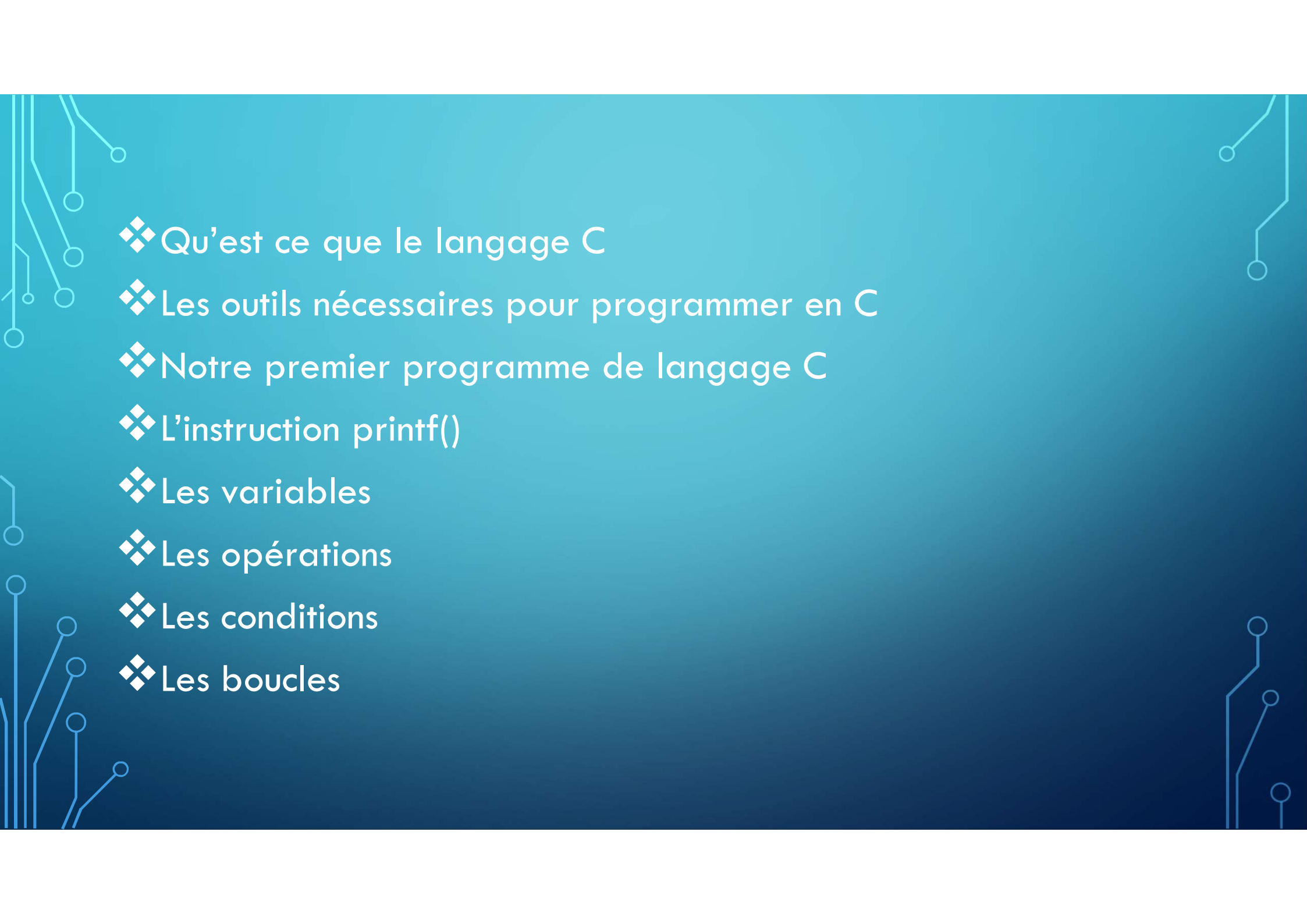


The background is a blue gradient with white circuit-like lines and nodes in the corners.

