

UNIDAD DIDÁCTICA 12

Son autores de esta unidad didáctica:

Angel Sánchez Solanilla
Máximo Bolea Campo
Andrés Sánchez Otín

Coordinación pedagógica:

Carmen Candiotti López-Pujato

I.- INTRODUCCIÓN

I.- INTRODUCCIÓN

PRESENTACIÓN

Con esta unidad didáctica se inicia un segundo bloque de actividades, relacionadas con el mundo del control y la robótica. En esta unidad didáctica se realiza una introducción al mundo del control de las máquinas; para ello, se utilizará alguno de los proyectos realizados hasta el momento. También se diseñarán y construirán máquinas específicamente pensadas para ser controladas por ordenador.

Una de las pretensiones de esta unidad es, por una parte, introducir al profesorado en el uso del ordenador en la tecnología, y por otra, enseñarle a secuenciar los contenidos y las actividades de control en el aula.

Los distintos apartados de esta unidad y de las sucesivas tienen un marcado carácter cíclico, es decir, en cada uno de los distintos apartados, se tratarán aparentemente los mismos conceptos; sin embargo, el enfoque y la profundidad serán distintos. En la sección "Manos a la obra" se sientan las bases teóricas y los conceptos que el profesorado debe conocer para impartir esta disciplina. En "Entre máquinas y herramientas" se desarrollan, de una manera práctica, los procedimientos y métodos analizados en la sección anterior, y se amplían según las necesidades de cada momento. En la sección "Con nuestros alumnos y alumnas" se presentan y desmenuzan los aspectos didácticos a tener en cuenta en el aula; asimismo, se ejemplifican todas las situaciones presentadas anteriormente con explicaciones que se dan en el aula y que pueden reproducirse aplicándolas al quehacer cotidiano. Cuando sea necesario tratar algún aspecto fundamental con detalle, se incluye un anexo en el que se desarrolla con una profundidad suficiente para el objeto de este curso.

1.- OBJETIVOS

En esta unidad se pretende que el profesorado consiga los siguientes objetivos:

1º.- Analizar y descomponer un problema complejo en otros más simples, de manera que, después de resolver cada uno de ellos, quede resuelto el problema inicial.

2º.- Emplear la metodología adecuada para aplicar el objetivo indicado en el apartado anterior con sus alumnos.

3º.- Utilizar un lenguaje de programación, LOGO, para el desarrollo de los algoritmos de robótica y control.

4º- Conocer y usar la controladora, elemento de conexión entre la máquina y el ordenador.

5º.- Fomentar el trabajo en grupo de los alumnos, a través de debates internos en el grupo en los que el profesorado actúe como dinamizador e incitador de reflexiones que conduzcan a la resolución de los problemas planteados.

2.- CONTENIDOS

II.- FUNDAMENTOS DE LOGO

- 1.- Winlogo usa ventanas.
- 2.- El modo directo.
- 3.- Modo procedimental.
- 4.- Algunas primitivas más.
- 5.- Programación modular.
- 6.- Uso de variables.
- 7.- Viaje total.
- 8.- Recursividad.

III.- MANOS A LA OBRA

- 1.- Procesos para la mejora y el control de la producción.
- 2.- El ordenador en los procesos de robotización.
- 3.- Necesidad de un lenguaje de programación.
- 4.- Evolución y tipos de robot.
- 5.- Entradas y salidas.

IV.- CON NUESTROS ALUMNOS Y ALUMNAS

- 1.- Planteamiento interdisciplinar y globalizado.
- 2.- La recepción de información.
- 3.- El proceso de la información.
- 4.- Actuación.
- 5.- Transferencia de los conceptos propios del proceso de control al ámbito del trabajo grupal

V.- ENTRE MÁQUINAS Y HERRAMIENTAS

MICROMUNDOS DE CONTROL

- 1.- Control externo de una máquina mediante ordenador.
- 2.- La automatización y la robotización.

3.- Actividad: control de un motor. El problema de la interfaz.

4.- Logo.

