

Variables static

Ámbito de una variable

- Las variables locales que hemos usado hasta ahora “mueren” cuando acaba la ejecución de la función
- Las variables **static** permanecen, y retienen su valor entre llamadas

```
void function()  
{  
    static <variable_type> <variable_name>  
    ...  
}
```

Reglas de uso

1. La declaración de una variable estática se ejecuta sólo una vez: la primera vez que se ejecuta la función que la contiene.
2. Una variable estática se inicializa solo una vez (como parte de la declaración). Será inicializada al valor 0 si no se indica lo contrario.
3. Cuando se produce una llamada a la función en la que reside la variable estática, ésta mantiene el ultimo valor que poseía al finalizar la anterior llamada a dicha función.

```
void my_function()
{
    static short count = 1;
    cout << "this function has been called "
          << count << "time(s) ";
    count++;
    return;
}

int main()
{
    for (int i = 0; i < 5; i++)
        my_function();
    return 0;
}
```

```
void is_it_larger(const float val)
{
    static float largest = val;
    if (largest < val)           // new val is larger
        largest = val;
    cout << "The largest value sent to this function "
         << "so far is " << largest << endl;
    return;
}

int main()
{
    for (int i = 0; i < 7; i++)
        is_it_larger(i);
    return 0;
}
```

```
#include <iostream>
#include <ctime>           // access to time clock

long my_rand()
{
    static unsigned long seed = time(NULL);
    seed = (104729 + seed * 7919) % 15485863;
    return seed;
}

int main()
{
    for (int i = 5; i > 0; i--)
        cout << my_rand() % 5 << endl;
    return 0;
}
```