

SCI6373 Programmation documentaire

Cours 5

Été 2025

Structure de contrôle while (tant-que)

while (tant-que) (1/6)

```
while (condition) {  
    bloc-d'énoncés-à-répéter  
} ;
```

condition
Comme pour un "if" ou le choix conditionnel ? :
condition est une expression qui retourne true ou false typiquement, **une comparaison**

while (tant-que) (2/6)

- Commence comme un if sans else
- Si condition fausse → ne fait rien
 - comme dans un "if" sans "else"
- Si condition vraie → le bloc est exécuté
 - À la fin de l'exécution du bloc, la condition est **réévaluée**
 - Si elle est encore vraie, le bloc est exécuté de nouveau

while (tant-que) (3/6)

- Le bloc est donc exécuté itérativement **tant que** la condition demeure vraie (d'où le nom)
- Il faut donc que "condition" soit une expression dont la valeur peut passer de true à false
 - Si condition toujours vraie : **boucle infinie**
 - Pour arrêter une boucle infinie : fermer l'onglet !

while (tant-que) (4/6)

- Pragmatiquement :
 - *condition* est une expression impliquant une variable (typiquement, une *comparaison*)
 - le bloc d'énoncés à répéter se termine par un énoncé qui modifie cette variable (typiquement, une *incrémentation*)

while (tant-que) (5/6)

Pattern typique :

```
v = expression;  
while (condition impliquant v) {  
    bloc-d'énoncés-à-répéter  
    énoncé qui modifie v  
};
```

Par exemple :

```
v = 0; ← Initialisation  
while (v < limite) {  
    bloc-d'énoncés-à-répéter  
    v++; ← Incrémantation  
};
```

limite est souvent la longueur d'une structure, par exemple, d'une chaîne de caractères

while (tant-que) (6/6)

Exemple :

```
chaîne = "ça va?" ;
n = 0; ← Initialisation
v = 0; ← Initialisation
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++; ← Incrémentation
};
```

Exemple 205

Un exemple pas à pas

Simulation

Environnement :

```
chaîne = "ça va?" ;
n = 0 ;
v = 0 ;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```

Simulation

Environnement :

```
chaîne = "ça va?" ;
n = 0 ;
v = 0 ;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```

Simulation

Environnement : chaîne : "ça va?"

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : chaîne : "ça va?"

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"
n : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"
n : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

v < 6



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";           0 < 6 → true
n = 0;
v = 0;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
    chaîne.charAt(0) → "ç"
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
"ç" == "a" → false
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 0

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

v < 6



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";           1 < 6 → true
n = 0;
v = 0;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
    chaîne.charAt(1) → "a"
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
"a" == "a" → true
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 0

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 1

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

v < 6



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";           2 < 6 → true
n = 0;
v = 0;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
    chaîne.charAt(2) → " "
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
" " == "a" → false
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 2

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1
v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

v < 6



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 3

```
chaîne = "ça va?";           3 < 6 → true
n = 0;
v = 0;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1
v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1
v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
          chaîne.charAt(3) → "v"
```



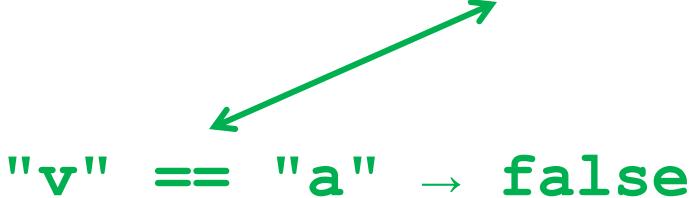
Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
"v" == "a" → false
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 3

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

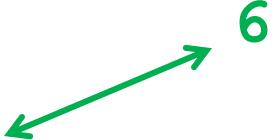
Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

v < 6



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";           4 < 6 → true
n = 0;
v = 0;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```


chaîne.charAt(4) → "a"

Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
"a" == "a" → true
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 1

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 4

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



A green arrow points from the value '6' to the character 'a' in the string "ça va?", indicating the current index of the character being checked.

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



v < 6

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";           5 < 6 → true
n = 0;
v = 0;
while (v < chaîne.length) {
    if (chaîne.charAt(v) == "a") n++;
    v++;
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
    chaîne.charAt(5) → "?"
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};  
    "?" == "a" → false
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 5

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 6

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 6

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 6

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```



Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 6

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

v < 6



Simulation

Environnement : **chaîne** : "ça va?"

n : 2

v : 6

```
chaîne = "ça va?";  
n = 0;  
v = 0;  
while (v < chaîne.length) {  
    if (chaîne.charAt(v) == "a") n++;  
    v++;  
};
```

6 < 6 → false

*L'exécution du **while** se termine, puisque la condition s'évalue à **false***

Simulation

- La boucle `while` s'est donc trouvée à compter le nombre de caractères dans `chaîne` qui sont égaux à "a"
 - Autrement dit, le nombre de "a" dans `chaîne`
- Cette boucle réalise le *PC* "balayage d'une structure", en l'occurrence, balayage d'une chaîne de caractères
 - chaque caractère est traité à tour de rôle, de la position 0 à `chaîne.length - 1`

