The background of the slide is a dark, blue-toned image of a circuit board. It features intricate patterns of gold and silver traces, along with various electronic components like capacitors and integrated circuits. Overlaid on the right side of the circuit board is a grid of binary code (0s and 1s) in a light blue/green color, giving it a digital or technological feel.

# Programmation 1: Fondamentaux

Structures Itératives ou Boucles

# Actions qui se répètent

Structures Itératives

# Actions répétitives :

Affichage des 5 premières valeurs entières positives

*Début*

*Faire*

*Afficher 1*

*Afficher 2*

*Afficher 3*

*Afficher 4*

*Afficher 5*

*Fait*

*Fin*

C'est moche ☹ ! Et pas efficace pour un grand nombre d'itérations, surtout si la séquence d'instructions est importante.

# Utilisation d'une structure itérative

Affichage des 5 premières valeurs entières positives

Début

Faire *Entier i*

*Pour i allant de 1 à 5*

*Faire*

*Afficher i*

*Fait*

Fait

Fin

L'incrémentation de la variable d'itération *i* est automatique et se fait par défaut de 1 en 1. En langage C, il faut penser à incrémenter la variable au risque d'avoir une boucle infinie.

# Pour ..

La boucle **pour** est utilisée surtout quand on connaît le nombre d'itérations qui vont être effectuées :

Exemple : Affichage des 5 premières valeurs entières positives

Début

Faire

Entier  $i$

Pour  $i$  allant de 1 à 5

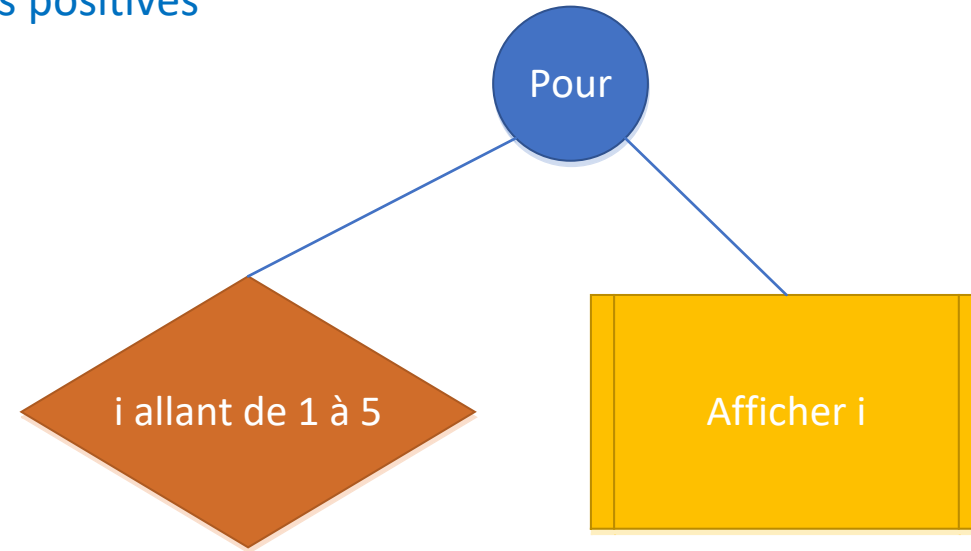
Faire

Afficher  $i$

Fait

Fait

Fin



L'incrémentation de la variable d'itération  $i$  est automatique et se fait par défaut de 1 en 1. En langage C, il faut penser à incrémenter la variable au risque d'avoir une boucle infinie.

En langage C: `for ( ; ; ) { }`

```
int i;  
for ( i=1; i < 5 ; i++) { printf("%d " , i );}
```

La valeur de la variable d'itération ne devrait pas être modifiée dans le bloc d'instructions à répéter.

En langage C: `for ( ; ; ) { }`

```
int i;  
for ( i=1; i < 5 ; i++) { printf("%d " , i );}
```

Champ1 : Action  
**exécutée une fois**  
avant d'entrer  
dans la boucle

Champ2 : **Condition d'arrêt** :  
en fait, on entre dans la  
boucle si la condition est  
vérifiée, c'est donc plutôt  
une condition pour  
poursuivre le traitement !

Champ3 : **Action exécutée  
après chaque itération** :  
permet de mettre à jour la  
variable d'itération qui  
contrôle la boucle

Bloc d'instruction : **Contient  
toutes les instructions à  
exécuter à chaque itération**

