

[A mettre apres les chapitre sur les fonctions et les tests ?](#)

Ce chapitre est encore en cours de rédaction, voire à l'état d'ébauche. N'hésitez pas à faire des remarques ou poser des questions sur le forum de [Zeste de Savoir](#) ou de [OpenClassroom](#).

Entrée console

qu'est ce que la console/terminale? Comment la lancer?

On connaît déjà `cout`, qui signifie "output C console", permet d'envoyer des messages sur la console. Également possible d'entrée des valeurs depuis la console avec `std::cin`

Note : non testable sur Coliru.com, besoin de compiler en local.

Permet d'attendre des instructions de l'utilisateur. Mode interactif : on lance le programme, celui-ci s'arrête, le temps de saisir des infos.

Différent de ligne de commande et `argc/argv` ⇒ info envoyée lors du lancement du programme, pas besoin de mettre en pause ou interaction dynamique avec l'utilisateur. (Note : création de scripts automatique, lancée par des robots, plus pratiques d'utiliser la ligne de commande. Certains outils proposent les 2, comme `gdb -i` pour lancer en mode interactif)

Les 2 sont possibles.

Exemple : (ligne commençant par `>` = ce que l'utilisateur tape sur une console/terminal)

```
> dosomething unfichier.txt
Vous voulez ouvrir le fichier "unfichier.txt". Confirmer
(o/n).
> o
Lecture du fichier "unfichier.txt" en cours...
```

```
Fin de lecture du fichier.
```

Première ligne = lance le programme "dosomething" avec un argument "unfichier.txt".
Seconde ligne : le programme demande confirmation et attend une réponse (taper sur les touches "o" ou "n" puis entrer).
Troisième ligne: le programme continue.

Ici, 2 modes d'entrée d'inso : via argument lors du lancement, et en interactif. Le programme attend la confirmation avant de poursuivre son travail.

Saisie simple

Création d'une variable, puis utilisation de `>>` avec `std::cin`. Typage fort du C++ fait que `std::cin` lit correctement le type de valeur selon le type.

```
#include <iostream>

int main() {
    int i {};
    std::cin >> i;
    std::cout << "input: " << i << std::endl;
}
```

Pour lancer le programme :

```
> a.out 123
input: 123
```

Entrer une ligne de texte

getline j'ai écrit sur le bac à sable, mais je préfère une approbation d'un programmeur plus expérimenté d3m0t3p

Gestion des erreurs

filtrer le texte entré, vérifier qu'il est conforme

pas encore vu les boucles...

Chapitre précédent	Sommaire principal	Chapitre suivant
---------------------------	---------------------------	-------------------------

[Cours, C++](#)