

# **Rapport de veille Technologique**

Infelta Elliot - SIO2

## **Sujet - Les langages compilés**

BTS SIO - Services Informatiques aux Organisations  
Option SLAM - Solutions Logicielles et Applications Métiers

Année scolaire 2021-2022

Lycée Saint-Rémi à Roubaix



# **Sommaire**

|   |              |
|---|--------------|
| <b>1) <u>Les langages compilés</u></b>      | <b>3-6</b>   |
| A - Définitions                             | 4            |
| B - Les langages compilés étudiés           | 5-10         |
| <b>2) <u>Évolution et qui choisir ?</u></b> | <b>11-15</b> |
| A - Classement Index TIOBE                  | 11-14        |
| B - La dernière mise à jour du GO           | 15           |

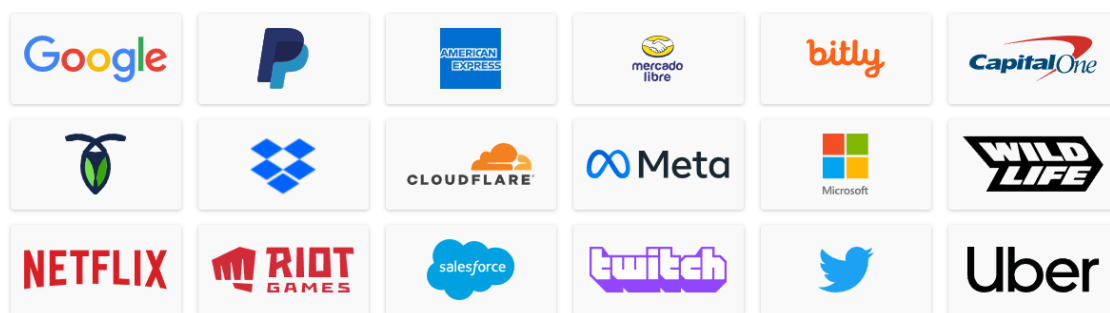
## Ma veille technologique :

J'ai décidé de choisir mon thème de veille autour de la croissance des langages compilés et du langage Go ( Golang ) via leur popularité et de leurs mises à jour perpétuelles.

En effet, le Go est un langage qui m'intéresse car il est utilisé dans de nombreux sites et applications que j'utilise fréquemment comme Twitch, Twitter, Netflix, la plateforme de jeu Riot Games et biens d'autres.

### Entreprises utilisant Go

Les organisations de tous les secteurs utilisent Go pour alimenter leurs logiciels et services [Voir toutes les histoires](#)

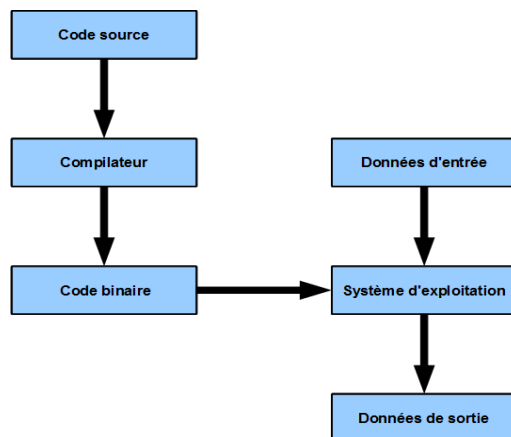


# 1) Les langages compilés

## A - Définitions

### Veille Technologique :

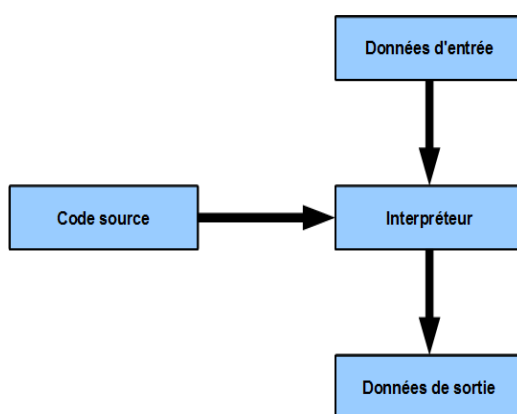
La veille technologique consiste à s'informer en continu sur les nouveautés du secteur informatique, les inventions des concurrents ainsi que les nouvelles technologies et ce dans le but d'être le meilleur dans son domaine.