VI.- GLOSARIO

VII.- BIBLIOGRAFÍA

VIII.- SOLUCIONES

3.- CONOCIMIENTOS PREVIOS

Esta es la primera unidad en la que se introduce el uso del ordenador, por lo que son necesarios unos sencillos conocimientos sobre su uso. En el caso de que no se posean, pueden adquirirse estudiando las explicaciones que aparecen en el apartado II, de Logo, y a través de consultas que se establezcan con la tutoría del curso.

II.- FUNDAMENTACIÓN CIENTÍFICO-TÉCNICA

II.- FUNDAMENTACIÓN CIENTÍFICO - TÉCNICA: FUNDAMENTOS DE LOGO

El objeto del presente apartado es dar una visión rápida y sucinta de cómo se estructura el lenguaje Logo, describir un método de trabajo que permita introducir al alumno en el uso de este entorno de programación y fundamentalmente dotar al profesorado de los mínimos conocimientos de Logo, si es que aún no los tiene, para poder utilizar este lenguaje en el área de tecnología y en especial en el control de máquinas; en lo que llamaremos el micromundo de control.



En este apartado, es necesario seguir el desarrollo trabajando con el ordenador para hacer las prácticas y actividades guiadas.

El lenguaje Logo se presenta en distintos formatos, que depende del fabricante del entorno; los más utilizados son ACTILOGO y WIN-LOGO fundamentalmente; el winlogo puede usarse en entorno del DOS y también en el de WINDOWS. Como últimamente la mayoría de los programas se desarrollan para el entorno Windows y el manejo de este sistema operativo es muy asequible para cualquier usuario, en el este curso se ha optado por hacer uso de WIN-LOGO para Windows.

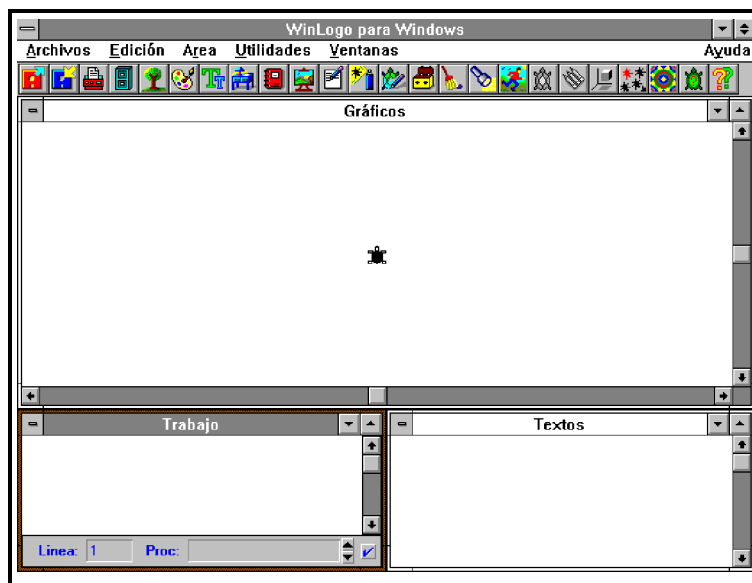
Para poder ejecutar el WIN-LOGO son necesarios, además de los materiales genéricos que requiere el entorno Windows sobre el que este lenguaje trabaja, los siguientes elementos: un ordenador compatible con IBM 386 o superior, dos Mb de memoria extendida y dos Mb de disco duro libres; opcionalmente serán necesarias una impresora y una tarjeta de sonido.

1.- WINLOGO USA VENTANAS

WinLogo posee varias ventanas o áreas de trabajo que pueden ser utilizadas para determinadas tareas. Estas ventanas son:

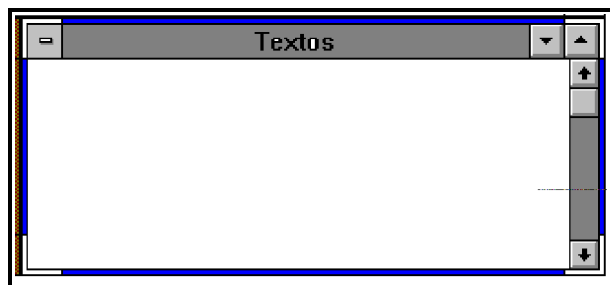
Gráficos	Textos
Trabajo	Formas
Edición	Trazado
Variables	

Cuando se inicia el programa se presentan las ventanas o áreas de trabajo más importantes, que son: la ventana de Gráficos (mundo de las tortugas), la ventana de Textos (zona en la que Logo escribe sus mensajes) y ventana correspondiente al área de Trabajo (zona en la que el usuario escribe sus mensajes).



