

INTRODUCCIÓN A LOS LENGUAJES DE PROGRAMACIÓN (WIN-LOGO)

Se define como **lenguaje de programación** a un conjunto de símbolos y reglas que sirven para realizar programas de ordenador.

Un **programa** es un conjunto de instrucciones, expresadas de manera algorítmica (de manera que estén encadenadas) que deben ser entendidas y ejecutadas por un ordenador.

Los elementos de los elementos de los lenguajes de programación son:

- **Léxico**, conjunto de símbolos y palabras clave. Cada lenguaje tiene su propio léxico.
- **Sintaxis**: reglas que permiten construir oraciones con sentido, empleando el léxico específico del lenguaje.

DISTINTOS LENGUAJES DE PROGRAMACIÓN

Existen multitud de lenguajes de programación para realizar distintos tipos de programas, algunos de los más utilizados son:

Lenguaje máquina, son las instrucciones que realmente entiende el microprocesador de un ordenador, son instrucciones muy sencillas, pero resulta muy complicado realizar programas con dicho lenguaje.

Basic, Pascal, Cobol, Fortran, C. Lenguajes utilizados desde hace tiempo para realizar muchos tipos de programas. Las versiones más modernas y empleadas de algunos de ellos son **Visual Basic** y **Visual C.**

SQL, lenguaje empleado para manejar bases de datos.

HTML, lenguaje en el que se codifica una página Web.

Lenguajes Script, son lenguajes que pueden incrustarse en el código de una página Web. Algunos de los más empleados son **Java-script, ASP, PHP.**

Win-logo, lenguaje diseñado para la educación, nosotros lo emplearemos para programar la controladora con la que haremos funcionar nuestros proyectos.

METODOLOGÍA DE PROGRAMACIÓN

A la hora de diseñar un programa, debemos seguir unos pasos desde que se plantea el problema a solucionar hasta que escribimos el programa en un lenguaje de programación, debemos de analizar el problema utilizando desde el LENGUAJE NATURAL hasta tener un PROGRAMA (EN LENGUAJE QUE ENTIENDA EL ORDENADOR).

ALGORITMO: conjunto de instrucciones encadenadas, que podemos traducir directamente a instrucciones de cualquier lenguaje de programación para realizar un programa.

Para describir algoritmos se suelen emplear diagramas de flujo que más adelante se verán.

Veámoslo con un ejemplo:

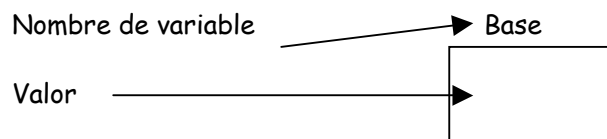
LENGUAJE NATURAL	Averiguar el área de un rectángulo.
LENGUAJE CIENTÍFICO	Area = Base x altura
ALGORITMO	Leer Base Leer Altura Hacer Area = Base x altura Escribir Area
PROGRAMA (en Win-logo)	

VARIABLES

Son nombres que representan valores, se emplean para almacenar temporalmente los datos de entrada, salida o datos auxiliares.

Hay que distinguir entre el nombre de la variable y el valor que tiene. El valor puede cambiar a lo largo del programa.

En las variables se pueden almacenar datos de diferentes tipos, numéricos, de texto etc.



CONDICIONES Y OPERADORES LÓGICOS

Condiciones son expresiones relacionales o lógicas empleadas en las instrucciones de los programas y que, al ser evaluadas, devuelven uno de los dos estados lógicos posibles: verdadero (1) o falso (0).

Operadores relacionales: =, <, >, <=, >= y <>.

Operadores lógicos: AND, OR, NOT.

Si tomamos a un alumno chico de 17 años indica si las siguientes condiciones son verdaderas o falsas:

Condición: Edad < 30 AND sexo = V

Condición: Edad >20 OR sexo = V

Condición: Edad <10 AND sexo = M

Condición: Edad = 18 OR sexo = M

Condición: NOT Edad = 17







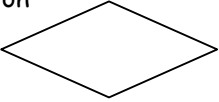


Condición: Edad <> 40

Condición: Edad >=10 AND Edad <= 18



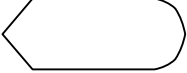
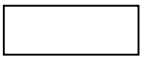

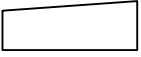
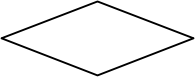
Condición: sexo = V AND Edad = 17

