

[http://genelaix.free.fr/IMG/pdf/threads\\_cours\\_tp.pdf](http://genelaix.free.fr/IMG/pdf/threads_cours_tp.pdf)

[https://www.bogotobogo.com/cplusplus/C11/1\\_C11\\_creating\\_thread.php](https://www.bogotobogo.com/cplusplus/C11/1_C11_creating_thread.php)

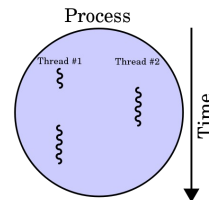
### Les threads en C++

Les threads (que l'on peut traduire par « fil d'exécution ») peuvent être mis en place sur une machine équipée d'un système d'exploitation multitâches

Un thread permet au sein d'un programme d'exécuter des codes différents en parallèle tout en partageant la mémoire du programme.

Le code exécuté dans la fonction main sera appelé le « main thread » ou thread principal, les autres threads, threads secondaires.

Comme le code est exécuté en parallèle, les fonctions bloquantes (ex while(1) ) ne sont plus un problème, seul le thread qui les exécute sera bloqué, le reste du programme pouvant continuer son exécution.



Soit le programme de test suivant qui n'utilise pas de threads:

```
#include <iostream>

using namespace std;

// affN affiche des nombres de 0 a maxi separes par une virgule
void affN(int maxi)
{
    int i=0;
    while(i<=maxi)
    {
        cout << i++ << ", "; // ici deux ecritures sur la console
    }
}

// affC affiche des caracteres ASCII de debut a fin separes par un tiret
void affC(char debut, char fin)
{
    char c=debut;
    while(c<=fin)
    {
        cout << c++ << "-"; // ici deux ecritures sur la console
    }
}

int main()
{
    affN(9);
    affC('A', 'Z');
    return 0 ;
}
```

Le résultat est :

0,1,2,3,4,5,6,7,8,9,A-B-C-D-E-F-G-H-I-J-K-L-M-N-O-P-Q-R-S-T-U-V-W-X-Y-Z-

On constate que les deux séries sont consécutives