Programmer *en C*

- 
- The background of the slide is a solid blue gradient. Overlaid on this background are white, stylized circuit traces. These traces consist of thin lines that branch out and terminate in small circles, resembling a printed circuit board (PCB) layout. The traces are located primarily along the left and right edges of the slide, with some extending slightly into the central area.
- ❖ Qu'est ce que le langage C
 - ❖ Les outils nécessaires pour programmer en C
 - ❖ Notre premier programme de langage C
 - ❖ L'instruction printf()
 - ❖ Les variables
 - ❖ Les opérations
 - ❖ Les conditions
 - ❖ Les boucles

QU'EST CE QUE LE LANGAGE C

C'est une langage de programmation.

un langage de programmation permet de décrire d'une part les structures des données qui seront manipulées par l'appareil informatique, et d'autre part d'indiquer comment sont effectuées les manipulations, selon quels algorithmes. Ils servent de moyens de communication par lesquels le programmeur communique avec l'ordinateur, mais aussi avec d'autres programmeurs ; les programmes étant d'ordinaire écrits, lus, compris et modifiés par une équipe de programmeurs.

(Wikipédia)

QU'EST CE QUE LE LANGAGE C

Classement Tiobe janvier 2018

1. Java (14,22%)
2. C (11,04%)
3. C++ (5,6%)
4. Python (4,68%)
5. C# (3,75%)
6. JavaScript (3,47%)
7. Visual Basic .Net (3,26%)
8. R (2,55%)
9. PHP (2,53%)
10. Perl (2,42%)

Parmi les autres langages ayant progressé en 2017, grimpé de la 16e à la 8e place, Kotlin (qui passe de la encore Erlang (23e position versus 44e sur un an). C langages prometteurs comme Julia, Hack ou encore

LES OUTILS NÉCESSAIRES POUR DÉBUTER EN C

- ❖ Un éditeur de texte
- ❖ Un compilateur
- ❖ Un débogueur

LES OUTILS NÉCESSAIRES POUR DÉBUTER EN C

Un IDE : **I**ntegrated **D**evelopment **E**nvironment (en français « environnement de développement »)

- ❖ un éditeur de code intelligent (coloration, autocomplétion, mise en forme) ;
- ❖ un compilateur (qui va transformer le code source rédigé par le développeur en code binaire, composé de 0 et de 1 uniquement) ;
- ❖ un débogueur (fonctionnalité d'aide à la correction de bugs), etc.