### Langage compilé :

Les langages compilés ont pour objectif de transformer un code source écrit par un développeur, en code pour la machine (code binaire) qu'un humain ne peut pas lire mais qui est très facile pour un ordinateur.

Cette action est réalisée par un logiciel, dit compilateur permettant au système d'exploitation d'utiliser le code binaire et les données d'entrées pour calculer les données de sortie, le résultat.



### Langage interprété :

Les langages interprétés ont pour objectif de lire le code source écrit par un développeur, de façon à l'interpréter pas à pas, en exécutant les lignes du code une par une, permettant de définir les étapes à réaliser par la suite.

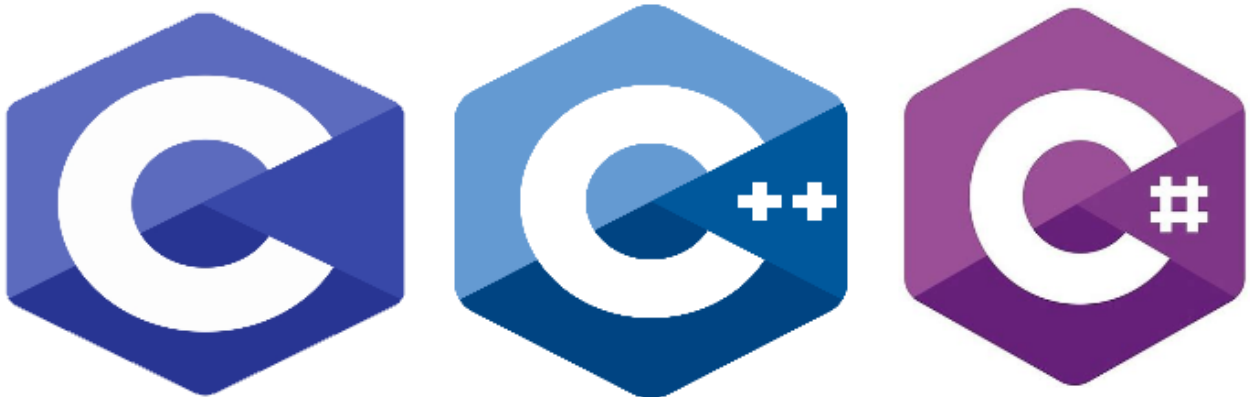
Cette action est réalisée par un interpréteur qui va relire tout le programme à chaque ligne d'instruction et qui va effectuer les actions correspondantes

### Langage intermédiaire :

Les langages intermédiaires sont des langages appartenant aux langages interprétés et compilés car le programme écrit avec ces langages peut dans certaines conditions subir une phase de compilation intermédiaire vers un fichier écrit dans un langage qui n'est pas intelligible (donc différent du fichier source) et non exécutable (nécessité d'un interpréteur).

## A - Les langages compilés étudiés

### Le C, le C++ et le C# :



Inventé au début des années 1970 pour réécrire Unix, C est devenu un des langages les plus utilisés, encore de nos jours.

De nombreux langages plus modernes comme C++, C#, Java et PHP ou JavaScript ont repris une syntaxe similaire au C et reprennent en partie sa logique.

Le C offre au développeur une marge de contrôle importante sur la machine (notamment sur la gestion de la mémoire) et est de ce fait utilisé pour réaliser les « fondations » (compilateurs, interpréteurs...) de ces langages plus modernes.

Le C est un langage de programmation procédural et généraliste.

Il est qualifié de langage de bas niveau dans le sens où chaque instruction du langage est conçue pour être compilée en un nombre d'instructions machine assez prévisible en termes d'occupation mémoire et de charge de calcul.

En outre, il propose un éventail de types entiers et flottants conçus pour pouvoir correspondre directement aux types de données supportés par le processeur.

Enfin, il fait un usage intensif des calculs d'adresse mémoire avec la notion de pointeur.

#### Hello World :

```
#include <stdio.h>

int main(void)
{
    printf("hello, world\n");
    return 0;
}
```



Inventé en 1985, le C++ est un langage de programmation compilé permettant la programmation sous de multiples paradigmes, dont la programmation procédurale, la programmation orientée objet et la programmation générique.

