



aprenderaprogramar.com

# Introducción a la mejora de programas. Documentación y mantenimiento. (CU00247A)

Sección: Cursos

Categoría: Curso Bases de la programación Nivel II

Fecha revisión: 2024

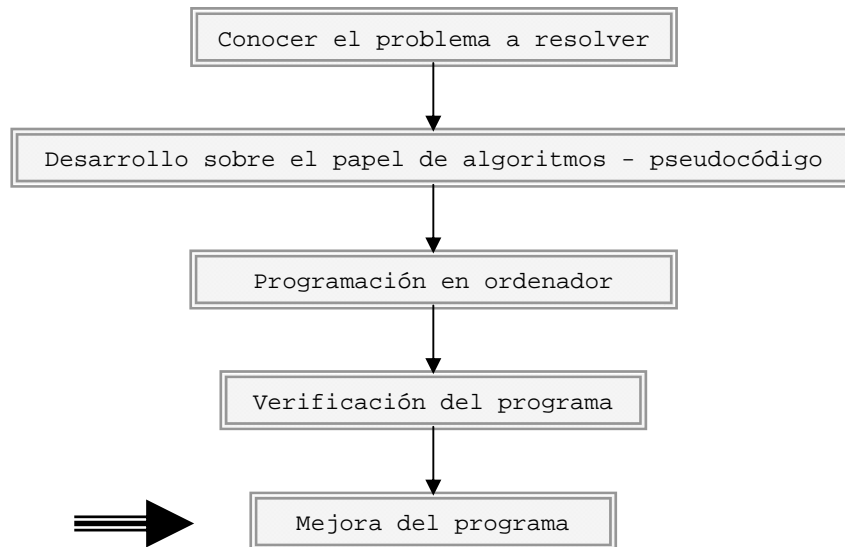
Autor: Mario R. Rancel

Resumen: Entrega nº 46 del Curso Bases de la programación Nivel II

24

## MEJORA DE PROGRAMAS. DOCUMENTACIÓN Y MANTENIMIENTO. INTRODUCCIÓN.

Vamos a concluir nuestro pequeño recorrido teórico – práctico por el arte de la programación por un lado hablando de mejora de programas y por otro con una serie de ejercicios de construcción de pseudocódigo. La mejora de programas se corresponde con la parte final de nuestro guión:



Se podrá observar que nos hemos “saltado” el paso de programación en ordenador. La finalidad es terminar de exponer los fundamentos genéricos de la programación que queríamos tratar, con el fin de que si algún lector desea utilizar otro lenguaje distinto al *Visual Basic* pueda dar por terminada aquí su utilización del libro. Los motivos para ello pueden ser varios como “Tengo las bases del *TurboPascal* que aprendí en el instituto” o “En mi empresa programan en *Java*” o simplemente “Quiero aprender este lenguaje tan moderno recién salido y que están dando un curso por fascículos en la revista que compro”.

Por tanto, etapa final para aquellos que se decidan por otro lenguaje y previa para los que se animen con el *Visual Basic*. En cualquier caso, puede resultar un poco abstracto lo que aquí se exponga. Por ello se hablará en términos muy genéricos cuyo sentido último se hará más patente a medida que avancemos en nuestra experiencia de programadores.

Hay una frase mágica entre los programadores. Esa frase es: ¡Ya lo tengo! Esta expresión puede suponer un antes y un después en el proceso de programación. Veamos por qué.

Hemos hablado de programar como “resolver problemas” y como un reto. Para un aprendiz, cualquier programa constituye un reto pues es la primera vez que se enfrenta a él. Para un programador experto, el trabajo de programación puede dividirse en una parte mecánica y en una parte creativa. La componente mecánica es aquella asociada a procesos muy conocidos y con alto índice de repetición como puede ser una lectura de datos, una petición de datos al usuario, un bucle, etc. y la creativa la asociada al diseño y estructuración de un programa. El peso de la mecánica y la creatividad es muy variable en función del cometido al que nos enfrentamos: hay trabajos eminentemente mecánicos, que requieren poco estudio y escasas ideas propias, frente a otros que pueden ser de naturaleza eminentemente creativa.

En un mismo equipo de trabajo, puede existir un programador de rango mecánico, por ejemplo llamémoslo programador junior, y otro de trabajo eminentemente creativo, llamémosle analista o coordinador. Cada cual en su nivel se enfrenta a pequeños o grandes problemas que se convierten en retos. Muchas veces ese reto tiene por objetivo inicial que el programa o algoritmo funcione, sin tener en cuenta la captura de datos, el desarrollo de opciones para el usuario (a través de menús), la estética de presentación en pantalla, etc. En definitiva todo aquello que constituye el esqueleto del programa.

Cuando el programador, después de horas, días o semanas de dedicación a la resolución de un problema consigue un embrión de programa o algoritmo que funciona dice: ¡Ya lo tengo! Esto viene a significar que se acabó el trabajo creativo y comienza el mecánico. Que el edificio ya tiene cimientos para soportar el peso, que se acabó el ascenso a la montaña y comienza el descenso o que la cabeza deja de echar humo. Pueden buscarse decenas de símiles para reflejar una sensación de satisfacción derivada de haber resuelto el problema, aunque todavía quede mucho trabajo. Y a lo que falta lo hemos llamado “Mejora del programa”. Puede comprender una mejora funcional (desarrollo de opciones, captura y archivo de datos, etc.), estética (forma de presentación en pantalla) y aspectos “de remate” como la documentación y la mejora de la presentación del código y archivos.

El proceso de mejora comprenderá pues lo que cada programador o equipo de trabajo haya dejado atrás. Puede llegarse al “programa que funciona” con un desarrollo funcional elevado pero una estética muy pobre, o al revés, con una estética casi definitiva pero un desarrollo funcional pobre. Que sea de una forma u otra depende del estilo personal del programador o de las directrices que haya impuesto el analista – coordinador. No se pueden establecer unas normas claras al respecto más que las que la experiencia nos vaya dictando.

Hay que tener presente que el concepto de mejora puede tener distintos niveles de aplicación: desde una parte de un módulo hasta un módulo o programa. Por tanto, aunque genéricamente nos refiramos a “programa”, la mejora, documentación y mantenimiento también puede afrontarse de acuerdo a la máxima “divide y vencerás”.

**Próxima entrega: CU00248A**

**Acceso al curso completo** en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:  
[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=36&Itemid=60](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=36&Itemid=60)