INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

Leçon sur les structures de type boucle

Université de Toulouse

Et c'est parti pour un tour !

- **Si…Sinon = petite bière !!!**
- **Nécessite d'introduire la notion de boucle**
- **Difficultés à prévoir avant d'avoir de bons réflexes.**
- **Seule vraie structure logique caractéristique de la programmation**
- **Deux sortes de boucle**
  - les boucles répétitives ou boucle **Tant Que**
  - les boucles itératives ou boucle **Pour**

Où est le problème ?

Supposons un jeu vidéo simple…. que vous devez quitter pour vous rendre en cours d'algo….

…d'où la question sous forme d'algorithme

```
Ecrire «Voulez vous sauver la partie (O/N)?»
Lire Choix
Si (choix == 'o') alors
        Sauve() ;
        Ecrire « Partie sauvée ! »
Sinon
        Ecrire « Partie non sauvée ! »
FinSi
```

Trop excité par le jeu, vous avez appuyé sur 'p' !!!

Où est le problème ?

Modification proposée :

```
Ecrire « Voulez vous sauvegarder la partie (O/N)? »
Lire Choix
Si ((Choix == 'o') OU (Choix == 'O')) alors
        Sauve() ;
        Ecrire « Partie sauvée ! »
Sinon
        Si ((Choix == 'n') OU (Choix == 'N'))
                Ecrire « Partie non sauvée, au revoir »
        Sinon
                Ecrire « Tu t'as gouré de touche ! »
        FinSi
FinSi
```

On ne répond pas mieux au problème … c'est presque même pire !

La solution…

Seule solution : être butor et redemander Tant Qu'une réponse correcte n'a pas été donnée

```
Ecrire « Voulez vous sauvegarder la partie (O/N)? »
Choix = 'R'
TantQue ((Choix ≠ 'O')ET(Choix ≠ 'o')ET(Choix ≠ 'n')ET(Choix ≠ 'N'))
        Lire Choix
        Si ((Choix == 'o') O
                Sauve() ;
                Ecrire « Pa
        Sinon
                Si ((Choix
                        Ecr                          revoir »
                Sinon
                         Ec                          che mon gars ! »
                FinSi
        FinSi
FinTantQue
```

Grande nouveauté : On revient en arrière dans le déroulement du programme

Généralité sur le TantQue

- **Ecriture générale :**

```
while (cond) :
    Intructions_de_la_boule

...
```

Les : et l'indentation délimitent les instructions de la structure

- **Remarques :**
  - **cond** (comme pour un *if*) : une variable booléenne, un test, des combinaisons logiques de plusieurs conditions….
  - Si **cond** est fausse dès le début, les instructions ne sont jamais réalisées ⇒ effet de bord possible (ex : une variable non initialisée)
  - **cond** doit être modifiée par les instructions !!!...sinon on réalise une boucle infinie

Boucle infinie

- Bête « noire » du programmeur…le programme se bloque et ne fait plus rien d'autres….
- Qq exemples simples :

```
while (True) :
    print('Oh ça tourne !!!')
```

```
a = 0
while(a < 10) :
    print(a+1)
```

```
Rep = 0
while(Rep != 3) :
    Rep = input('Que vaut 1+2 : ')
```

- Lors de l'exécution un « Ctrl + C » interrompt la boucle en cours.. Autre solution plus radicale : *tuer la console*

Exemple

- Que fait l'exemple suivant ?

```
Nombre = int(input ('Donnez un nombre :'))
Resu = 0
while (Nombre > 0) :
        Resu = Resu + 1
        Nombre = Nombre - 3

print('Resu vaut : ',Resu)
```

- Quel sera l'affichage avec :
  - Nombre = -1
  - Nombre = 10
  - Nombre = 3

Le jeu de 7 erreurs

- Où sont les 7 erreurs :

```
Val = input ('Donnez un nombre :')
while (Val < 150 or Boucle < 100)
    print("Boucle:",Boucle,"Val:",Val)
    Boucle = Boucle + 1
    Val = Val/2 + 2Boucle

    print('Val final vaut : ',Val)
```

La 7ième erreur : le nombre d'erreurs à trouver

```
 Val = int(input ('Donnez un nombre :'))
Boucle = 0;
while (Val < 150 and Boucle < 100) :
    print('Boucle:',Boucle,'Val:',Val)
    Boucle = Boucle + 1
    Val = Val/2 + 2*Boucle

print('Val final vaut : ',Val)
```

Application directe

- Ecrire un programme qui affiche les valeurs successives de la récurrence suivante tant que la valeur calculée reste inférieure à 5000 :

$$X_{n+1} = 1.5X_n + 0.25$$

- La valeur initiale X_0 sera réelle et saisie par l'utilisateur
- Modification 1 : afficher également la valeur de l'indice n
- Modification 2 : sortie de la boucle également si l'indice n devient supérieur à 10