Ses bonnes performances, et sa compatibilité avec le C en font un des langages de programmation les plus utilisés dans les applications où la performance est critique.

Créé initialement par Bjarne Stroustrup dans les années 1980, le langage C++ est aujourd'hui normalisé par l'ISO.

Hello World :

```
#include<iostream>

int main()
{
    std::cout << "Hello, new world!\n";
}
```



Inventé en 2001, le C# est un langage de programmation orientée objet, commercialisé par Microsoft depuis 2002 et destiné à se développer sur la plateforme Microsoft .NET.

Il est dérivé du C++ et très proche du Java dont il reprend la syntaxe générale ainsi que les concepts, y ajoutant des notions telles que la surcharge des opérateurs, les indexeurs et les délégués.

Il est utilisé pour développer des applications web, ainsi que des applications de bureau, des services web, des commandes, des widgets ou des bibliothèques de classes.

Hello World :

```
using System;

public class HelloWorld
{
    public static void Main(string[] args)
    {
        Console.WriteLine("Hello world!");
    }
}
```

Tableau des différences entre le C, C++ et C# :

| <b><u>Le C</u></b>  | <b><u>Le C++</u></b>   | <b><u>Le C#</u></b>   |
|---|--|---|
| Ne supporte pas la programmation orientée objet donc il ne prend pas en charge le polymorphisme, l'encapsulation et l'héritage. | En tant que langage de programmation orienté objet, C++ prend en charge le polymorphisme, l'encapsulation et l'héritage.                               | C# est un langage purement orienté objet. Il apporte un typage sûr, ainsi que les possibilités d'encapsulation, d'héritage et de polymorphisme des langages orientés objet. |
| Le C ne prend pas en charge la surcharge des fonctions et des opérateurs.   | Le C++ supporte à la fois surcharge des fonctions et d'opérateurs.   | Étant un langage dérivé du C++, il supporte à la fois surcharge des fonctions et d'opérateurs   |
| Le C ne fournit pas de support direct pour le traitement des erreurs (également appelé traitement des exceptions).              | Le C++ fournit un support pour la gestion des exceptions. Les exceptions sont utilisées pour des erreurs « difficiles » qui rendent le code incorrect. | Le C# fournit un support pour la gestion des exceptions similaire à celui de JAVA.  |
| Extension du fichier C est .C   | Extension du fichier C++ est .CPP  | Langage uniquement interprétable sur le support ASP.NET.  |

## Le COBOL :



Le Cobol est un langage de programmation créé en 1959. Son nom est l'acronyme de COmmon Business Oriented Language qui révèle sa vocation originelle : être un langage commun pour la programmation d'applications de gestion.

Le Cobol possède plusieurs particularités.

En effet, la totalité des variables et des structures de données utilisées sont définies au début du programme, avant la division procédurale contenant les instructions.

De plus, la définition des variables est faite telle une structure arborescente définie par une suite de lignes de code, chaque ligne commence par un nombre qui définit le niveau d'imbrication du champ ou du groupe de variables.

### Exemple :

```
01 NomPrenom.  
  05 Prenom PIC X(20).  
  05 Nom     PIC X(20).
```

Le Cobol est aussi reconnu pour ses capacités arithmétiques en virgule fixe, notamment pour les traitements par lots où il présente d'excellentes performances. Ceci lui a permis d'être reconnu particulièrement pour les traitements comptables utilisés dans les secteurs de la banque, des assurances, des grandes administrations.

### Exemple :

En Cobol → `ADD montant TO total-jour total-mois total-annee.`

Donne en C →

```
total_jour += montant; total_mois += montant; total_annee += montant;
```

### Hello World (en format libre qui est plus simple de compréhension) :

```
Identification division.  
Program-id. Hello.  
Procedure division.  
Display "Hello world!" line 15 position 10.  
Stop run.
```



## Le Pascal :



Inventé en 1970, le Pascal est un langage de programmation impératif qui, conçu pour l'enseignement, se caractérise par une **syntaxe claire, rigoureuse et facilitant la structuration des programmes**.

Il a été le premier langage de ce type largement répandu dans le monde, et ce, 2 ans avant le langage C. Cependant, il possède des points communs avec le C, le Java, le C++ et le C#.

