

2º de Bachillerato



Tecnologías de la Información y Comunicación

Contenidos

Conceptos básicos de programación:
Iniciándonos en la programación

Encontramos numerosos ejemplos que ponen de manifiesto la potencia de los ordenadores, sobre todo para realizar tareas que implican gran cantidad de cálculos matemáticos.

La rapidez y eficiencia con que, por ejemplo, devuelve los resultados una simple calculadora nos hace reflexionar sobre **dos cuestiones**:

1. *¿Cómo puede trabajar tan rápido?*
2. *¿Nunca se equivoca en sus resultados?*



Imagen en pixabay de [animatedheaven](#) bajo licencia [CC](#)

Si intentamos dar respuesta a las dos preguntas planteadas conoceremos el funcionamiento interno de la máquina, es más, podremos modificar algunos aspectos para realizar mejoras o incluso añadir funciones nuevas según nuestras necesidades. Dichas respuestas son clave para entender los fundamentos de la **programación** de ordenadores.

Nuestro viaje en esta asignatura será apasionante, divertido y sobre todo práctico. En este primer tema, **te presentamos los conceptos y herramientas fundamentales** para iniciarte en el mundo de la programación.

¡Adelante!

1. ¿Qué vas a aprender en este tema?



Al finalizar el tema:

- Aprenderás a afrontar la **resolución** de un problema.
- Tendrás una idea básica de cómo **diseñar** un programa informático.
- Conocerás el software necesario para **construir** programas.
- Descubrirás cómo podemos **comunicarnos** con la máquina.

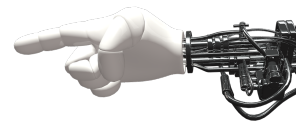


Imagen en pixabay de [DirtyOpi](#)
bajo licencia [CC](#)

En líneas generales, intentaremos prepararte para pensar como lo hace cualquier programador, que para enfrentarse a un problema, diseña una posible solución con sus dos **principales herramientas**:

1. Un ordenador
2. Una metodología de programación.

A modo de **ejemplo** se detalla el siguiente, que corresponde a resolver el problema de **sumar dos números, suponiendo que nuestra máquina no es capaz de realizar sumas cuyos sumandos sean mayores que 100**. Este puede ser un caso típico de problema que puede tener una calculadora al sumar números grandes.

Debemos, en primer lugar, **buscar una alternativa** a la suma tradicional, para ello usamos el concepto básico de descomposición numérica, esto es, un número por ejemplo de tres cifras se compone de centenas, decenas y unidades. En la siguiente tabla se detalla el proceso según la línea propuesta:

Sumar los números 342, 217

Sumandos	Suma de centenas			Decenas	Unidades	Resultado
342	100	100	100	40	2	$100 + 100 + 100 + 40 + 2 = 342$
217	100	100	-	10	7	$100 + 100 + 10 + 7 = 217$

2. Objetivos de la programación



Los principales **objetivos** propuestos serán:

- Conocer las principales características de los diferentes **lenguajes** de programación.
- Aprender las **fases** o etapas que debemos seguir para elaborar un programa informático.
- Valorar la importancia de realizar un buen **análisis** previo al desarrollo de un programa informático.
- Conocer y manejar **herramientas** para construir y ejecutar programas informáticos.
- **Depurar** un programa para minimizar riesgos de fallos y hacerlo más eficiente.
- Valorar la importancia del concepto de **seguridad** informática en la práctica.

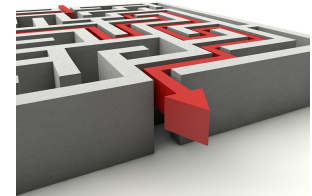


Imagen en pixabay de [Peggy-Marco](#)
bajo licencia [CC](#)

Como hemos comentado, aunque se expondrán de forma rigurosa los contenidos necesarios usados en programación, el enfoque de esta asignatura será eminentemente práctico. Es inútil pensar que conseguiremos realizar programas informáticos de calidad (rápidos, eficientes y estructuralmente correctos) sin tener en cuenta las tres reglas fundamentales que todo programador, con cierta experiencia, ha tenido que llevar a cabo en su proceso de aprendizaje: *practicar, practicar y practicar*.

La verdadera *prueba de fuego* será **la puesta en práctica** de los conceptos y metodologías que iremos aprendiendo a lo largo del curso. Ello **nos permitirá**:

- **Adelantarnos** a posibles fallos antes de la ejecución de un programa.
- **Estructurar** el código de forma que pueda entenderse según un orden y sintaxis lógica.
- Y sobre todo: **aprender de los errores** para no cometerlos en el futuro.

