

Listado de errores a evitar

Errores fatales (😱)

Entregar una práctica con los siguientes errores implica que lo nota obtenida por el el alumno es **un 0 (cero)**.



- La práctica no compila
- No se indican los apellidos y el nombre del alumno como primera línea comentada del fichero entregado. Si es un grupo de alumnos hay que poner los apellidos y nombre de cada alumno en una línea

```
// Apellido1 Apellido2, Nombre del alumno  
  
#include <stdio.h>  
...
```

- El fichero entregado tiene un nombre diferente a ***ejercicios1.c*** o cualquiera de sus versiones en Canvas (***ejercicios1-1.c, ejercicios1-2.c, ...***)
- Los nombres de los procedimientos que implementan cada uno de los ejercicios no concuerdan con los que aparecen en el enunciado de la práctica. Los nombres de los procedimientos deben corresponder con los siguientes:
 - void ejercicio1();
 - void ejercicio2();
 - void ejercicio3();
 - void ejercicio30p();

Cartagena99

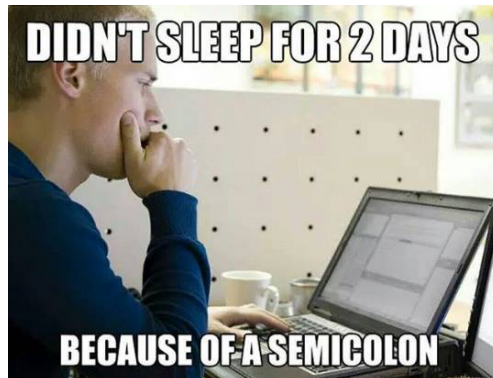
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70

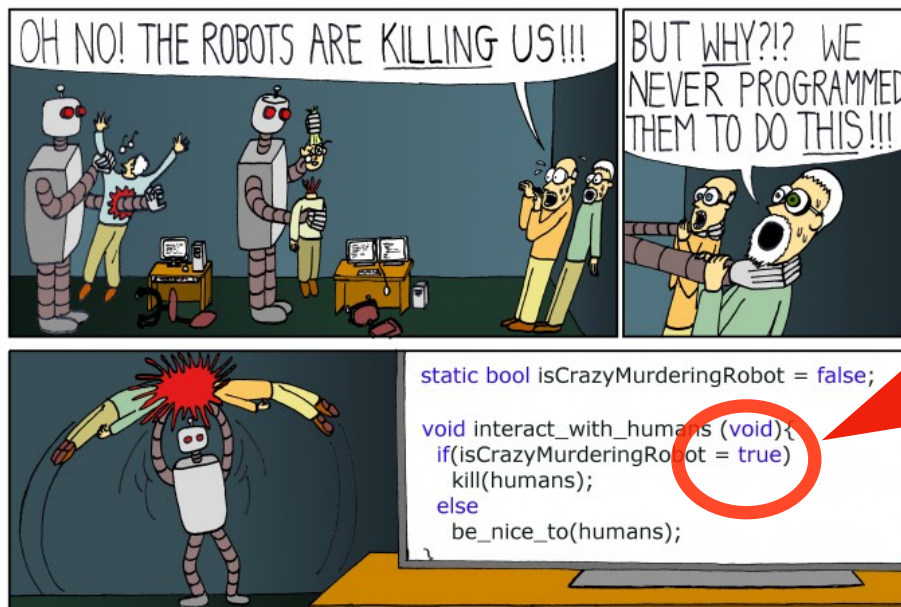
Errores graves

Si en el fichero **ejercicios1.c** entregado por el alumno, se comente alguno de los siguientes errores se **descontarán 3 puntos** a la nota final obtenida en la práctica.

- Modificación de la variable iteradora dentro de un bucle for
- Uso incorrecto del punto y coma al definir una sentencia if o un bucle



- Uso de variables globales
- Utilizar goto
- Utilizar break para detener la ejecución de un bucle
- Utilizar un return para detener la ejecución de un bucle
- Utilizar continue dentro de un bucle
- Hacer uso de una variable sin valor inicial
- Confundir el operador de asignación = con el operador de comparación ==



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70

Cartagena99

- Uso de funciones no importadas o no declaradas previamente en el código
- Asignar variables a si mismas (**var = var**)
- Código duplicado
- Uso de un tipo de dato incorrecto
- Ramas vacías en instrucciones **if**, **if-else** o **if-else if- else**
- Hacer que un bucle ejecute su bloque de instrucciones sin hacer nada en lugar de pararlo mediante la condición correspondiente
- Hacer comparaciones consigo mismo (**var == var**)
- Poner en el código instrucciones que se pueden realizar mediante estructuras repetitivas
- Generación de número aleatorios sin acotar

Errores leves

Si en el fichero **ejercicios1.c** entregado por el alumno, se comente alguno de los siguientes errores se **descontará 1 punto** a la nota final obtenida en la práctica.

- Código sin formatear (código sin indentar, todo el código a la misma altura, sin espacios en blanco, sin saltos de línea, código ofuscado, ...)
- Definición de variables entre las líneas de código en lugar de al principio de los procedimientos o funciones
- Warnings en tiempo compilación (excepto el provocado por definir la función *main* como *main()*). Para mostrar todos los *warnings* se debe utilizar la opción **-Wall** cuando se invoca al compilador
- Definir variables que no se utilizan en el código



Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70