Le langage Pascal de base était conçu à usage purement éducatif et était assez limité. Par exemple, les chaînes de caractères, absentes du langage d'origine, ont rapidement été intégrées.

Le Pascal possède différentes caractéristiques :

Un langage dit verbeux en anglais; l'utilisation de blocs d'instructions avec Begin et End; l'insensibilité à la casse des mots clés, variables, fonctions et procédures; le typage de données avec la possibilité de définir de nouveaux types et d'en créer.

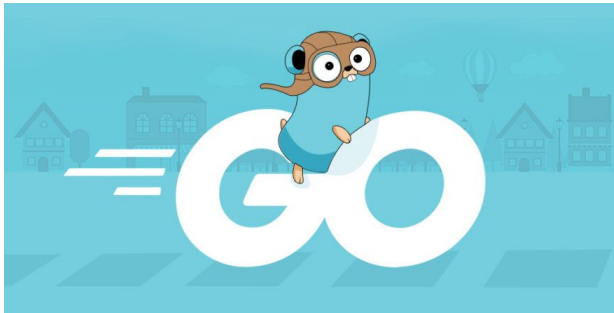
Ainsi que le Type Set permettant des opérations sur des ensembles comme en mathématique; l'orthogonalité soit le fait de décomposer un programme complexe en unités simples, permettant ainsi à chaque composant d'être indépendant des autres.

Enfin la surcharge des opérateurs comme les opérateurs +, and, or, not par exemple; le polymorphisme; la compilation très rapide et la possibilité d'avoir des fonctions récursives permettant à une fonction de s'appeler avec des paramètres différents.

### Hello World :

```
program HelloWorld(output);  
begin  
  writeln('Hello World');  
  readln;  
end.
```

## Le Go ( Golang ) :



Inventé en 2007 et inspiré par les langages du C et du Pascal, le Go est un langage compilé qui est **rapide**, **statiquement typé** et **orienté objet**. Il est aussi **open-source** et est destiné à un usage général.

- *Rapide* -> langage compilé, compris par une machine, pas besoin de lire ligne par ligne
- *Statiquement typé* -> Type des variables vérifié par le compilateur donc pas d'erreur de type + les types des variables ne peuvent pas être changés
- *Orienté objet* -> pas de classe et constructeur mais type personnalisé, des interfaces -> mimiquer le comportement des classes
- *Open-source* -> code source disponible et modifiable par tous .

Après l'introduction officielle de Go en tant que projet open source en fin 2011, Go a rapidement trouvé un grand nombre de soutiens dans la communauté, qui est toujours impliquée dans le développement et l'optimisation du langage de programmation.

La version finale de la première version stable (1.0) a eu lieu le 28 mars 2012 et depuis la version 1.1, qui a suivi un an plus tard, Google publie des mises à jour régulièrement (tous les six mois environ).

Pour utiliser le Go ( Golang ), il est nécessaire d'installer le set-up de la dernière version du Go.

Après l'installation du .exe, il faut créer un fichier.go et, sur l'invitation de commande, saisissez ceci : `go get -v golang.org/x/tools/gopls`

Suite à cela, il est donc possible de faire fonctionner son fichier.go par 2 moyens :

- `go run fichier.go`
- `go build fichier.go` puis `./fichier.go`

Run permet de lancer le fichier golang, celui-ci est plus lent d'exécution que le build.

Build permet de compiler le fichier golang en .exe, celui-ci est rapide à l'exécution.

### Hello World :

```
package main

import "fmt"

func main() {
    fmt.Printf("Hello, world\n")
}
```

## 2) Évolution et qui choisir ?

### A - Classement Index TIOBE

Ma veille technologique se base sur l'évolution des langages compilés et principalement le Golang.

En faisant divers recherches sur le sujet, je me suis rendu compte qu'il n'y avait, à proprement parlé, de classement de popularité et / ou d'utilisation des langages compilés comme le langage C, C++, C#, Cobol, Pascal ou le Golang.

Cependant, via mon flux RSS, j'ai trouvé un classement, l'index TIOBE, permettant de classer les différents langages existant sur leur popularité, et donc de leur utilisation.

