

Enunciado actividad de documentación utilizando *doxygen*

Objetivo

El objetivo de esta actividad es convertir nuestro código en una aplicación bien documentada. Para ello, además de seguir el material didáctico *Guía básica de estilo de programación* visto con anterioridad, se requiere que parte de la documentación del código fuente se realice empleando la herramienta *doxygen*.

Introducción a la herramienta *doxygen*

Doxygen es una herramienta que permite generar documentación con formato a partir de los comentarios incluidos en el propio código fuente. La herramienta soporta diferentes lenguajes de programación como Fortran, C, C++, Java, VHDL, PHP, C# o Python, entre los más populares.

La inclusión de comentarios que pueden ser procesados por *doxygen* se realiza utilizando una sintaxis especial que incluye etiquetas, que son textos predefinidos con sentido para la herramienta.

A continuación se muestra un ejemplo de un comentario en una cabecera de un fichero .h

```
/**
 * @brief Utilidades de manejo de caracteres
 *
 * Este modulo contiene los prototipos de las funciones de manejo de
 * caracteres.
 * @file utilcad.h
 * @author Profesores PPROG
 * @version 1.0
 * @date 23-01-2015
 */
```

Los comentarios que son interpretados por *doxygen* tienen que comenzar con `/**` y terminar con `*/`. Las etiquetas de *doxygen* comienzan con `@`, y permiten definir campos dentro de los comentarios que serán tratados de manera especial por la herramienta. En el caso de los ficheros en concreto, `@brief` sirve para incluir un resumen sobre la utilidad del módulo/archivo, `@file` para indicar el nombre del fichero, `@author` los nombres de los autores, `@version` la versión del documento, y `@date` la fecha del mismo.

Las funciones siguen una sintaxis similar:

```
/**
 * @brief compara dos cadenas de caracteres
 *
 * strcmp() compara lexicograficamente dos cadenas de
 * caracteres s1 y s2.
 * @param s1 cadena de caracteres correctamente formada.
 * @param s2 cadena de caracteres correctamente formada.
 * @return entero mayor, igual, o menor que 0 dependiendo de si
 * s1 es mayor, igual o menor que s2
```

```
*/  
int strcmp(const char *s1, const char *s2);
```

La etiqueta *@brief* se emplea al comienzo de los comentarios de cabeceras de funciones y de estructuras y tipos de datos, además de los comentarios de cabecera de ficheros fuentes, también para definir una frase resumen de cada uno de ellos. En el caso de las funciones, la etiqueta *@param* permite comentar los parámetros de las mismas, mientras que la etiqueta *@return* permite describir los valores de retorno.

Constantes, variables globales, tipos enumerados y estructuras de datos se comentan de manera ligeramente distinta:

```
const double NA = 6.02214179E23;          /*!< Numero de Avogrado */  
  
/**  
 * @brief Lista de posibles errores  
 */  
enum Errores {  
    OK,          /*!< No hay error */  
    ERROR_MEM,   /*!< Error en la reserva de memoria */  
    ERROR_FICH   /*!< Error al abrir el fichero */  
};  
  
/**  
 * @brief Punto en 2D  
 */  
 * Esta estructura define un punto en dos dimensiones.  
 */  
typedef struct  
{  
    float x;      /*!< Coordenada x */  
    float y;      /*!< Coordenada y */  
} Punto;
```

Nótese que los campos individuales se comentan con */*!< ... */*. Hay que tener en cuenta que *doxygen* dispone de varios estilos de sintaxis habiéndose elegido la que se muestra en los ejemplos por su similitud con la documentación *javadoc* de Java.

Ejecución

Antes de ejecutar *doxygen* se recomienda crear el fichero de configuración *Doxyfile*:

```
> doxygen -g
```

Para ejecutar *doxygen*:

```
> doxygen Doxyfile
```

Uso de doxygen

La herramienta *doxygen* incluye un conjunto bastante amplio de etiquetas¹. Ahora bien, como se ha podido comprobar, la inclusión de estas etiquetas reduce la legibilidad del código fuente, por lo que se recomienda su uso moderado. Las

¹ <http://www.stack.nl/~dimitri/doxygen/commands.html>

etiquetas incluidas en este documento están dentro de las más habituales².

Las partes del código fuente que tiene sentido comentar empleando la sintaxis de *doxygen* siempre van a ser las partes públicas, es decir, aquellas que se corresponda con ficheros “.h” y los comentarios de cabecera de los ficheros “.c”. Queda desaconsejado su uso en el resto del código. En especial, dentro del código de las funciones.

Entregables y criterios de corrección

El equipo deberá realizar la documentación del código de la I3 para su entrega junto con todos los ficheros creados de dicha I3.

La evaluación de esta actividad se engloba en la del entregable de la I3, según la rúbrica incluida en el correspondiente enunciado.

Links de interés.

<http://www.stack.nl/~dimitri/doxygen/>

<http://plugins.netbeans.org/plugin/14326/doxygen-integration>

<http://doxymacs.sourceforge.net/>

² En algunos proyectos @date y @version se omiten ya que existen herramientas de control de versiones que generan automáticamente esta información.