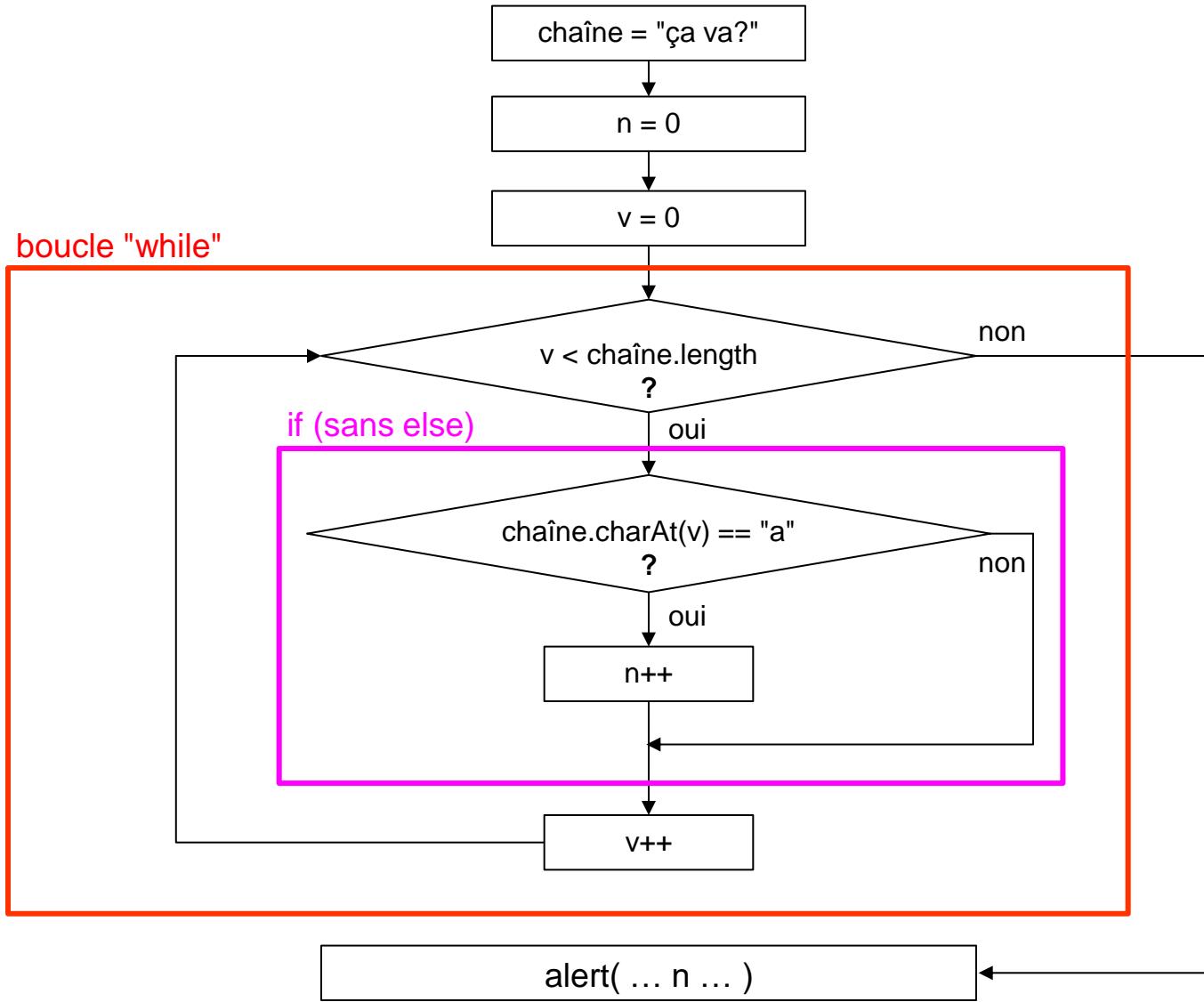
Organigrammes (1/3)

- Le déroulement d'un énoncé while s'illustre bien grâce à un outil graphique appelé "organigramme"
- Permet de suivre visuellement le flux d'exécution des énoncés dans un programme
- Applicable aussi au if et au if-else

Organigrammes (2/3)

- Les points de décision sont indiqués par un losange contenant une *condition*
 - Le flux d'exécution se "divise" en deux, selon que la *condition* est vraie ou fausse
- Voici l'organigramme pour la séquence d'énoncés analysés, incluant le while et le if (sans else) imbriqué

Organigrammes (3/3)



Bonne pratique avec **while** (1/2)

- Toujours en mettre *le minimum* dans le bloc qui est répété
- Similaire à la bonne pratique pour if, if...else
- Ici, c'est une question de ne pas répéter des opérations pour rien
 - Vaut mieux les faire une seule fois *avant* la boucle et mettre le résultat dans une variable

Bonne pratique avec **while** (2/2)

- Cette bonne pratique s'applique aussi la structure de contrôle itérative `for` (à venir sous peu)

TP 1

Votre premier "contrat"

- Pondération : 10%
 - À rendre vendredi soirée
 - Remise sur StudiUM
-
- Je vous donnerai le TP 2 en PM demain

Conseils pour TP

- *Pendant le développement d'un script qui prend son intrant avec un prompt(), remplacez le prompt par une simple spécification d'une valeur fixe*
 - Remplacer : nom = prompt ("Nom?") ;
 - par : nom = "Totoche" ;
- Ça évite d'avoir à répondre au prompt() à chaque recharge de la page HTML

Mini-quiz #1

- En classe vendredi 30 mai AM
 - Durée: \approx 40 minutes
 - Pondération : 15%
 - Sur StudiUM