#### Qu'est ce que le classement Index TIOBE ?

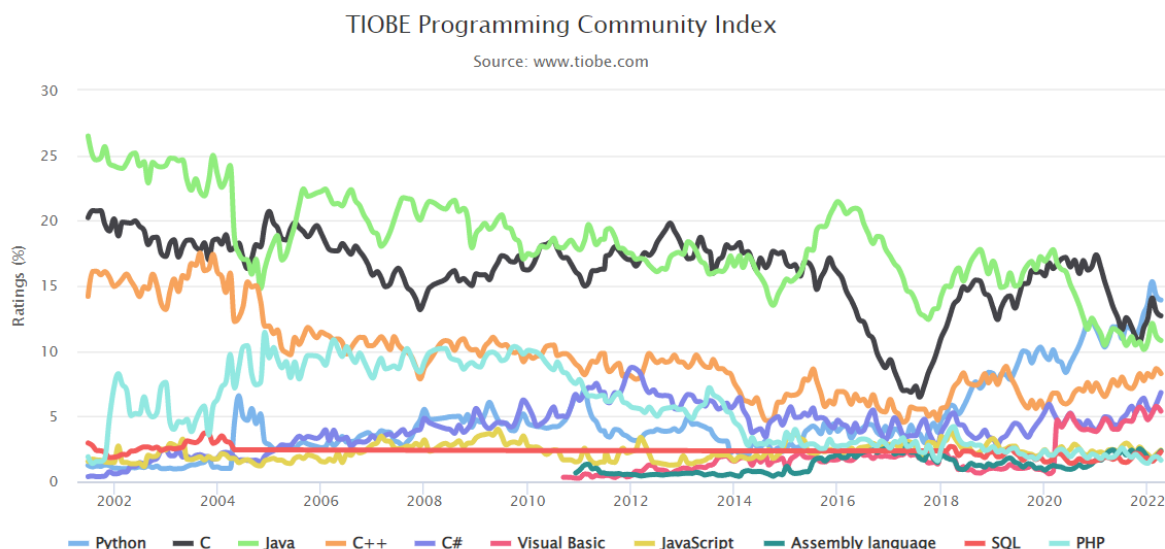
L'index TIOBE mesure la popularité des langages de programmation en se basant sur le nombre de pages web retournées par les principaux moteurs de recherche lorsqu'on leur soumet le nom du langage de programmation.

Il est mis à jour une fois par mois et donne l'historique depuis 2002.

Ce classement est issu de l'entreprise TIOBE Software BV basée aux Pays-Bas, à Eindhoven.

Via ce classement, j'ai pu déterminer des évolutions sur les langages compilés mais aussi interprétés ou intermédiaires.

Dans le graphique des évolutions des 10 langages de programmation les plus populaire, on y retrouve 3 langages compilés, le C, C++ et le C#



( Avril 2022 )

Dans ce graphique nous pouvons observer que le langage C est le second plus populaire avec ces 12.71% de rating à la date du 5 avril 2022. Le C++ est à la quatrième place avec 8.28% et le C# à la cinquième place avec 6.82%.

Depuis le mois de janvier 2022, j'ai suivi l'évolution de 6 langages compilés pris en exemple.

- 1** C (2ème place)
- 2** C++ (4ème place)
- 3** C# (5ème place)
- 4** Go (13ème place)
- 5** Pascal (14ème place)
- 6** Cobol (25ème place)

Voici ci-contre le classement du mois de janvier 2022.

Nous retrouvons à la seconde place le C avec 12.44%. Le C++ à la quatrième place avec 8.29% suivi du C# à la cinquième place avec 5.68%.

À la 13ème place, le Golang avec 1.04% suivi du Pascal dit "Delphi/Object Pascal" avec 0.99%.

Enfin le Cobol à la 25ème position avec un rating de 0.51%.

L'année dernière à la même date, Index TIOBE indique :

- une baisse de 4.94% pour le C et 0.37% pour le Go.
- une hausse de 0.73% pour le C++, 1.73% pour le C# et 0.20% pour le Pascal.

- 1** C (2ème place)
- 2** C++ (4ème place)
- 3** C# (5ème place)
- 4** Go (11ème place)
- 5** Pascal (15ème place)
- 6** Cobol (26ème place)

Voici ci-contre le classement du mois de février 2022.