3. Un ejemplo real ¿Qué es programar?



¡Por fin tendré mi casa!

La mayoría de nosotros pensamos, en un primer momento, que una vez conseguido el crédito que necesitamos para adquirir nuestra propia casa, podemos mudarnos y cumplir sin más nuestro deseo.

Hacer la compra semanal, afrontar el pago de recibos relativos al consumo eléctrico y agua, etc, requieren de una organización extra por nuestra parte. Así deberemos tener en cuenta, además de la casa en sí, la gestión de la misma que será clave para:

- **Gestionar** de forma racional los **recursos económicos** con los que contamos.
- **Elaborar un plan** de acción a llevar a la práctica.
- **Ejecutar el plan** fijado.
- **Comprobar** que se han cumplido **los objetivos** iniciales propuestos.

La situación anterior, es un ejemplo real que podemos trasladar al campo de la programación, y que puede aclarar muchos aspectos que un programador debe tener en cuenta.

¿Qué es programar?

Programar **es diseñar, codificar, depurar y mantener el código fuente** de programas informáticos con el objetivo de conseguir que un ordenador dé solución a un problema planteado.

Disponiendo sólo del ordenador, contamos con una herramienta de gran capacidad de cálculo, como de una enorme excavadora diseñada para ayudar en el proceso de construcción de nuestra casa. Sin embargo, es ahí donde entra en juego el factor humano, en este caso **el programador**, que análogamente al conductor de la excavadora dispone de los conocimientos, técnicas y experiencia necesarias **para conseguir que la máquina realice la tarea** para la que ha sido diseñada.



Imagen en pixabay de [geralt](#)
bajo licencia [CC](#)

4. Planteamiento del problema



¡Empecemos!

En la línea de nuestra gestión económica, supongamos que queremos comparar precios de distintos productos o servicios. Ello nos lleva a **diseñar un programa informático que nos permita saber el mayor de dos números** enteros introducidos por teclado. En este supuesto hay mucha información que como programadores debemos extraer para elaborar nuestro programa. De una **forma ordenada**:

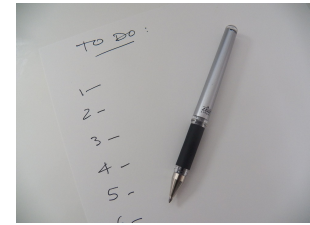


Imagen en pixabay de [geralt](#)
bajo licencia [Cc](#)

1. Los **datos de entrada** son dos números enteros (sin decimales) del tipo 1, 2, 3,...
2. Ambos números serán entrados por el usuario por teclado, así debemos **preguntar** en primer lugar cuál será el primer número y posteriormente cuál será el segundo.
3. Una vez introducidos los números, el ordenador (sistema o programa) deberá **compararlos** con el fin de determinar el mayor.
4. Por último, se **mostrará el resultado** correspondiente al mayor número de los dos entrados.

Aunque hemos diseñado nuestro plan de acción en cuatro pasos que a priori parecen dar respuesta al problema planteado, es recomendable revisarlo nuevamente pues, concretamente, el paso 3 tal y como está planteado no contempla el caso de que los dos números sean iguales.

Por tanto, habrá veces en las que será preciso **desglosar** uno o varios **pasos principales en pasos intermedios más sencillos** que en conjunto darán respuesta a sus correspondientes pasos principales. Esta idea se conoce en programación como "*divide y vencerás*" y se explicará detalladamente más adelante.

5. El lenguaje de programación



Siguiendo con el problema planteado en la sección anterior, en este momento disponemos de nuestro plan de acción que deberemos poner en práctica, pero ¿cómo? Necesitamos comunicarnos con el ordenador para:

1. **Transmitir** cada paso de nuestro plan de acción.
2. **Obtener** los resultados esperados.

El idioma que usaremos para tal fin es el lenguaje de programación que, como es lógico, debe ser entendido por ambas partes (el programador y la máquina).

