

Universidad Popular Autónoma del estado de Puebla

Facultad de Tecnologías de Información

Estándar de Codificación y documentación para programas en C Versión 1.0.

Propósito	El siguiente documento tiene como propósito definir la forma en que los programas deberán documentarse y construir para ser entregados al profesor.
Estándares de Documentación	
Encabezado del programa	Todos los programas deberán tener un encabezado que dé una descripción del programa además de información. Esta deberá enmarcarse en un cuadro de asteriscos
Formato del encabezado	<pre> /***** * Nombre_del_archivo.c * * Breve descripción del programa * * Versión: Número de Versión del programa. Cambia si se modifica* * * el programa.¹ * * Autor: Tu nombre * * Fecha: Fecha de realización del programa * *****/ </pre>
Ejemplo de encabezado	<pre> /***** * BINARY.C * * Este programa ilustra los métodos de búsqueda binarias iterai- * * va y recursiva * * Versión: 1.0 * * Autor: Roberto Salazar Márquez * * Fecha: 15/03/2005 * *****/ </pre>
Formato de secciones	Cada sección del programa deberá separarse con un comentario en asteriscos con el nombre de la sección al centro y con una línea en blanco antes y después de la misma.
Ejemplo de una sección	<pre> /***** Definición de funciones *****/ </pre>
Documentación para una función	Todas las funciones deberán documentarse para hacer mas clara su definición.
Formato de Documentación de Función	<pre> /***** * Función: Nombre de la función * * Argumentos: Nombre - Descripción del objetivo del argumento. * * Todos y cada uno de los argumentos deben definirse * * Valor de Retorno: Tipo y descripción del valor de retorno y si * * regresa valores específicos describir cada uno* * * de ellos. * * Comentario: Descripción del la tarea realizada por la función * * en un arreglo de forma iterativa. * *****/ </pre>
Ejemplo de documentación de Función	<pre> /***** * Función: Binariaitv * * Argumentos: arr[] - Arreglo de enteros donde se buscará el num. * * n - numero a buscar * * Valor de Retorno: Numero de posición (índice) donde está el n- * * mero buscado, -1 si no se encontró * * Comentario: Versión iterativa de la función que busca un número * * en un arreglo de forma iterativa. * *****/ </pre>
Comentarios	<ul style="list-style-type: none"> • Documentar el código para que el lector pueda entender la operación • Los comentarios deberán explicar tanto el propósito como la conducta del código • Comentar la declaración de las variables para indicar su propósito.
Buen Comentario	<code>if(num_registros > limite) /* Todos los registros se procesaron? */</code>
Mal comentario	<code>if(num_registros > limite) /* checar si num_registros es menor que */</code> <code>/* el limite */</code>
Secciones Mayores	Cuando existan bloques de código con un solo propósito, (Secciones mayores) estas deberán incluir un comentario que describa que se hará en la siguiente sección
Ejemplo de Documentación de una sección mayor	<pre> /***** * Esta sección del programa implementará las definiciones de cada * * una de las primitivas de la Pila * *****/ </pre>
Estándar de Codificación	
Líneas en blanco	Debe haber las líneas en blanco suficientes para separar las estructuras del programa y hacerlas mas claras

Identificadores	Use nombres descriptivos para todas las variables, funciones, constantes, etc. Evite en lo más posible, el uso de letras simples. Cuando el identificador esté formado por varias palabras estas se separarán por el guión bajo (_).
Ejemplos de Identificadores	<pre>int numero_estudiantes, edad; /* BUEN identificador */ char vqgli, angi, ddd; /* Estos están MAL */</pre>
Indentar (Sangría)	<ul style="list-style-type: none"> • Indentar cada bloque o nivel de instrucciones a un nivel mas adentro que el anterior • Las llaves se abren y cierran en su propia línea y se encuentran alineadas.
Ejemplo de Indentación	<pre>void captar_moneda(int monedas) { if(monedas==precio) /* se dio el precio exacto */ { printf("Depositado \$%d\n",precio); estado=1; /* La maquina cambia de estado a */ /* esperar selección */ } }</pre>
Mayúsculas y minúsculas	<ul style="list-style-type: none"> • Todas las instrucción #define deberán estar en mayúsculas • Todos los demás identificadores deberán ir con minúsculas
Ejemplo de Mayúsculas y minúsculas	<pre>#define MAXARRAY 30 int calificaciones[MAXARRAY]; void paso_argumentos_demo(int args)</pre>

¹ El número de versión se divide en **VersiónMayor.VersiónMenor**. Cada vez que el código se modifique deberá incrementarse el número de versión. La naturaleza del cambio definirá cual es la versión que cambia. Si el cambio es para corregir defectos o pequeños detalles cambiará el número de versión menor. Si el cambio es significativo, es decir estructural o conceptual, la versión que cambia es la Mayor.