NOTRE PREMIER PROGRAMME DE LANGAGE C

Structure minimal d'un programme de C

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main()
```

```
{
```

```
    printf("Hello world!\n");
```

```
    return 0;
```

```
}
```

Inclusion ou directives
de préprocesseur

la fonction principale

Les instructions

L'INSTRUCTION PRINTF

<code>\n</code>	retour à la ligne
<code>\b</code>	n'imprime pas la lettre précédente
<code>\r</code>	n'imprime pas tout ce qui est avant
<code>\t</code>	tabulation horizontale
<code>\v</code>	tabulation verticale
<code>\"</code>	"
<code>\'</code>	,
<code>\?</code>	?
<code>\!</code>	!
<code>\\</code>	\

LES VARIABLES EN C

Type	Size in bits	Minimum	Maximum
char	8	-128	127
unsigned char	8	0	255
short	8	-128	127
unsigned short	8	0	255
int	16	-32768	32767
unsigned int	16	0	65535
long	32	-2,147,483,648	2,147,483,647
unsigned long	32	0	4,294,967,295
float	32	-3.40282×10^38	3.40282×10^38
double	64	-1.79769×10^{308}	1.79769×10^{308}
pointer or void	system dependent	?	?

Table 1 - basic C types

LES VARIABLES EN C

Code format	Type de l'argument à afficher	Format d'affichage et exemples
%c	caractère	Ex : a G ? +
%hd	entier <i>short int</i> (ou caractère) signé	Base 10. Ex : -12
%hu	entier <i>short int</i> (ou caractère) non signé	Base 10. Ex : 463
%hX	pour afficher en hexadécimal (<i>short</i> ou <i>char</i>)	Hexadécimal. Ex : 9A0F
%ld	entier <i>long int</i> signé	Base 10. Ex : -1289
%lu	entier <i>long int</i> non signé	Base 10. Ex : 46399
%lX	entier <i>long int</i> en hexadécimal	Hexadécimal. Ex : B4E98A0F