Pantalla inicial de WIN-LOGO

Cómo trabajar con ventanas



Ventana de texto

Las ventanas no están siempre en la misma posición: se pueden mover, cambiar de tamaño, superponer o cerrar.

Solamente se puede actuar sobre las ventanas activas. Para determinar cuándo una ventana se encuentra activa es suficiente comprobar que su barra superior se encuentra resaltada.

Para activar una ventana basta colocar el puntero del ratón sobre ella y pulsar su botón izquierdo; el camino indicado sirve solamente si la ventana está presente en pantalla; en caso contrario, es decir cuando la ventana no esté en pantalla, basta elegir la opción Ventanas de la barra de menús y seleccionar el nombre de la ventana que se quiere activar; la ventana así activada se superpondrá a las ventanas que aparecían antes en la pantalla.

Para desactivar una ventana basta con activar una nueva.

Para cerrar una determinada ventana en primer lugar ésta debe aparecer como activa; después, la ventana puede cerrarse utilizando alguno de los procedimientos siguientes:

a.- Pulsando [Ctrl]+[F4].

b.- Haciendo doble "clic" en el recuadro superior izquierdo (botón de control) de la ventana activa.

c.- Haciendo un "clic" en el botón de control de la ventana y eligiendo la opción Cerrar.

Para mover una ventana hay que situar el puntero del ratón sobre el título de la ventana y mantener pulsado el botón izquierdo del ratón mientras la ventana se desplaza hasta la posición deseada. Una vez que la ventana ha llegado hasta el lugar deseado basta soltar el botón.

Para modificar el tamaño de una ventana lo primero que hay que hacer es activarla y después situar el puntero del ratón sobre uno de sus lados hasta que aparece una doble flecha; pulsado el botón del ratón el borde de la ventana puede ser arrastrado hasta que éste alcance el tamaño deseado.

Si se desea restaurar automáticamente una ventana al número y disposición iniciales, basta elegir en la barra de Menú la opción Ventanas y seleccionar, dentro de ella, el comando "Restaurar Entorno".

2.- EL MODO DIRECTO

Las palabras clave que se usan para comunicarse con el entorno se llaman PRIMITIVAS. Las primitivas pueden escribirse en la ventana de TRABAJO. Cuando el objetivo de la primitiva sea dibujar, sus resultados se muestran en el área de GRÁFICOS y, por el contrario, aparecen en el área de TEXTOS cuando la primitiva tiene por objeto dibujar.



Ventana de Trabajo

Con el ejemplo siguiente quedará más claro el funcionamiento del modo directo:

Lo primero que debe hacerse es activar el área de TRABAJO; para ello se sitúa el puntero del ratón sobre algún punto de la ventana de TRABAJO y después se

hace "clic" en ella con el ratón. Instantáneamente el título de esa ventana aparecerá resaltado, lo cual indica que esa zona es activa.

Si se escribe:

AVANZA 40 y se pulsa [INTRO]

La tortuga dibujará una recta de cuarenta unidades de longitud en la ventana GRÁFICOS, ya que el resultado de la primitiva AVANZA es un dibujo.

Prueba con otra primitiva, escribe ESCRIBE "HOLA y después pulsa [INTRO]. En la ventana de TEXTOS aparecerá escrita la palabra HOLA. Las órdenes de escritura se ejecutan en la zona de TEXTOS.

RECUERDA



Las órdenes se escriben en la ventana de Trabajo. Logo muestra sus mensajes y textos en la ventana de Textos, sus gráficos los representa en la ventana de Gráficos.

Si se comete un error al escribir una primitiva como por ejemplo ordenar a la primitiva una acción incompleta, Logo envía su mensaje de error a través de la ventana de TEXTOS. Para comprobarlo se teclea: ESSCRIBE "HOLA; la palabra escribe se ha escrito con dos eses; el ordenador en este caso emitirá un mensaje de error en la ventana de TEXTOS. El mensaje de error que va a parecer es: no se cómo hacer ESSCRIBE.