DIAGRAMAS DE FLUJO

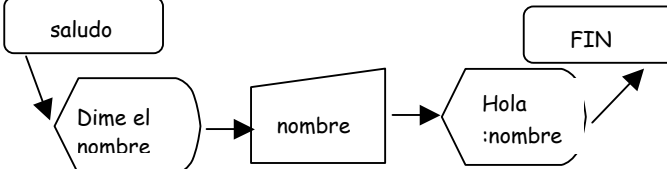
Es una herramienta gráfica que puede ayudar a establecer o analizar el algoritmo adecuado de un programa. Los símbolos más empleados son:

<p>Terminal (inicio, fin)</p> 	<p>Salida de información por pantalla</p> 
<p>Entrada manual</p> 	<p>Llamada a un proceso definido o subrutina</p> 
<p>Entrada o salida de datos</p> 	<p>Proceso</p> 
<p>Decisión</p> 	<p>y</p>  <p>o</p> 

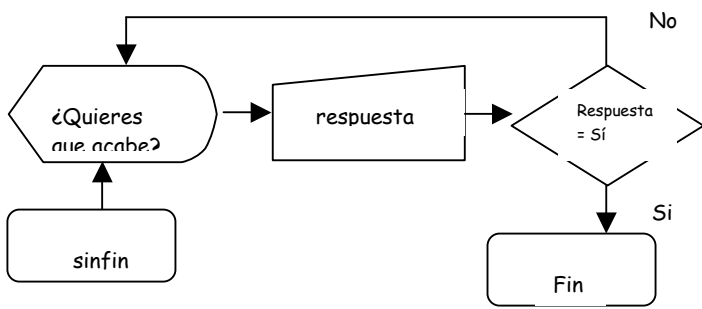
PRINCIPALES INSTRUCCIONES EN WIN-LOGO

<p>PARA nombre_del_programa</p> 	<p>Todo programa empieza con la línea PARA seguida del nombre del programa.</p> <p>El nombre del programa tiene que ser texto sin espacios. Se diferencian mayúsculas y minúsculas.</p>
<p>FIN</p> 	<p>La última línea de todo programa tiene que ser FIN, cuando pulsamos ENTER en esa línea el ordenador nos devuelve el mensaje "acabas de definir" y el nombre del programa.</p> <p>Si hacemos algún cambio en el programa debemos de ir a la línea FIN y pulsar ENTER.</p>
<p>ESCRIBE (ES)</p> <p>ES "Pepe"</p> <p>ES [Dime una frase]</p> <p>ES 45</p> <p>ES :nombre</p> 	<p>El ordenador escribe en la pantalla de trabajo lo que le indiquemos:</p> <p>Una palabra, una frase (entre corchetes), un número, o el valor de una variable, :más nombre de variable.</p> <p>Para escribir distintos datos mezclados la orden entre paréntesis:</p> <p>(ESCRIBE "Hola :nombre [Cómo estás?])</p>
<p>HAZ "nombre_de_variable Valor"</p> <p>HAZ "nombre LEEPALABRA"</p> <p>HAZ "contador 0"</p> 	<p>Asigna a una variable identificada por comillas seguido de un nombre, un valor. El valor puede ser un número, una palabra o un valor tomado del teclado (LEEPALABRA)</p>
<p>LEEPALABRA (LP)</p> 	<p>El ordenador captura el valor que le demos a través del teclado y pulsemos ENTER.</p>
<p>LEECAR (LC)</p> 	<p>El ordenador captura una sola pulsación de tecla sin esperar a pulsar ENTER</p>
<p>SI Condición [órdenes 1] [órdenes 2]</p> <p>Ejemplos de Condiciones:</p> <p>:nombre = "Pepe"</p> <p>:numero > 20</p> 	<p>El ordenador evalúa la Condición si es verdadera ejecuta el primer corchete, si es falsa ejecuta el segundo.</p> <p>El segundo corchete puede faltar.</p>
<p>BT</p> <p>ESPERA n</p>	<p>Borra pantalla de texto</p> <p>Espera n centésimas de segundo para seguir el programa.</p>

