

# Runtrack C++ – Basics

## Introduction

---

Créez sur GitHub un répertoire nommé “runtrackC++”. Dans ce répertoire, créez un dossier “**jour01**”. Pour chaque job, créez un dossier “**jobXX**” où **XX** est le numéro du job. N’oubliez pas d’envoyer vos modifications dès qu’un job est avancé ou terminé et mettez des commentaires explicites lors de vos commits.

Le langage C++ étant un langage compilé, il est nécessaire pour pouvoir exécuter votre code d’utiliser un **compilateur**.

Nous utiliserons GCC (GNU Compiler Collection).

En fonction de votre OS installer votre environnement :

- **Linux** : GCC devrait être déjà présent. Si ce n’est pas le cas, l’installer avec votre package manager (apt, pacman...)
- **Mac** : GCC (ou clang, qui est une bonne alternative) devrait être déjà présent. Si ce n’est pas le cas, installer GCC avec votre package manager (brew)



→ **Windows** : Suivre le guide d'installation de [MSYS2](#), y compris

l'installation de GCC, puis ajouter le dossier bin/ de MSYS2 contenant les exécutables de GCC a votre variable d'environnement système Path.

Utiliser la commande "g++ -v" pour vérifier que le compilateur est bien installé et présent dans la variable d'environnement PATH.

## Job 0

---

Éliminer les parenthèses superflues dans les expressions suivantes :

```
// Which parentheses ?  
a = (x+5)  
a = (x=y) + 2  
a = (x==y)  
(a<b) && (c<d)  
(i++) * (n+p)
```

## Job 1

---

Soit les déclarations ci-dessous :

```
char c = '\x01' ;  
short int p = 10 ;
```

Quels sont le type et la valeur de chacune des expressions suivantes :



```
x = p + 3
y = c + 1
z = p + c
w = 3 * p + 5 * c
```

## Job 2

---

Écrire un programme en C++ permettant d'afficher le message "**Hello World**".

## Job 3

---

Écrire un programme en C++ permettant d'afficher le message "**Hello World**" **N** fois, tant que **N** n'est pas égal à 0.

## Job 4

---

Écrire un programme en C++ permettant **d'additionner** deux nombres et **d'afficher leur addition**.

## Job 5

---

Écrire un programme en C++ permettant de **multiplier** deux nombres et **d'afficher leur produit**.



## Job 6

---

Écrire un programme en C++ qui affiche la table de multiplication d'un entier donné par l'utilisateur.

## Job 7

---

Écrire un programme en C++ qui permet d'afficher si un nombre entier saisi au clavier est **pair** ou **impair**.

## Job 8

---

Écrire un programme qui demande à l'utilisateur d'entrer une année, puis détermine si cette année est bissextile ou non. Par exemple, les années 2020, 2024 et 2028 sont bissextiles, mais les années 1900, 2100, 2200 ne le sont pas.

## Job 9

---

Écrire un programme en C++ qui permet **d'afficher** le plus grand de trois entiers saisis par l'utilisateur.

## Job 10

---

Vous êtes marchand de fruits et légumes à Noailles. Vous avez besoin d'un programme qui affiche **le prix TTC** de vos produits frais. Écrire un programme en C++ qui demande à l'utilisateur de taper le prix HT d'un kilo de carottes, le



nombre de kilos de carottes, le taux de TVA (ex : 15 %). Le programme doit afficher le prix TTC d'une marchandise donnée.

## Job 11

---

Écrire un programme en C++ qui permet d'échanger le contenu de deux entiers **n** et **m** saisis par l'utilisateur, puis et afficher ces entiers avant et après l'échange.

## Job 12

---

Écrire un programme en C++ qui demande à l'utilisateur de taper cinq entiers et qui affiche leur moyenne. **Vous êtes limités à deux variables.**

## Job 13

---

Écrire un programme qui demande à l'utilisateur de taper **un entier N** et qui calcule la somme des cubes de  **$5^3$  à  $N^3$** .

## Job 14

---

Écrire un programme en C++ qui demande à l'utilisateur d'entrer un nombre, puis inverse ce nombre et l'affiche à l'utilisateur. **Par exemple : 12345 devient 54321.**



## Compétences visées

---

→ C++

## Rendu

---

Le projet est à rendre sur <https://github.com/prenom-nom/runtrackC++>.

## Bases de données

---

- [Apprendre le c++](#)
- [La syntaxe de base](#)
- [Aide mémoire C++](#)
- [Les boucles](#)