

FUNDAMENTOS DE PROGRAMACION

Fuente:

<https://www.areatecnologia.com/TUTORIALES/FUNDAMENTOS%20DE%20PROGRAMACION.htm>

Introducción

Los fundamentos de programación son las bases **comunes** a todos los programas. Es lo primero que tendrás que aprender incluso antes de elegir el programa con el que quieres programar.

Lo primero que tienes que saber es que el ordenador es una máquina eléctrica y solo entiende el llamado código binario (1 y 0).

1 = hay corriente

0 = No hay corriente

Este es su lenguaje.

Entendernos con él mediante este código es muy difícil, por eso los lenguajes de programación se dividen en dos tipos diferentes dependiendo de su cercanía al lenguaje del ordenador. El código del ordenador se basa en asignar a cada carácter (letra, signo, número, etc.) una combinación de 8 ceros y unos (8 bits = byte) mediante un código que se llama ASCII.

Por ejemplo la letra A se representa con la combinación siguiente: 01100001.

Los lenguajes más cercanos al idioma del ordenador, llamados de bajo nivel, son muy complicados (casi como el código del ordenador) y poco usados. El más conocido es el código o lenguaje máquina, un código que el ordenador puede interpretar directamente. Aquí tienes un ejemplo:

```
8B542408 83FA0077 06B80000 0000C383
```

De este tipo de lenguajes, que solo suelen utilizar los que programan los ordenadores para su uso, no vamos hablar aquí. Hablaremos de los conocimientos comunes a los lenguajes de alto nivel.

Los **lenguajes de programación de alto nivel** permiten dar órdenes al ordenador con un lenguaje parecido al nuestro (Visual Basic, Pascal, Logo, C++, JavaScript, etc.) y siempre o casi siempre en inglés. Hay programas de alto nivel como el GML o el Java que son programas interpretados, es decir, se analizan y ejecutan las instrucciones por el propio programa directamente. Otros necesitan un **compilador**, pero eso no es un problema, solo es un programa (**software**) que **se encarga de traducir el programa hecho en lenguaje de programación al código del ordenador** para que lo entienda. Con un tipo u otro es igual, lo importante es que los lenguajes, como todo, hay que aprenderse los, pero tienen una ventaja, y es que tienen muchos puntos en común.

Estos puntos son lo que vamos a estudiar aquí, los **fundamentos de programación** común a cualquier lenguaje de alto nivel.

Una vez aprendidos los fundamentos, tendrás que elegir el lenguaje que quieras usar, pero con estos conocimientos, todos te resultarán muy fáciles de aprender, solo tendrás que aprender unas cuantas instrucciones en inglés.

Programas y Algoritmos

Los lenguajes de programación, cuentan todos en su haber con un juego de "instrucciones". Una instrucción no es más que una orden que nosotros le damos a la máquina. Y es que, al fin y al cabo, **un programa no es más que una secuencia de instrucciones**

(escritas en algún lenguaje de programación) pensado para resolver algún tipo de problema.

Hay que tener claro que, si no sabemos resolver este problema, no podremos escribir el programa. Si no sabemos que es una suma, sería casi imposible hacer un programa para que nos sume dos números, a no ser que alguien nos ayudara. A ti se te puede ocurrir una manera de resolverlo, a tu compañero, otra, lo importante es que las dos formas de resolverlo lleven al mismo resultado, la suma.

La forma con el que resolvéis el problema, es lo que se llama **algoritmo**, y es lo que vamos a ver a continuación.

¿Qué es un Algoritmo?

Un algoritmo es una secuencia de PASOS a seguir para resolver un problema.

Por ejemplo, cuando quiero ver una película de vídeo, podría hacer los siguientes pasos (algoritmo):

1. Elijo una película de las de mi colección.
2. Compruebo SI TV y vídeo están conectados a la red (y procedo).
3. SI la TV está apagada, la enciendo, SI NO, pues no. Y lo mismo con el vídeo.
4. Introduzco la película en el vídeo. Dejo el estuche sobre el vídeo.
5. SI la TV no está en el canal adecuado, la cambio, SI NO, pues no.
6. Cojo los mandos a distancia (el del TV y el del vídeo).
7. Me pongo cómodo.
8. Pulso PLAY en el mando del vídeo.