%d	entier <i>int</i>	Base 10. Ex : -546
%X	pour afficher un <i>int</i> en hexadécimal	Hexadécimal. Ex : 9A0F
%lf	Réal double précision (<i>double</i>)	Virgule flottante. Ex : -3.141592
%le		Avec exposant. Ex : -1.450000e-7
%f	Réal simple précision (<i>float</i>)	Virgule flottante. Ex : -3.141592
%e		Avec exposant. Ex : -1.450000e-7
%s	Chaîne de caractères	Ex : bonjour !

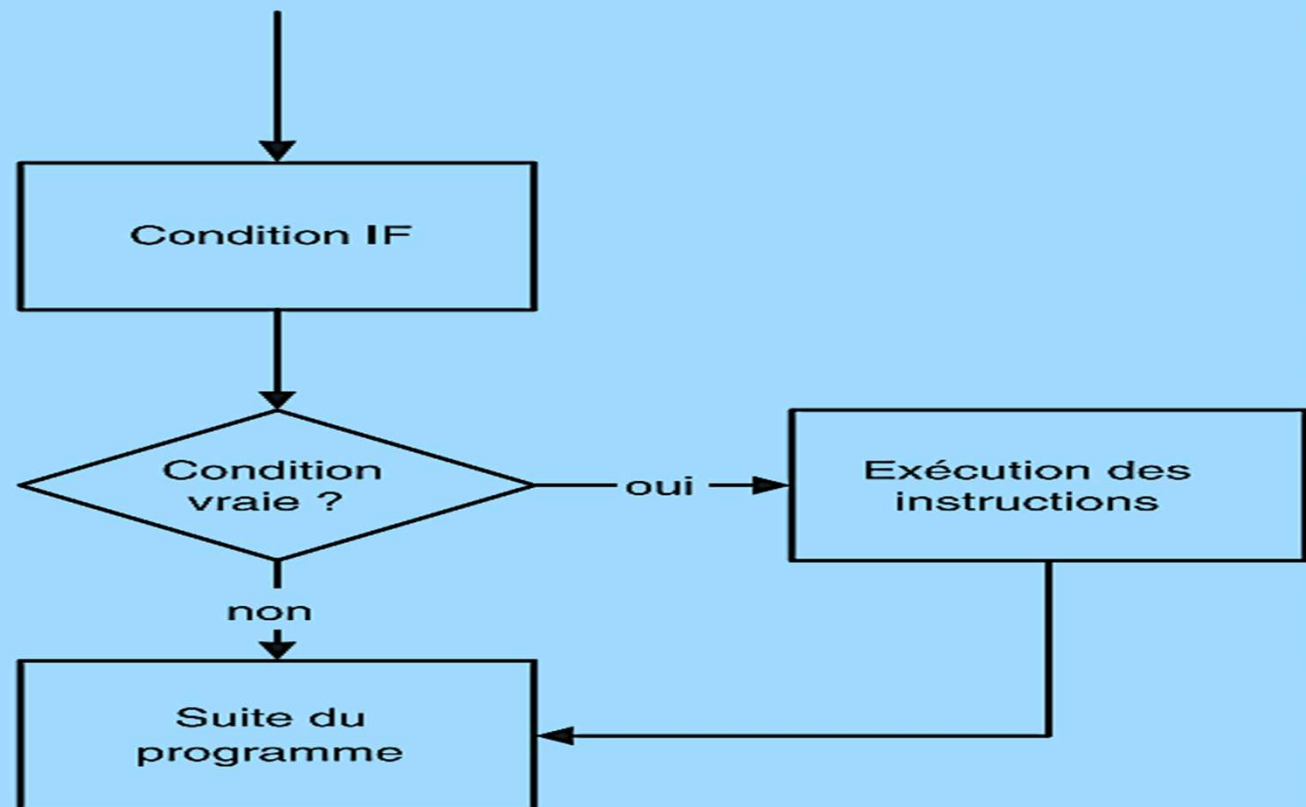
Les codes formats

LES OPÉRATEURS

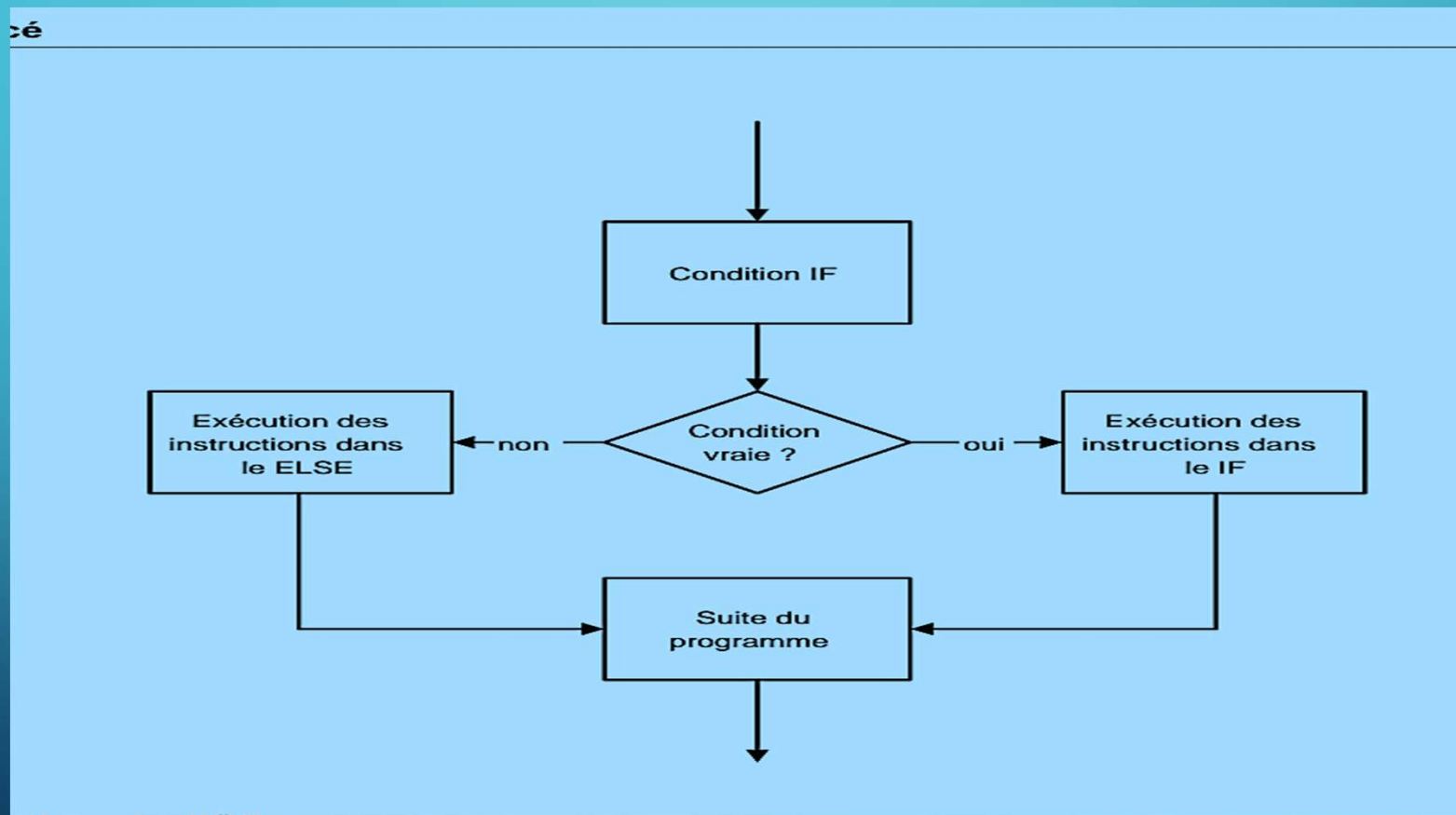
Les opérateurs

- Opérateurs **arithmétiques**
+, -, *, /, % (modulo), ++(incrément), --(décrément)
- Opérateurs d'**assignation**
= (affectation), += (ajouter à), *=, /=, -=, %=
- Opérateurs de **comparaison**
== (égalité), <, <=, >, >=, != (différent)
- Opérateurs **logiques**