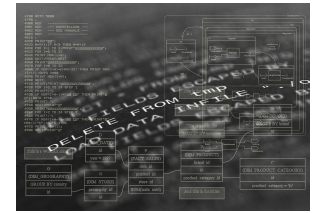


Imagen en pixabay de [geralt](#) bajo licencia [CC](#)

Así, **un lenguaje de programación es un conjunto de símbolos y caracteres combinados entre sí, de acuerdo con una sintaxis ya definida y respetando unas reglas establecidas, para posibilitar la comunicación con la CPU del ordenador.**

En la práctica, normalmente se usarán, por un lado, lenguajes que emulen la forma de pensar del programador y, por otro, el denominado lenguaje máquina que es el único que entiende el ordenador. En este proceso será necesario una conversión o transformación de cada paso o instrucción escrito por el programador a dicho lenguaje máquina.

Curiosidad

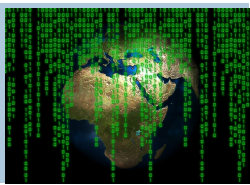


Imagen en pixabay de [geralt](#) bajo licencia [CC](#)

Curiosidad 1:

Los **primeros lenguajes de programación** preceden a la computadora moderna. En un inicio los lenguajes eran **códigos**.

En la década de 1940 fueron creadas las **primeras computadoras modernas, con alimentación eléctrica**. La velocidad y capacidad de memoria limitadas forzaron a los programadores a escribir programas, en lenguaje ensamblador muy afinados. Finalmente se dieron cuenta de que la

programación en lenguaje ensamblador requería de un gran esfuerzo intelectual y era muy propensa a errores.

El rápido crecimiento de **Internet** en la década de 1990 fue un gran acontecimiento histórico para los lenguajes de programación. Con la apertura de una plataforma totalmente nueva para los sistemas informáticos, Internet creó una oportunidad para adoptar nuevos lenguajes. En particular, el lenguaje de programación Java se hizo popular debido a su pronta integración con el navegador web Netscape Navigator, y varios lenguajes de scripting alcanzaron un amplio uso en el desarrollo de aplicaciones personalizadas para servidores web. (Texto en [Wikipedia](#)).

Curiosidad 2:

¿Cuántos lenguajes de programación crees que existen? 10? 30? 50? Que te parece **1150 lenguajes diferentes de programación**. En la pagina <http://www.99-bottles-of-beer.net/> muestran 1150 lenguajes diferentes haciendo una sola cosa: Un programa que imprime la canción "99 botellas de cerveza".

Hay algo nuevo que aprender.



5.2. Resultado final



Como futuros programadores, y **dependiendo del objetivo y uso** de nuestro programa, el resultado final será distinto.

- Si queremos construir una aplicación web que se ejecute del lado del servidor probablemente usemos PHP, Java, etc.
- Si de otra forma nuestra aplicación está pensada para consultar y mostrar información de una base de datos quizás SQL será nuestra opción.
- Puede que queramos sólo resolver determinados problemas matemáticos, aquí R o Python pueden ser una opción.



Imagen en pixabay de [kropekk_pl](#) bajo licencia CC

Además, debemos tener en cuenta también **aspectos relativos a dónde y cómo se mostrará** el resultado, como por ejemplo:

- tipo de pantalla (formato ancho, cuadrado, etc).
- resolución de pantalla (calidad de imágenes, ventanas de texto, etc).
- memoria del ordenador (capaz de soportar la carga de procesamiento, etc).
- usuario final al que va dirigido (invidentes, sordos, profesional especializado, etc).
- Temática y estilo de redacción (formal, coloquial, etc).

Curiosidad

¿Sabes cuál es el lenguaje de programación más difícil que existe? Error si contestaste "Perl". **El lenguaje de programación más difícil se llama Malbolge.** Entre las peculiaridades de este lenguaje están:

1. En el 98, su programador lo lanzó al dominio público.
2. Desde su aparición, se tardó más de 2 años en aparecer su primer programador.
3. ¿Por qué digo "aparición"? Malbolge es un lenguaje de programación generado por un programa escrito en LISP, fue pues, generado por una máquina.
4. Cada instrucción es encriptada (le resta 33, le suma el puntero y le realiza mod 94) después de que se ejecuta. Por lo que cada instrucción es diferente cada vez que se ejecuta. Es altamente impredecible.
5. ¿Crees que es exagerado? Esta sería la versión de *hello world* de Malbolge.

```
1: (<=:9876Z4321UT.-Q+*)M'&%$H"!~}|Bzy?={z]KwZY44Eq0/{mlk**
2: hKs_dG5[m_BA{?-Y;;Vb'rR5431M}/.zHGwEDCBA@98\6543W10/.R,+O<
```

6. Repasamos



Te aconsejamos realizar un pequeño **questionario**, con el fin de asegurarte que realmente has comprendido los principales conceptos del tema.

