



UNIVERSIDAD DE MÁLAGA
DPTO. DE LENGUAJES Y C. DE LA COMPUTACIÓN
E.T.S. DE INGENIERÍA INFORMÁTICA

FUNDAMENTOS DE LA PROGRAMACIÓN

Obligaciones y Recomendaciones en cuanto al Estilo de Programación Estructurada

Obligaciones: (su incumplimiento implicará directamente una calificación de 0 puntos en el problema)

1. El uso de variables globales está prohibido.
2. Sólo están permitidas las sentencias y estructuras de selección e iteración que aparecen en los apuntes y que se han visto en clase (`if`, `switch`, `while`, `do-while`, `for`, etc.). No se podrán usar otras como `goto`, `continue`, etc.
3. La sentencia `break` sólo se podrá utilizar en la estructura `switch`.
4. No modificar la variable de control de un bucle `for` dentro de su cuerpo.

Recomendaciones:

1. En un programa, tras la inclusión de las bibliotecas de C++, se definirán las constantes y tipos (enumerados (`enum`), registros (`struct`) y arrays (`typedef`)) necesarios.
2. Dentro de un procedimiento o función (incluida la función `main`) primero se declararán todas las variables locales necesarias (a excepción de la variable de control de un bucle `for`, que puede declararse en él) y después aparecerán las sentencias.
3. En una función, sólo aparecerá una sentencia `return`, y será la última sentencia de dicha función.
4. Para el diseño de un bucle, se utilizará la estructura `for` cuando el número de repeticiones o iteraciones se conozca a priori. En otro caso (el número de repeticiones depende de alguna condición que se va modificando con las distintas iteraciones) se utilizarán las estructuras `while` y `do while`.
5. Definir siempre un bloque (secuencia de sentencias entre llaves) para las estructuras de selección e iteración, incluso si sólo hay una sentencia en la secuencia de sentencias.
6. Para el paso de parámetros de tipos estructurados se seguirá el criterio de utilizar siempre el paso por referencia. En el caso de que el parámetro sea de entrada, el paso por referencia será constante (`const`).

7. Para la definición de identificadores es conveniente seguir las siguientes pautas:

a) El identificador de una constante irá en mayúsculas.

Por ejemplo: `const int MAX= 100;`

b) El identificador de un tipo de datos comenzará por T.

Por ejemplo: `typedef int TVector[MAX];`

c) El identificador de una variable irá en minúscula.

Por ejemplo: `TVector vector;`