# Tant que .. faire..

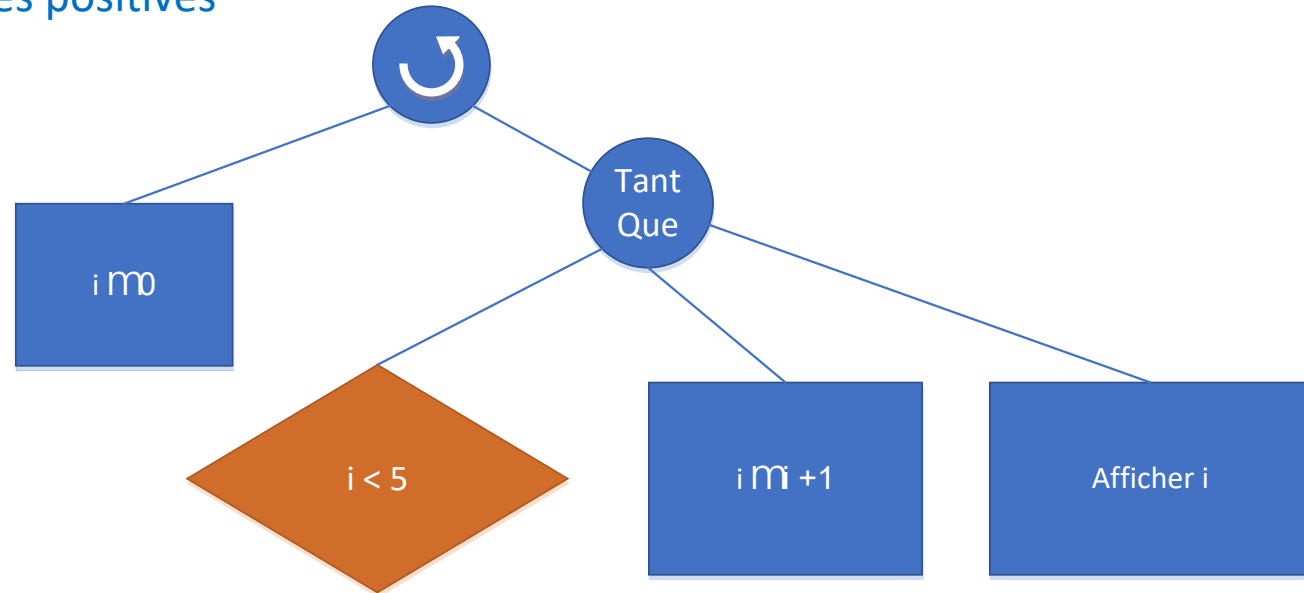
Affichage des 5 premières valeurs entières positives

Début

Faire Entier  $i \leftarrow 0$   
Tant que  $i < 5$   
Faire  
 $i \leftarrow i + 1$   
Afficher  $i$   
Fait

Fait

Fin



- L'incrémentation de la variable d'itération  $i$  n'est PAS automatique !
- Elle doit être initialisée avant l'entrée dans la boucle avec la bonne valeur
- Il faut réfléchir à l'endroit où on incrémente la variable (en début, en fin de séquence)
- La boucle doit toujours modifier la variable qui la contrôle au risque d'avoir une boucle infinie