&& (et), || (ou), ! (non)

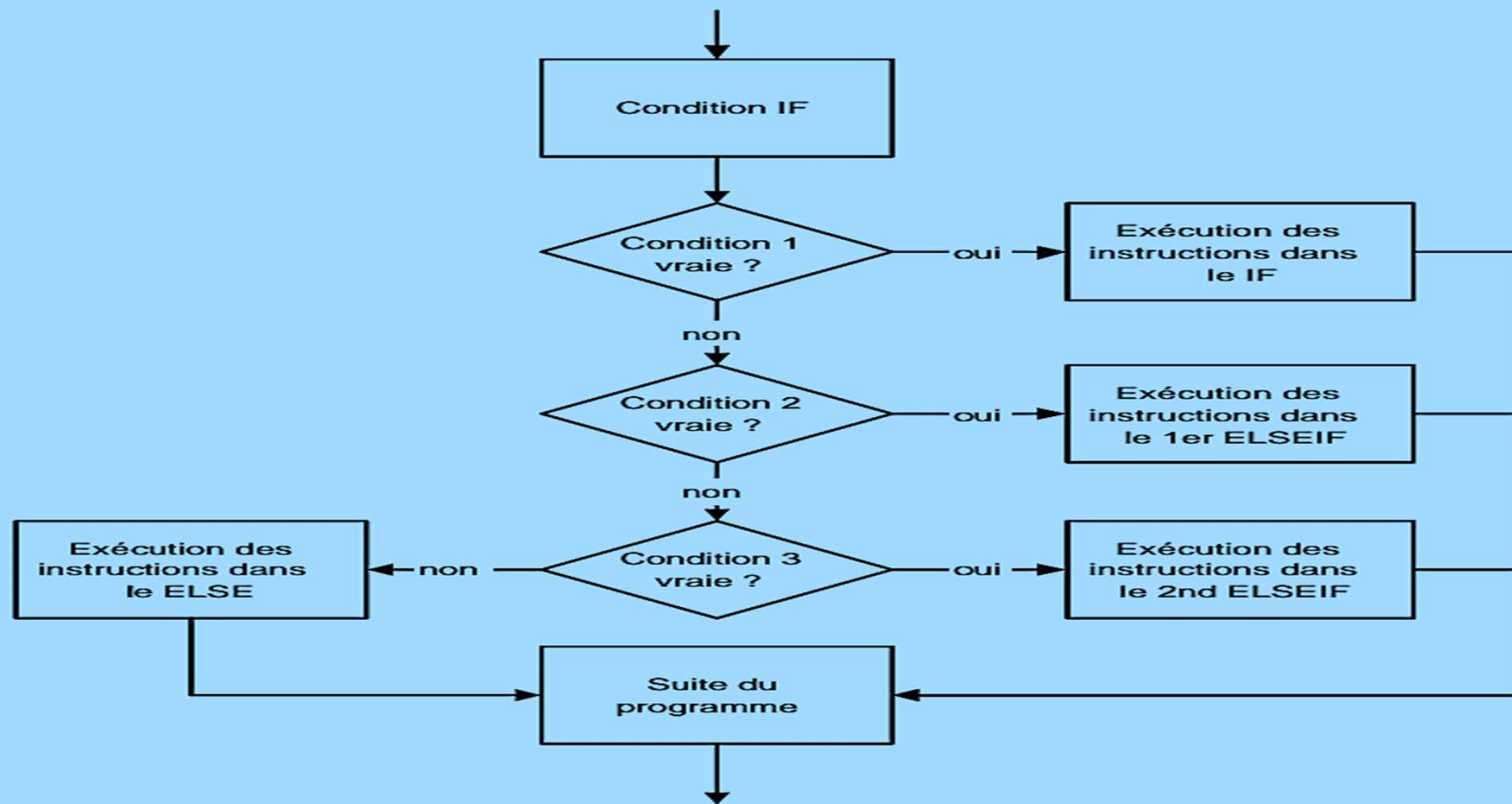
LES CONDITIONS



LES CONDITIONS



LES CONDITIONS



LES CONDITIONS

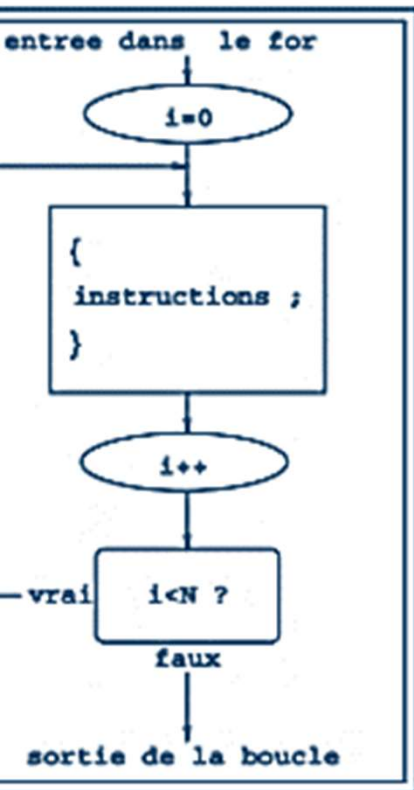
Symbole	Signification
==	Est égal à
>	Est supérieur à
<	Est inférieur à
>=	Est supérieur ou égal à
<=	Est inférieur ou égal à
!=	Est différent de

LES CONDITIONS

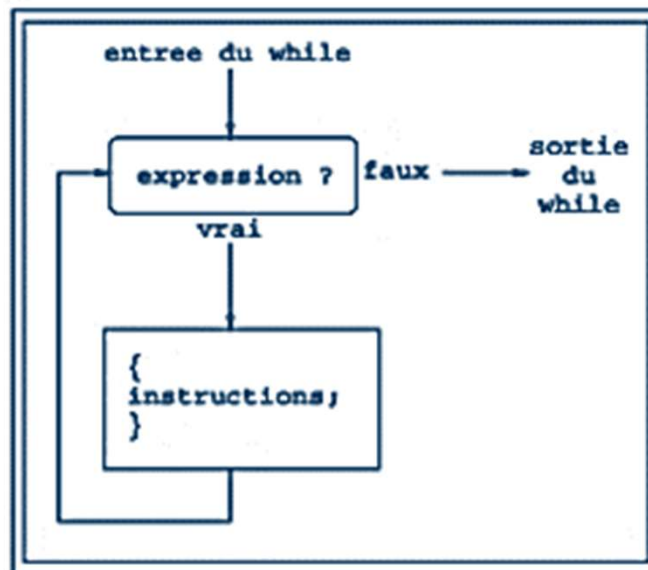
&&	ET
 	OU
!	NON

LES BOUCLES

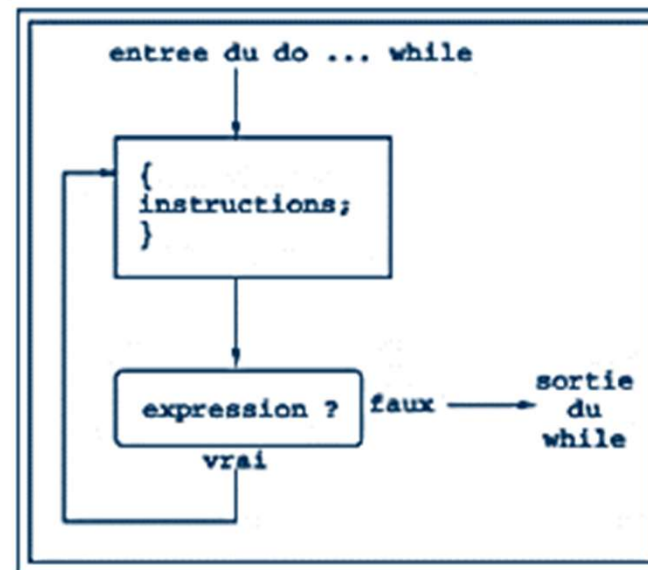
Les boucles



Organigramme
d'une boucle for



Organigramme
d'une boucle while



Organigramme
d'un do ... while

LES BOUCLES

Qu'est-ce qu'une boucle ? C'est une technique permettant de répéter les mêmes instructions plusieurs fois.

LES BOUCLES



Instructions
Instructions
Instructions
Instructions



LES BOUCLES