Comprueba lo aprendido

El principal objetivo de la programación es

- resolver problemas con ayuda del ordenador
[-----]
- corregir los errores que comete el programador
[-----]
- tiene gran capacidad para realizar cálculos matemáticos
[-----]

Mostrar retroalimentación

Solution

1. Correcto
2. Incorrecto
3. Incorrecto

¿Qué es programar?

- Elaborar un programa informático para solucionar un problema
[-----]
- Ayudar al programador a detectar errores de cálculo
[-----]
- Organizar la información con que cuenta el programador
[-----]

Mostrar retroalimentación

Solution

1. Correcto
2. Incorrecto
3. Incorrecto

Para construir un programa normalmente habrá que

- obtener antes el resultado para comprobar que nuestro programa es eficiente
[-----]
- detallar, sin importar el orden, los pasos a seguir
[-----]

Mostrar retroalimentación

Solution

- 1. Correcto
- 2. Incorrecto
- 3. Incorrecto

El lenguaje de programación

- tiene una sintaxis propia pensada para comunicarse con la máquina
[-----]
- es el idioma elegido por los informáticos para administrar un ordenador
[-----]
- no es necesario si sólo queremos escribir instrucciones
[-----]

Mostrar retroalimentación

Solution

- 1. Correcto
- 2. Incorrecto
- 3. Incorrecto

Referente al editor

- ayuda al programador a construir programas ofreciendo ventajas al respecto
[-----]
- es la persona encargada de construir programas informáticos
[-----]
- tiene gran capacidad para realizar cálculos matemáticos
[-----]

Mostrar retroalimentación

2. Incorrecto

3. Incorrecto

7. ¿Te atreves?



La programación es un campo de la rama de la Informática que actualmente está muy valorado. Las ofertas de trabajo son numerosas para los profesionales con un perfil de estas características.

Además, aprender a programar, o de dicho de una manera más técnica aprender **Ciencias de la Computación**, puede ser una herramienta muy valiosa para cualquier persona que deba enfrentarse continuamente a problemas de diversa índole.

En esta sección te invitamos a que abras tu campo de posibilidades, conociendo más sobre esta disciplina y sobre todo poniéndote a prueba a tí mismo.

Nuestra primera propuesta te presenta **La Hora del Código**, una puerta abierta y en continuo movimiento para que conozcas más detalles, a nivel internacional, sobre el mundo de la programación.



Imagen en pixabay de [OpenClipart-Vectors](#) bajo licencia CC

La **Hora del Código** es un movimiento global, que llega a decenas de millones de estudiantes en más de 180 países. Cualquier persona, en cualquier lugar del mundo puede organizar una Hora del Código. Tutoriales, de una hora de duración, están disponibles en más de 30 idiomas. No se necesita experiencia. **Para edades entre 4 y 104 años.**

Cualquier persona puede participar, para ello simplemente hay que registrarse. Además sólo por participar se puede ganar un premio.

Para aclarar ciertas dudas, a continuación te especificamos algunas de las **preguntas frecuentes** publicadas en dicha plataforma:



Imagen en La Hora del Código de <https://hourofcode.com/es> bajo licencia CC

● ¿Qué es la Hora de Programación o del Código?

La Hora del Código es una introducción de una hora de duración a las Ciencias de la Computación, diseñada para mostrar que todo el mundo puede aprender a programar y así comprender los fundamentos básicos de la disciplina.

● ¿Cuándo es la Hora del Código?