# tant que.. faire, autre version

Affichage des 5 premières valeurs entières positives

Début

Faire

Entier  $i \leftarrow 1$

Tant que  $i \leq 5$

Faire

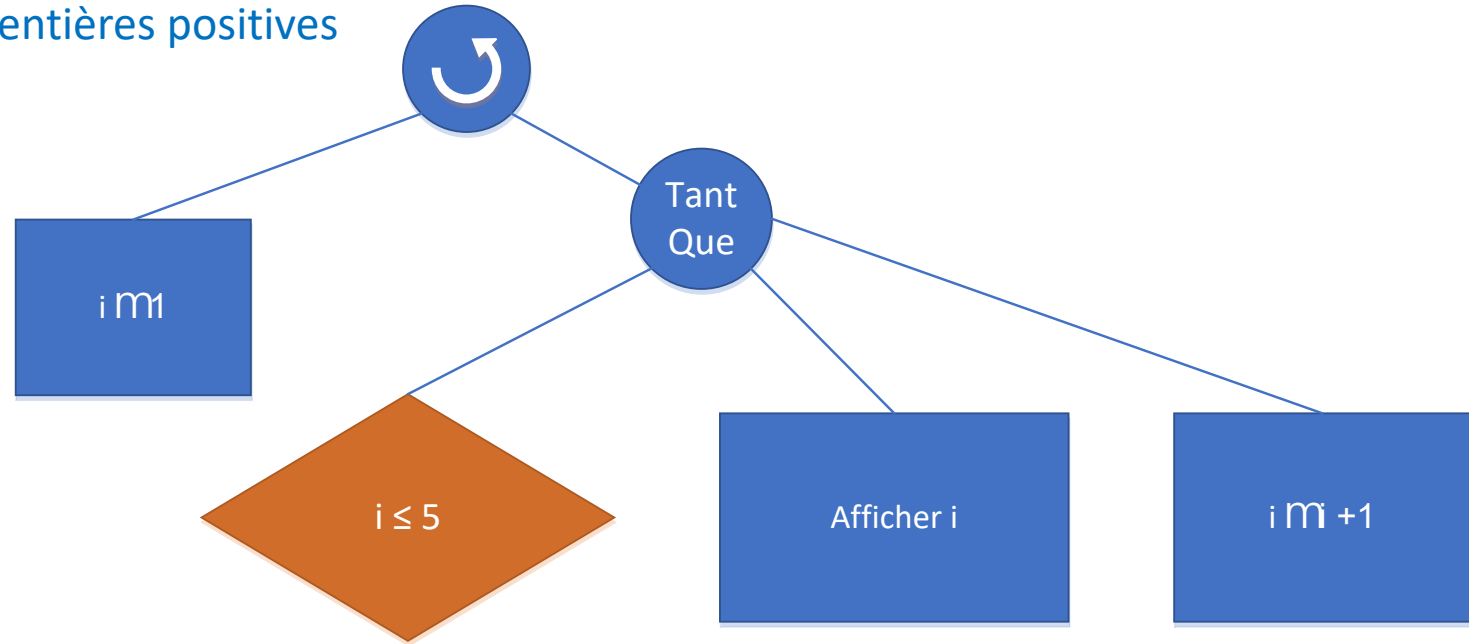
Afficher  $i$

$i \leftarrow i + 1$

Fait

Fait

Fin



- $i$  vaut 1 avant d'entrer dans la boucle
- La condition vérifie que  $i$  ne dépasse pas la valeur 5
- La valeur est affichée
- $i$  est alors incrémenté en vue de tenter une nouvelle itération. En sortie de boucle  $i$  vaut 6.

# faire .. Tant que

Affichage des 5 premières valeurs entières positives

Début

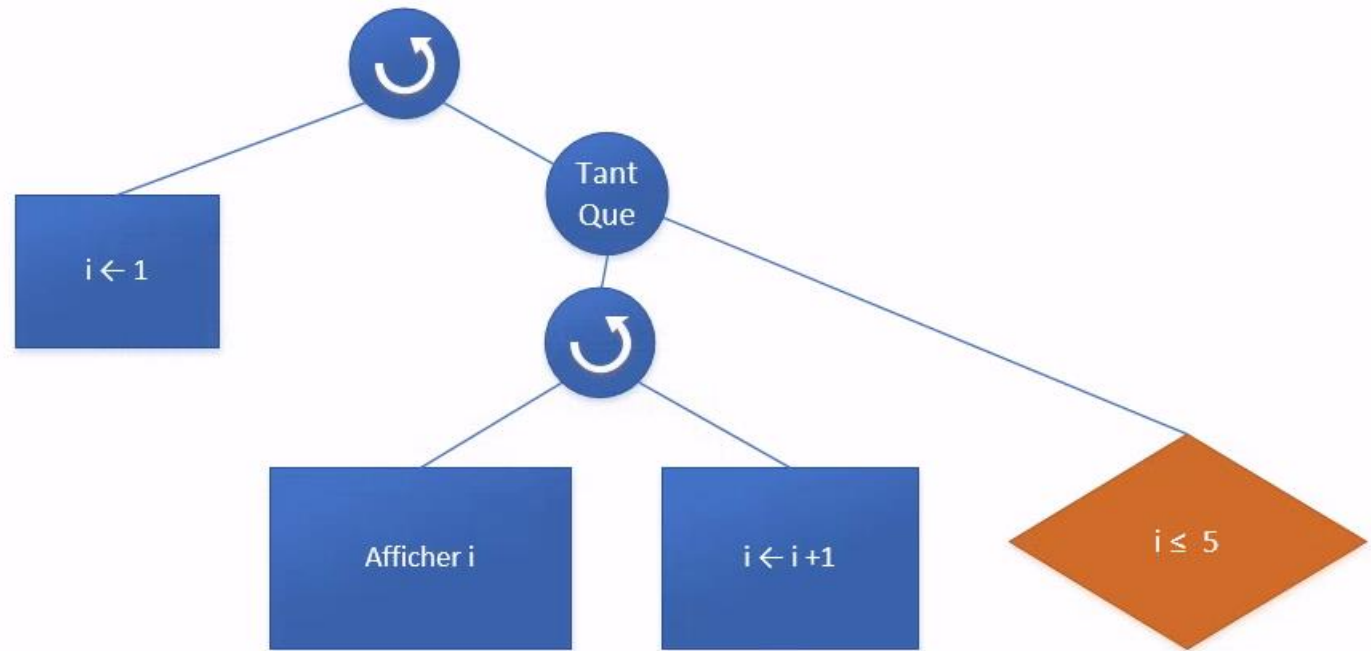
Faire Entier  $i \leftarrow 1$

Faire  
Afficher  $i$   
 $i \leftarrow i + 1$

Fait  
Tant que  $i \leq 5$

Fait

Fin



- La séquence d'instructions est au moins exécutée une fois
- La condition d'arrêt est évaluée après.

En langage C:

`while ( ) { }`

ou

`do { } while ( ) ;`

```
int i = 1;

while (i <= 5) {

    printf(" %d ", i);
    i = i + 1;
}
```

```
i = 1;

do {

    printf(" %d ", i);
    i = i + 1;
}
while (i <= 5);
```

## Exercice :

On souhaite écrire un programme qui offre la possibilité à son utilisateur de calculer la moyenne, d'un nombre quelconque de valeurs entières, jusqu'à ce qu'il entre la valeur négative -1

De quoi à ton besoin ?

- de communiquer avec l'utilisateur
  - Commencer ?
  - Première valeur ?
  - Encore une valeur ?
- d'une ou plusieurs valeurs, ou aucune..