On retrouve, comme le mois précédent, le même classement cependant le rating est différent.

À la seconde place le C avec 14.08%, le C++ à la 4ème place avec 8.01% suivi du C# avec 5.37%.

Le golang monte à la 11ème place avec 1.23% mais le Pascal et le Cobol descend d'une place avec 0.90% et 0.42%.

L'année dernière à la même date, Index TIOBE indique :

- une baisse de 2.26% pour le C, 0.05% pour le Go et 0.12% pour le Pascal.
- une hausse de 1.13% pour le C++ et 0.93% pour le C#.

- 1** C (2ème place)
- 2** C++ (4ème place)
- 3** C# (5ème place)
- 4** Pascal (12ème place)
- 5** Go (13ème place)
- 6** Cobol (22ème place)

Voici ci-contre le classement du mois de mars 2022.

On retrouve comme les 2 autres mois précédents le même trio de tête des langages compilés.

Le C avec 13.06%, le C++ avec 8.66% et le C# avec 5.92%.

Le Pascal monte à la 12ème place, devant le Go à la 13ème place avec respectivement 1.12% et 0.98% de rating. Enfin le Cobol ferme la marche à la 22ème place avec 0.54%.

L'année dernière à la même date, Index TIOBE indique :

- une baisse de 2.27% pour le C, 0.33% pour le Go et 0.07% pour le Pascal.
- une hausse de 2.14% pour le C++ et 0.95% pour le C#.

**1** C (2ème place)

**2** C++ (4ème place)

**3** C# (5ème place)

**4** Pascal (12ème place)

**5** Go (13ème place)

**6** Cobol (25ème place)

Voici ci-contre le classement du mois d'avril 2022.

On retrouve comme les 3 autres mois précédents le même trio de tête des langages compilés.

Le C avec 12.71%, le C++ avec 8.28% et le C# avec 6.82%.

Le Pascal reste à la 12ème place, devant le Go à la 13ème place avec respectivement 1.18% et 1.09% de rating. Enfin le Cobol ferme la marche à la 25ème position avec 0.55%.

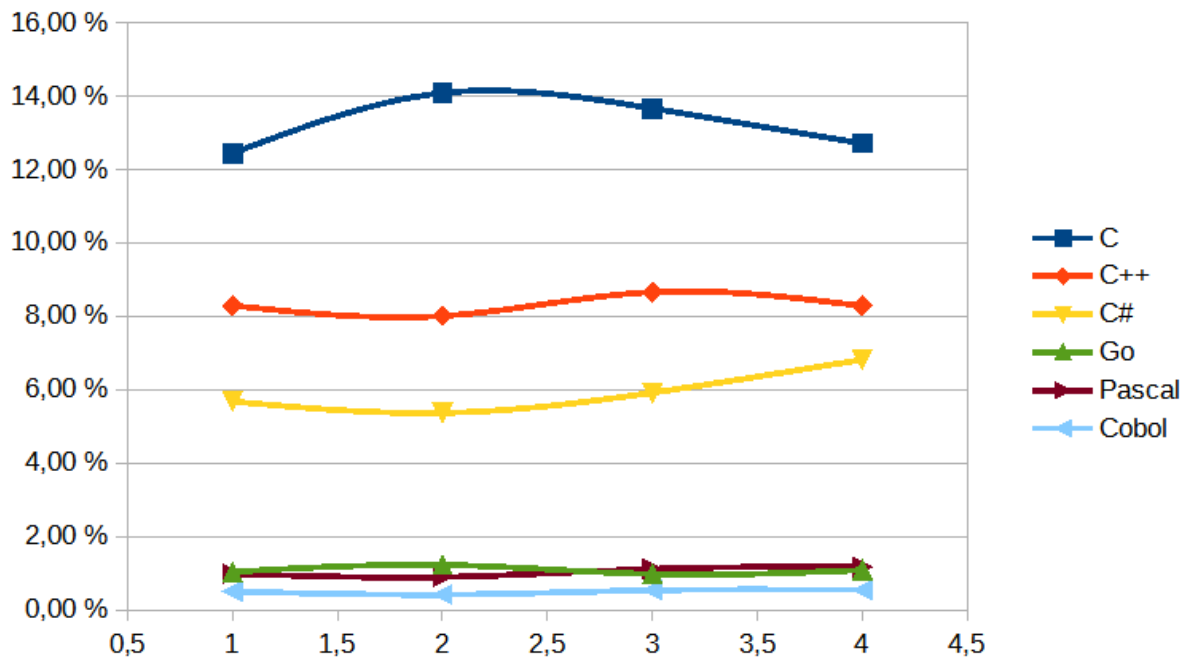
L'année dernière à la même date, Index TIOBE indique :

