

# Introduction au langage C

SIF1053

**Présenté par Khadim Io, étudiant en maîtrise Mathématiques Informatiques Appliquées**

Université du Québec à trois rivières  
Baccalauréat en Informatique  
Département Mathématiques-Informatiques  
Architecture des ordinateurs

**Automne 2022**

# Plan

---

- Introduction
- Présentation du langage C
- Les variables, les types de base
- Les opérateurs (d'affectation, arithmétiques, d'assignation, d'incrémentation, logiques, de comparaison)
- Les entrées-sorties
- Structure d'un programme en C
- Installation d'un éditeur de code: Visual Studio Code
- Exemple de programme en C: Conversion d'un nombre entier décimal dans sa forme binaire

# Introduction

---

**Programmation:** Ensemble d'activités pour l'écriture d'un programme

**Langage de programmation:** Un moyen permettant d'écrire des algorithmes

Exemple: C, C++, Python, Java

**Algorithme:** Ensemble ordonné d'opérations et d'instructions visant à résoudre un problème

Exemple: -Trier la liste de noms des étudiants par ordre alphabétique

-Rechercher le plus court chemin entre une source et une destination

# Présentation de langage C

---

**Un peu d'histoire:** Créé vers les années 1970 par Dennis Ritchie et Kenneth Thompson

Dérivé du langage B qui est lui même dérivé du langage BCPL(Basic Combined Programming Language)

**C, un langage impératif:** Paradigme impératif

**C, un langage de bas niveau ou de haut niveau ?**

**C, un langage compilé:** Compilateur

**C, un langage inspirant ?** C++, Java, PHP, etc.

# Les variables et les types de base

---

## Comment nos ordinateurs gardent nos informations ?

### **Analogie:** Marc, le jeune investisseur immobilier

Marc vient de construire un appartement à louer sans chambre parce qu'il tout l'argent pour y arriver ou du moins il ne veut pas prendre le risque de tout construire et de ne pas trouver de clients, huuun ! Il est fou lui ?

Non du tout, la stratégie de Marc c'est de construire une chambre à chaque fois qu'il a une demande.

Le premier client de marc s'appelle François et il est logé dans l'appartement Paradis à la chambre SS\_12.

**Variable:** Un élément qui associe un identifiant à une valeur dans la mémoire programmée.

Maison = Mémoire

Chambre = Identifiant

Client = Valeur

# Les variables et les types de base

---

## Comment nos ordinateurs gardent nos informations ?

Pour gérer son clients, Marc utilise un programme informatique qui lui permet de suivre ses activités. Pour y arriver, il a besoin de leurs **noms** et **prénoms**, leurs, la **somme** à payer par mois et leur donne son **numéro** de compte bancaire pour les virements.

« **François Bonhomme** est un client de Marc, il paye **500.50\$** par mois et fait le virement au compte **00259801**. »

Types de base en trois catégories:

Les entiers: **Integer** en anglais (Mot clé « **Int** »)

Les nombres flottants/décimaux: **Float** en anglais (Mot clé « **Float** »)