PROGRAMAS BÁSICOS

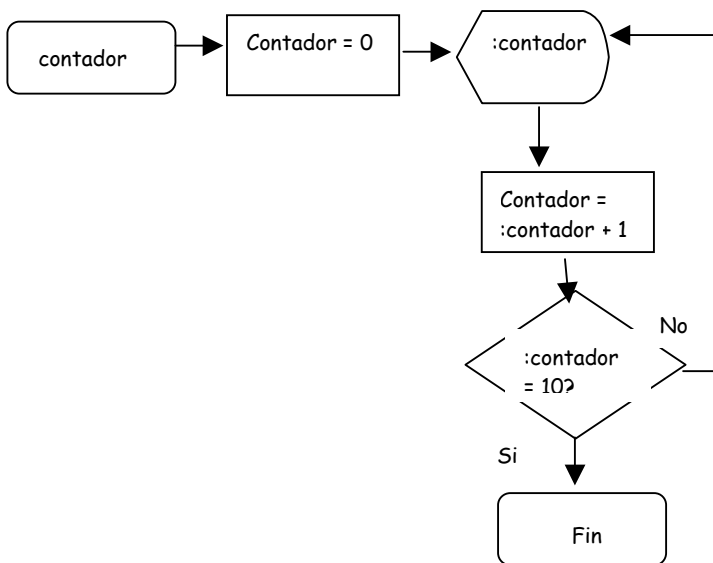
<p>Leer una variable por teclado y la escribe.</p> 	<p>PARA saludo</p> <p>ES [Dime tu nombre]</p> <p>HAZ "nombre LEEPALABRA"</p> <p>(ESCRIBE [Hola] :nombre)</p> <p>FIN</p>
--	---

Programa que finaliza si respondemos "si" por teclado.



PARA sinfín
 ES [¿Quieres salir?]
 HAZ "respuesta LEEPALABRA
 Si :respuesta = "si [ALTO] [sinfín]
 FIN

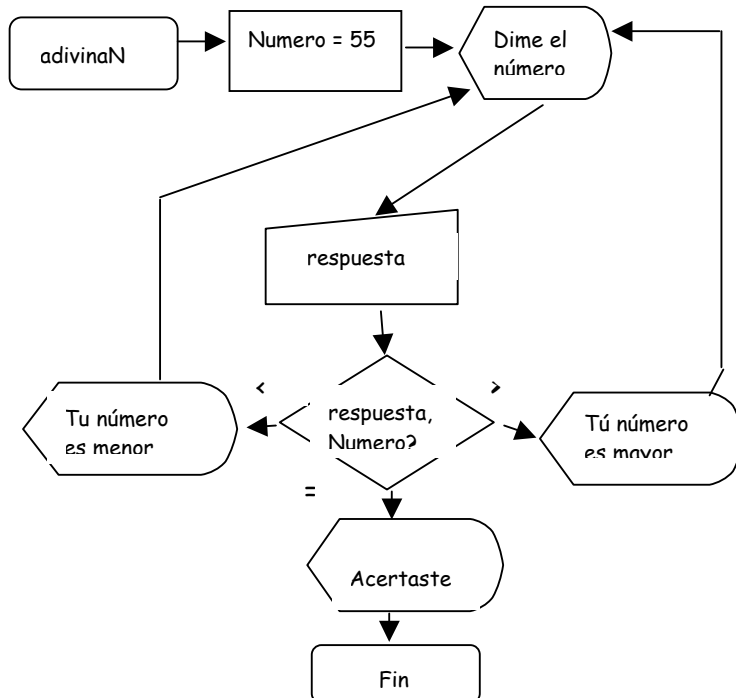
Contador hasta 10



Para realizar este programa hay que dividirlo en dos partes, contador1 y contador2.

PARA contador1
 HAZ "contador 0
 contador2
 FIN
 PARA contador2
 ES :contador
 HAZ "contador :contador + 1
 SI :contador = 10 [ALTO]
 contador2
 FIN

Adivina el número



En el programa introducimos un número (55) que hay que adivinar, si introducimos un número mayor o menor el ordenador nos lo dice y vuelve a preguntar:

PARA adivinaN
 HAZ "numero 55
 ES [Dime el número]
 HAZ "respuesta LEEPALABRA
 SI :respuesta < :numero [ES [tu número es menor] adivinaN]
 SI :respuesta > :numero [ES [tu número es mayor]adivinaN]
 SI :respuesta = :numero [ES [Acertaste]]
 FIN

Realizar el algoritmo y el diagrama de flujo de los siguientes programas.