- une baisse de 1.61% pour le C, 0.14% pour le Go et 0.29% pour le Pascal.
- une hausse de 1.14% pour le C++ et 1.91% pour le C#.

D'après ces 4 classements sur les 4 premiers mois de cette année 2022, nous pouvons construire ces 2 graphiques récapitulatifs de croissance mais aussi de régression de popularité de ces 6 différents langages compilés :

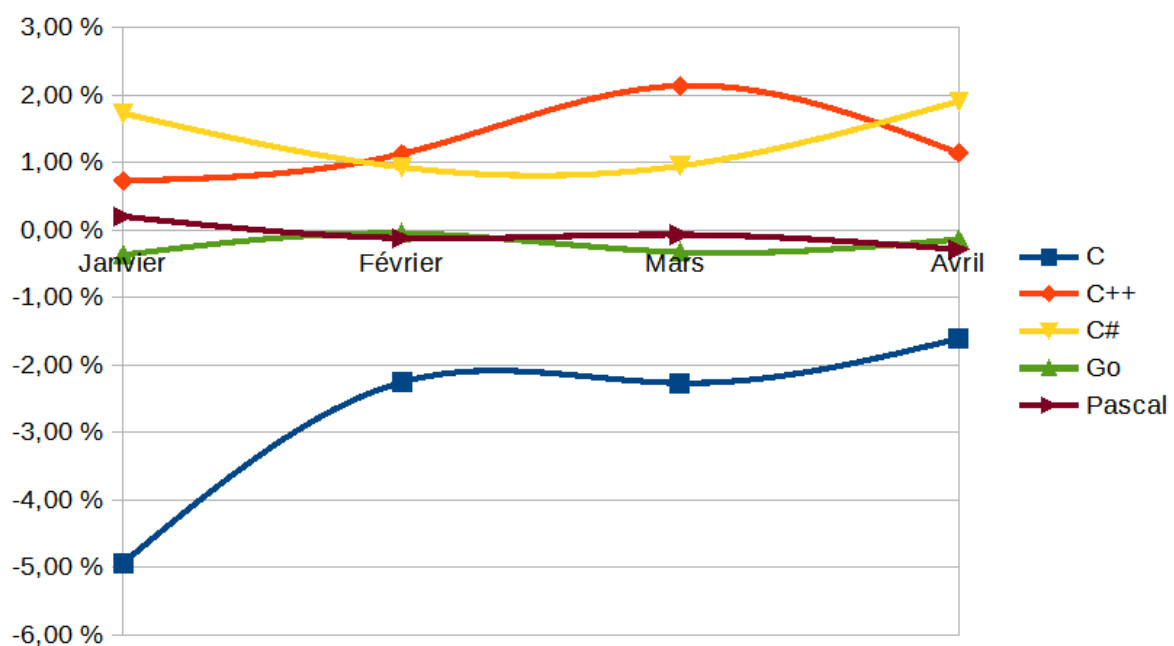
- Premier graphique :

|               | <u>Janvier</u> | <u>Février</u> | <u>Mars</u> | <u>Avril</u> |
|---------------|----------------|----------------|-------------|--------------|
| <u>C</u>      | 12,44 %        | 14,08 %        | 13,66 %     | 12,71 %      |
| <u>C++</u>    | 8,29 %         | 8,01 %         | 8,66 %      | 8,28 %       |
| <u>C#</u>     | 5,68 %         | 5,37 %         | 5,92 %      | 6,82 %       |
| <u>Go</u>     | 1,04 %         | 1,23 %         | 0,98 %      | 1,09 %       |
| <u>Pascal</u> | 0,99 %         | 0,90 %         | 1,12 %      | 1,18 %       |
| <u>Cobol</u>  | 0,51 %         | 0,42 %         | 0,54 %      | 0,55 %       |