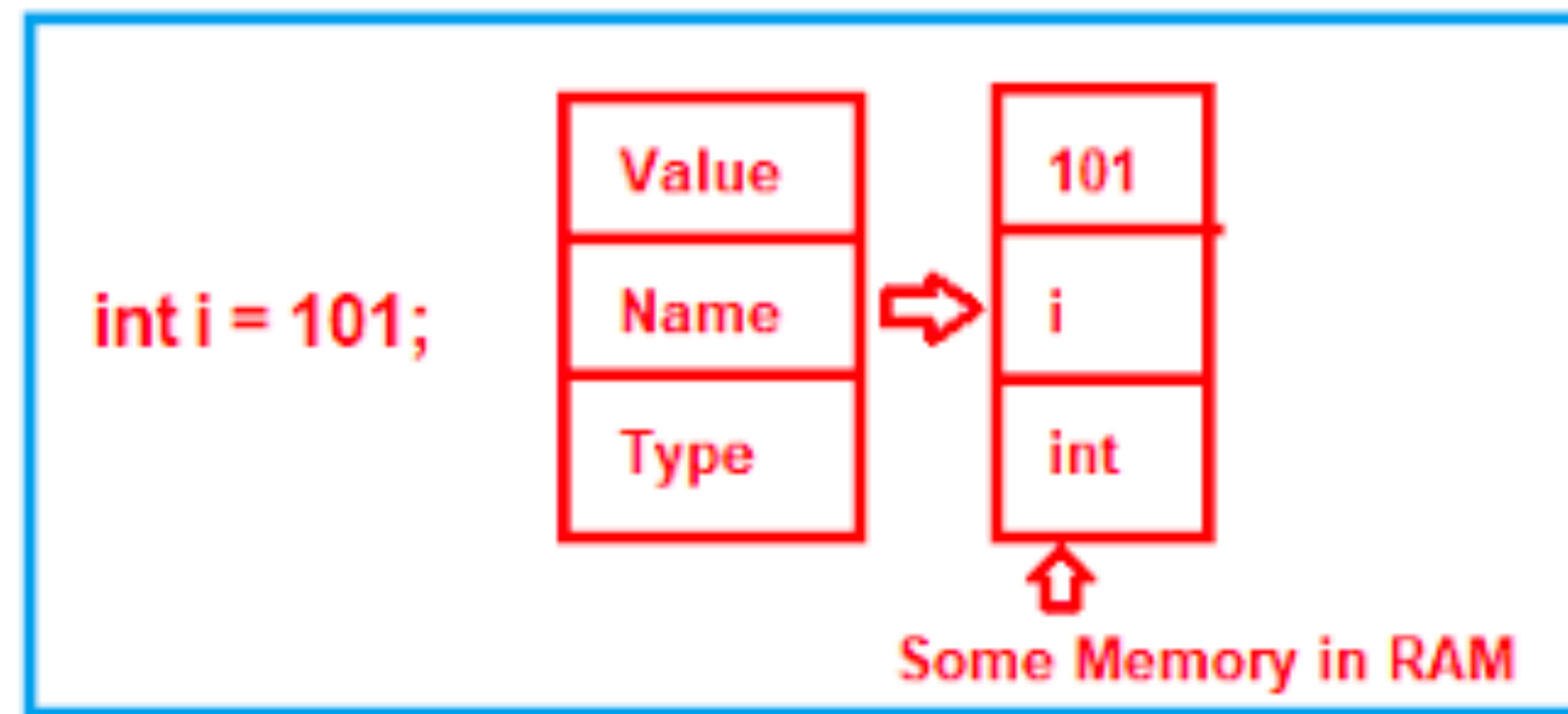
Les caractères: **Character** en anglais (Mot clé « **Char** »)

Selon la taille et le signe: int (**short, long**), float (**double, float**), etc.

**C**, un langage **statiquement/fortement typé**.

# Les variables et les types de base

Comment nos ordinateurs gardent nos informations ?



Exemple d'une variable dans notre mémoire RAM

# Les opérateurs

Des symboles permettant la réalisation d'une opération.

**Affectation:** Affecter une valeur à une variable

Exemple: « = ».  
a = 10;

**Arithmétique:** Effectuer des calculs mathématiques

Exemple: « + », « - », « \* », « / », « % ».  
a = 10;  
b = 20;  
C = a + b;  
**Sortie:** « c = 30 »

**Assignment:** Simplifier des opérations

Exemple: « += », « -= », « \*= », « /= », « % ».  
a += b; <=> a = a + b;  
**Sortie:** « a = 30 »



# Les opérateurs

Des symboles permettant la réalisation d'une opération.

**Incrémentation:** Incrémenter ou décrémenter une variable par unité

Exemple: « ++ », « -- ».

```
a = 10;
```

```
a ++;
```

Sortie: « c = 11 »

**Logiques :** Effectuer des vérifications selon des conditions

Exemple: « && », « || », « ! ».

```
a = 10;
```

```
b = 20;
```

```
c = 30;
```

```
if(a > b && b > c)
```

**Comparaison:** Comparer des variables

Exemple: « == », « < », « > », « <= », « >= ».

```
if(a == b)
```

# Les entrées-sorties

---

## Dialoguer avec la machine

**Notions de fonction:** Un sous-programme qui exécute un ensemble d'instructions bien définies

Exemple: payer(), manger(), rouler(), afficher(), calculer(), etc.

**Notions de bibliothèque:** Une collection de fonctions

Exemple: <stdio.h>, <stdlib.h>, etc.

**printf :** Afficher de l'informations à l'utilisateur

Exemple: printf(« Bienvenue à l'UQTR »);

**scant:** lire une saisie de l'utilisateur

Exemple: scanf(« %d », a);

**Attention aux guillemets (obligatoire) !**

# Structures d'un programme en C

---

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

```
int main(void) {
```

```
    printf("hello, world\n") ;
```

```
    return 0 ;
```

```
}
```

# Installation d'un éditeur de code

---



Code::Blocks



Visual Studio Code

# Notre premier programme en C

---

Ouvrons le fichier: « `coursSIF1053_Dec2Bin` »



Nous n'avons aucun droit sur cette image