **connu**
- Utilisation d'un **compteur** (1, 2, 3, ...)

Syntaxe

- **Pour** compteur \leftarrow valeur initiale **à** valeur finale [par pas de pas] **Faire**
 Actions
 FinPour
- Compteur à déclarer
 variable de type entier
- Clause **par pas de** facultative
 par défaut : 1 (boucle incrémentale)
 pas habituel : -1 (boucle décrémentation)

Fonctionnement

- **A la première boucle**
 initialisation de la variable compteur à la valeur initiale
- **Au début de chaque boucle**
 test de la variable de boucle (si > valeur finale, fin de la boucle)

- **A la fin de chaque boucle**
incrémenter la variable compteur de la valeur du pas

Exemple

- Vous me ferez 100 lignes : *j'aime l'informatique*
- Table de multiplication de 7
Toutes les tables de multiplication
- Moyenne de 10 notes

En prog

- **C#** : 3 expressions dans le for (init, test bouclage, incrémenter)
possibilité de déclarer la variable compteur dans la boucle
...

```
for (int compteur = 1 ; compteur <= 10 ; compteur++)  
{  
    Console.WriteLine ("J'aime l'informatique");  
}
```

Rq : accolades inutiles si une seule instruction à l'intérieur

- **Python** : utilisation de l'expression range(min, max, pas)
valeur max exclue

```
for compteur in range(1,11) :  
    print ("J'aime l'informatique")
```

2. Boucle Répéter ... Jusqu'à

Présentation

- Nombre de répétitions **inconnu, mais au moins une fois**
- Condition d'arrêt en fin de boucle

Syntaxe

- **Répéter**
Actions
Jusqu'à condition d'arrêt
- Test en **fin de boucle**
au moins une boucle
- Ne pas oublier de modifier les variables de la condition dans la boucle
sinon boucle infinie

Fonctionnement

- **A la fin de chaque boucle**
si la condition d'arrêt est vraie, on sort de la boucle
sinon on repart au début de la boucle

Exemple

- *Aimez-vous l'informatique*
- Saisie du code carte bleu

En prog

- **C#** : faire... tant que (condition de bouclage : inverse de l'algo)
indentation du while

```
...
do
{
    Console.WriteLine ("Aimez-vous l'informatique ?");
    reponse = Convert.ToChar (Console.ReadLine()) ;
    if (reponse == 'N')
        Console.WriteLine ("Erreur. Recommencez");

} while (reponse != 'O')
Console.WriteLine ("Moi aussi");
```

Rq : accolades obligatoires

- **Python** : pas de boucle répéter... jusqu'à (→ boucle while)

3. Boucle *Tant que ... Faire*

Présentation

- Nombre de répétitions **inconnu, peut être 0**
- Condition de **bouclage**

Syntaxe

- **Tant que** (condition de bouclage) **Faire**
Actions
FinTq
- Variables de la condition de bouclage à initialiser **avant la boucle**
- Si la condition de bouclage est fausse dès le départ, 0 boucle

Fonctionnement

- **Au début de chaque boucle**
test de la condition de bouclage.
Si VRAI on entre dans la boucle, si FAUX on sort de la boucle

Exemple

- *Aimez-vous l'informatique ?*

En prog

- **C#** : condition de bouclage
...
reponse = ...
while (reponse != 'O')
{
 ...
}

Rq : accolades inutiles si une seule instruction à l'intérieur

- **Python** :
while (reponse != 'O') :
 ...
 ...
-