- Second graphique :

|                      | Janvier | Février | Mars    | Avril   |
|----------------------|---------|---------|---------|---------|
| <b><u>C</u></b>      | -4,94 % | -2,26 % | -2,27 % | -1,61 % |
| <b><u>C++</u></b>    | 0,73 %  | 1,13 %  | 2,14 %  | 1,14 %  |
| <b><u>C#</u></b>     | 1,73 %  | 0,93 %  | 0,95 %  | 1,91 %  |
| <b><u>Go</u></b>     | -0,37 % | -0,05 % | -0,33 % | -0,14 % |
| <b><u>Pascal</u></b> | 0,20 %  | -0,12 % | -0,07 % | -0,29 % |



## B - La dernière mise à jour du Go

Actuellement le Go est un langage permettant de réaliser différentes application software comme :

- Services Cloud et Réseau ( outils et API )
- Interfaces de ligne de commande ( open source et librairies )
- Développement Web ( performance de mémoire et IDE )
- DevOps et fiabilité du site ( temps de construction rapide, syntaxe allégée, formateur automatique et générateur de documents )

Cependant ces performances et possibilités sont apparus suite à diverses mises à jour plus ou moins importantes d'où la dernière en date est du 15 mars 2022, la version go 1.18.

Celle-ci est une mise à jour majeure pour le Golang. Elle inclut des correctifs de sécurité pour les librairies et packages, des corrections de bugs et problèmes pour :

- le compilateur
- l'éditeur de lien
- le runtime
- la commande vet
- les librairies et packages

Ainsi La dernière version de Go, la version 1.18, est une version importante, comprenant des modifications du langage, de l'implémentation de la chaîne d'outils, du runtime et des bibliothèques.

Voici une liste des changements les plus visibles.

- La syntaxe des déclarations de fonction et de type accepte désormais les paramètres de type.
- Les fonctions et les types paramétrés peuvent être instanciés en les suivant avec une liste d'arguments de type entre crochets.
- Le nouveau jeton a été ajouté à l'ensemble des opérateurs et de la ponctuation. ~
- La syntaxe des types d'interface permet désormais l'incorporation de types arbitraires (pas seulement de noms de types d'interfaces) ainsi que d'éléments d'union et de type. Ces interfaces ne peuvent être utilisées que comme contraintes de type. Une interface définit désormais un ensemble de types ainsi qu'un ensemble de méthodes. ~T
- Le nouvel identificateur prédéclaré est un alias pour l'interface vide. Il peut être utilisé à la place de . anyinterface{}
- Le nouvel identificateur prédéclaré est une interface qui désigne l'ensemble de tous les types qui peuvent être comparés à l'aide de ou . Il ne peut être utilisé qu'en tant que (ou incorporé dans) une contrainte de type. comparable==!=

## **Mes sources :**

### **Pour les définitions :**

Qu'est-ce que la veille technologique ? Définition et Enjeux (datascientest.com)

France-IOI – Langages interprétés et langages compilés

### **Pour le C, le C++ et le C#:**

C (langage) – Wikipédia (wikipedia.org)

C++ – Wikipédia (wikipedia.org)

C# – Wikipédia (wikipedia.org)

Différences entre C et C++ - WayToLearnX

Différence entre C++ et C# - WayToLearnX

### **Pour le COBOL :**

Cobol – Wikipédia (wikipedia.org)

### **Pour le Pascal :**

Pascal (langage) – Wikipédia (wikipedia.org)

### **Pour le GOLANG :**

Go (langage) – Wikipédia (wikipedia.org)

(6) [0/??] Introduction de la série | Tutoriel Français Golang pour débutants 2021 - YouTube

### **Pour l'index TIOBE :**

Index TIOBE – Wikipédia (wikipedia.org)

index | TIOBE - The Software Quality Company

TIOBE Index for March 2022

TIOBE Index for February 2022

TIOBE Index for January 2022

### **Pour l'évolution du Go :**

<https://qarea.com/blog/the-evolution-of-go-a-history-of-success>

<https://go.dev>

<https://go.dev/doc/devel/release>