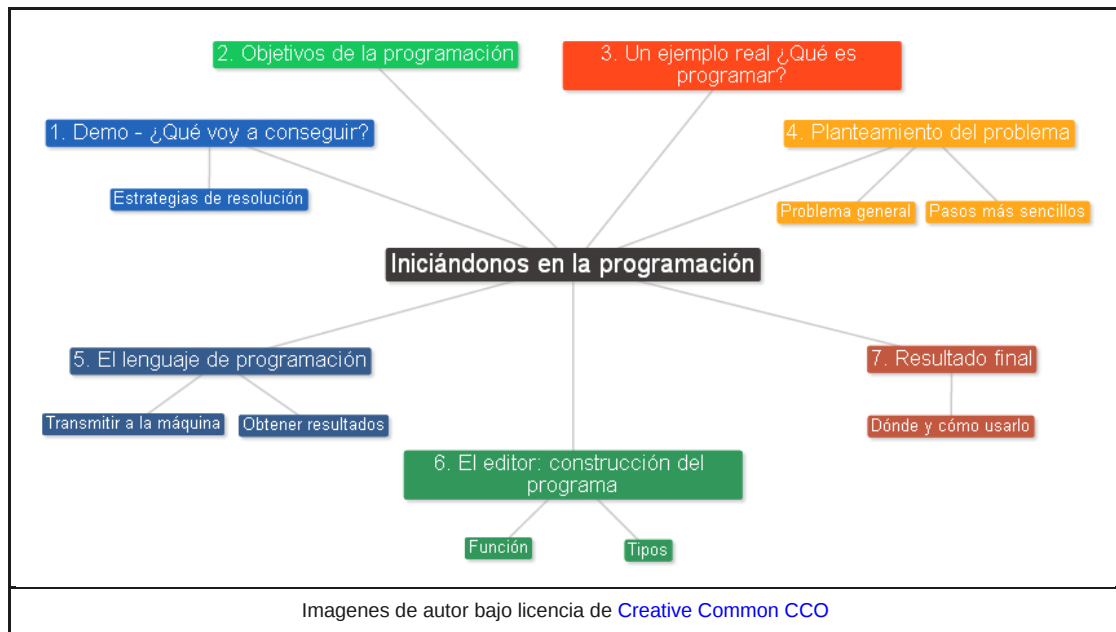
Cualquiera puede organizar "Una Hora del Código" en cualquier momento, pero el objetivo fundamental de esta campaña es que decenas de millones de estudiantes prueben "La Hora del Código" en celebración de la **Semana Educativa de las Ciencias de la Computación**. ¿Se trata de una hora específica? No. Puedes celebrar "La Hora del Código" en cualquier momento de esa semana. (Y si no puedes hacerlo durante dicha semana hazlo una antes o una después).

● ¿Por qué Ciencias de la Computación?

Todos los estudiantes deben tener la oportunidad de aprender Ciencias de la Computación. Ayuda a fomentar las habilidades de resolución de problemas, la lógica y la creatividad. Introducir a los estudiantes de forma temprana en las ciencias de la computación, les permitirá disponer de conocimientos imprescindibles en el s.XXI; fundamentales sea cual sea su futuro desarrollo profesional. Ver más estadísticas en [Code.org](#).

● ¿Quién está detrás de la Hora del Código?

La Hora del Código está organizada por [Code.org](#), una organización pública sin fines de lucro (501c3) dedicada a promover la participación en escuelas e institutos de las Ciencias de la Computación, con una especial atención en incrementar el número de mujeres y estudiantes de colectivos minoritarios que aprenden a programar. Una **coalición sin precedentes de socios** que se ha reunido para apoyar la Hora del Código, también — incluyendo Microsoft, Apple, Amazon, Boys y Girls Clubs of America y College Board.



Descarga archivo imprimible.

Aviso Legal

El presente texto (en adelante, el "**Aviso Legal**") regula el acceso y el uso de los contenidos desde los que se enlaza. La utilización de estos contenidos atribuye la condición de usuario del mismo (en adelante, el "**Usuario**") e implica la aceptación plena y sin reservas de todas y cada una de las disposiciones incluidas en este Aviso Legal publicado en el momento de acceso al sitio web. Tal y como se explica más adelante, la autoría de estos materiales corresponde a un trabajo de la **Comunidad Autónoma Andaluza, Consejería de Educación y Deporte (en adelante Consejería de Educación y Deporte)**.

Con el fin de mejorar las prestaciones de los contenidos ofrecidos, la Consejería de Educación y Deporte se reserva el derecho, en cualquier momento, de forma unilateral y sin previa notificación al usuario, a modificar, ampliar o suspender temporalmente la presentación, configuración especificaciones técnicas y servicios del sitio web que da soporte a los contenidos educativos objeto del presente Aviso Legal. En consecuencia, se recomienda al Usuario que lea atentamente el presente Aviso Legal en el momento que acceda al referido sitio web, ya que dicho Aviso puede ser modificado en cualquier momento, de conformidad con lo expuesto anteriormente.

Régimen de Propiedad Intelectual e Industrial sobre los contenidos del sitio web.

Imagen corporativa. Todas las marcas, logotipos o signos distintivos de cualquier clase relacionados con la imagen corporativa de la Consejería de Educación y Deporte que ofrece e