1. El ordenador nos pregunta el nombre, se lo introducimos por el teclado y nos responde "Hola" más nuestro nombre.
2. El ordenador nos pregunta si queremos salir del programa, si respondemos sí se detiene si respondemos no el ordenador vuelve a preguntar.
3. Añadimos al ejercicio un contador, de manera que cada vez que nos pregunta si queremos salir nos informa de la vez que lo hace.
4. Dado un número natural, el ordenador nos dice si es par o impar.

5.- El ordenador nos pregunta el año que nacimos, nos dice la edad que tenemos y si cumplimos 18 años durante 2007 nos dice que este año podemos conducir coches, si somos menores nos responde "tendrás que esperar".

Realizar el diagrama de flujo y el programa.

6.- CONTADOR 0 A UN NÚMERO

1. El ordenador nos pide un número
2. el ordenador nos muestra los valores del 0 a dicho número, esperando 2 segundos entre cada valor.

7.- CUENTA ATRÁS.

1. El ordenador nos pide un número.
2. Nos va dando los valores desde ese número hasta cero, con espera de 2 segundos.

8.- ADIVINAR NÚMERO

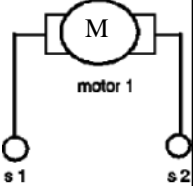
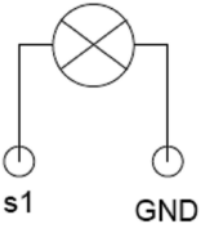
1. El ordenador nos pregunta por un número, (dicho número lo introducimos en el programa).
2. Le introducimos un número.
3. Si es mayor o menor, el ordenador nos lo dice y así hasta que acertemos.

9.- JUEGO DE PREGUNTAS Y RESPUESTAS

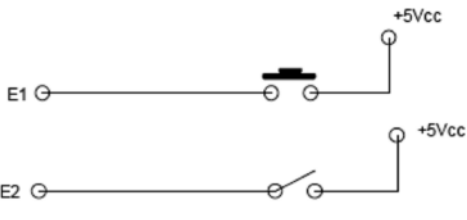
1. El ordenador nos hace tres preguntas (Ejemplo. ¿Cuál es la capital de España?)
2. Respondemos a cada una.
3. El ordenador nos dice el número de aciertos.

Instrucciones para el control de la controladora INVESTRÓNICA

Activación de salidas.

	<p>Conexión de un motor a una salida digital. Conectando un motor a las salidas digitales S1 y S2, quedará conectado a la posición 1. M1 "I Se activa una salida, el motor gira en un sentido. M1 "D Se activa la otra salida el motor gira en sentido contrario. M1 "P Se desactivan las dos salidas, el motor se para. Lo mismo para:</p> <p style="text-align: center;">M2 S3-S4 M3 S5-S6 M4 S7-S8</p>
	<p>Conexión de bombillas a salidas digitales. Como en las bombillas no es necesario tener en cuenta la polaridad, para ahorrar salidas digitales podemos conectar dichos actuadores a una sola salida digital y a (GND). La conexión de una bombilla a una sola salida digital y el conector GND de 0 voltios se realiza como indica el gráfico. De esta manera, por cada salida S1... S8, se puede conectar un actuador que no necesite control de polaridad</p> <p>SAL1 1 Activa la salida 1 SAL1 0 Desactiva la salida 1. Lo mismo con SAL2, SAL3, etc.</p>

Detección de entradas.

	<p>Como puede verse en el gráfico adjunto, la conexión de los sensores digitales a la controladora se realiza conectando uno de los terminales a una entrada digital (E1, E2,...) situada en el lateral derecho de la controladora y el otro terminal, a +5V, situado en el lateral izquierdo de la controladora.</p>
---	---

Para leer el valor de la entrada digital se emplea la instrucción SD número de entrada.

SD 1, SD 2, SD 3, etc.

Si escribimos la orden ES SD 1, el ordenador nos devuelve 1 ó 0 dependiendo de que la entrada 1 esté activado o no.

La instrucción anterior no nos sirve de mucho. Normalmente la instrucción de detección de entradas la emplearemos con la orden **SI**.

SI SD 1 = 1 [M1 "P] Si la entrada 1 está activada se detiene el motor M1

Realizar el diagrama de flujo del ejemplo anterior.