Fíjate bien en unos detalles que son fundamentales y que aparecen en este algoritmo:

- La descripción de cada paso no me lleva a ambigüedades, los **pasos** son absolutamente **explícitos y no inducen a error**.
- **El número de pasos es finito**. Tienen un principio y un fin. Según lo visto, una mejor definición de algoritmo sería: **“Un algoritmo es una sucesión finita de pasos (no instrucciones como en los programas) no ambiguos que se pueden llevar a cabo en un tiempo finito.”**
- Este "lenguaje" el algoritmo está escrito en nuestro idioma, pero ahora necesitamos acercarnos a un poco más al lenguaje del ordenador.

Pero **el primer paso para realizar un programa es sacar su algoritmo**, es como explicar lo que queremos que haga nuestro programa. Ahora entiendes por qué decíamos antes que, si no sabemos nosotros resolver el problema, no podríamos crear el programa. No seríamos capaces de hacer su algoritmo.

Ahora que ya tenemos el algoritmo, para el siguiente paso se puede utilizar dos formas: Sacar el Diagrama de Flujo del algoritmo o su pseudocódigo. Algunos programadores hacen los dos.

Diagrama de Flujo

Un diagrama de flujo es una representación gráfica del algoritmo. Expresamos los pasos del algoritmo mediante un esquema con unos símbolos establecidos.

Un diagrama de flujo debe proporcionar una información clara, ordenada y concisa de todos los pasos a seguir.

Veamos un ejemplo:

Un algoritmo para cocinar un huevo para otra persona sería:

- Pregunto si quiere el huevo frito.
- Si me dice que sí, lo frío, si me dice que no, lo hago hervido.

- Una vez cocinado le pregunto si quiere sal en el huevo.
- Si me dice que no lo sirvo en el Plato. Si me dice que si le hecho sal y después lo sirvo en el plato.

Ahora que ya sabemos todos los pasos, mediante el algoritmo, podemos hacer un esquema con estos pasos a seguir.

Este esquema será el Diagrama de Flujo.



Si uno tiene experiencia puede prescindir del algoritmo escrito, pero siempre tendremos que tenerlo en mente para hacer el diagrama de flujo sin equivocarnos. Como nosotros nos centraremos más en el pseudocódigo, no hablaremos más de los diagramas de flujo.

El Pseudocódigo

El pseudocódigo es una forma de escribir los pasos, pero de la forma más cercana al lenguaje de programación que vamos a utilizar, es como un falso lenguaje, pero en nuestro idioma, en el lenguaje humano.

Una de las mayores dificultades con las que se encuentran los hispanoparlantes que empiezan a programar es el idioma. Por eso es bueno utilizar el pseudocódigo, algo así como un falso lenguaje de programación en español, que ayuda a asimilar con más facilidad las ideas básicas.

Por ejemplo si queremos escribir algo en pantalla, en pseudocódigo podríamos poner:

Escribir "Hola" o Escribir 20+30.

También podemos usar:

mostrar por pantalla "Hola"

Realmente el pseudocódigo lo podríamos escribir como nosotros quisiéramos, ya que realmente no es el programa en sí, solo es una ayuda para posteriormente realizar el programa mediante el lenguaje de programación que utilicemos, eso sí, es de gran ayuda, tanto que es imprescindible. Pero aunque lo podamos escribir de cualquier forma, la mayoría de los programadores suelen usar un vocabulario en común.

Este vocabulario será el que veamos aquí.

Recuerda que el pseudocódigo para un programador es fundamental.

Si sabes hacer el pseudocódigo del programa, pasarlo a cualquier lenguaje de programación es muy sencillo, solo tendrás que aprender los comandos equivalentes a las instrucciones en pseudocódigo.

Después de leer este documento, conteste las preguntas y realice las siguientes actividades, las cuales pueden ser realizadas en grupos de hasta 3 personas.

1. ¿Cuál es la importancia del pensamiento lógico secuencial en las ciencias de la computación y la informática?

2. ¿Qué es un algoritmo?, ¿Cuáles son las partes de un algoritmo?

3. ¿Qué tipos de algoritmos existen?, investigue.

4. La suma secuencial de los números del 0 al 100, ¿se podría transformar en un algoritmo cuantitativo?, justifique su respuesta.

5. En una hoja de cuaderno, por ambos lados, realice un algoritmo para las siguientes acciones
 - a. Sacar dinero de un cajero automático
 - b. Calcular el promedio de 3 calificaciones