Realiza las pruebas siguientes: escribe los textos tal y como aparecen a continuación y pulsa [INTRO] después de cada línea.

AVANZA 80
AV 80
BORRAPANTALLA
AV 40
GIRADERECHA 90
AV 40 GD 90
BP AV 60
GIRAIZQUIERDA 45
RETROCEDE 60
GI 90 RE 56
ESCRIBE "CASA
ESCRIBE [LA CASA ES GRANDE]
BP

ACTIVIDAD RECOMENDADA



Dibuja un cuadrado y un triángulo haciendo uso de las primitivas que has visto hasta ahora. Prueba también a hacer una circunferencia.

Con el ejercicio anterior se pretendía que practicases con un conjunto de primitivas y sus abreviaturas. Escribe ahora las líneas de la lista siguiente y observa los mensajes que aparecen en la ventana de Textos:

AVANZA
RETROCEDE
GIRAIZQUIERDA

Analiza los mensajes escritos en el zona de textos; el error se produce porque las primitivas requerían un parámetro para su correcta definición; recuerda que existen otras primitivas para la que no es necesario definir ningún parámetro. Escribe las órdenes anteriores completándolas con un parámetro correcto.

Primitivas que has estudiadas hasta el momento:

PRIMITIVA	ABREVIATURA	ACCIÓN
AVANZA x	AV x	la tortuga se desplaza x unidades hacia adelante
RETROCEDE x	RE x	la tortuga se desplaza x unidades hacia atrás
GIRADERECHA x	GD x	la tortuga gira a la derecha x grados
GIRAIZQUIERDA x	GI x	la tortuga gira a la izquierda x grados
ESCRIBE "X"	ES "X"	escribe en la ventana de textos la palabra situada detrás de las comillas
ESCRIBE [XX XX]	ES [XX XX]	escribe en la ventana de textos la frase situada entre los corchetes
BORRAPANTALLA	BP	se borra la pantalla de gráficos

También se pueden escribir dos o más órdenes en una misma línea. Por ejemplo:

BORRAPANTALLA AVANZA 40 GIRADERECHA 90 AVANZA 60

Después de escribir la línea anterior en la ventana de trabajo y pulsar [INTRO], se ejecutarán las cuatro instrucciones.

Edición en modo directo

También se puede repetir una orden directa pulsando [INTRO] después de haber colocado el cursor en la línea en la que la orden se había ejecutado.

ACTIVIDAD GUIADA



Escribe la siguiente orden y pulsa [INTRO]:

GD 45

El cursor baja una línea y el programa espera a que escribas otra orden. Sube el cursor a la línea anterior y vuelve a pulsar [INTRO]. Repite la acción varias veces, verás cómo se ejecuta como si la hubieras escrito de nuevo.

Como se ha explicado anteriormente, en una línea se pueden escribir varias órdenes seguidas. Logo las ejecuta una tras otra.

Borra la pantalla de Gráficos con la orden BP y, en una línea en blanco escribe las dos órdenes siguientes y pulsa [INTRO]:

AV 60 GD 90

Sube el cursor a la línea anterior y pulsa [INTRO]. Repite la operación dos veces más. Cuando concluyas el ejercicio observa que Logo ha dibujado un cuadrado.

Esta posibilidad permite corregir un error sin necesidad de reescribir toda la línea de órdenes en la que aparecía dicho error.

Escribe y ejecuta:

HESCRIBE "ERROR

Logo envía un mensaje a la ventana de textos el siguiente mensaje:

no sé como hacer para HESCRIBE

Sube el cursor a la línea errónea, suprime la H y pulsa [INTRO]. Observa que en la ventana de textos Logo escribe la palabra que le habías ordenado.

Una primitiva interesante: REPITE

Merece la pena dedicarle un apartado específico a la primitiva REPITE debido a su gran utilidad en la creación de gráficos y en el micromundo de control. Esta primitiva no tiene abreviatura; su sintaxis es:

REPITE n[acciones]

ACTIVIDAD GUIADA



Realiza los siguientes ejercicios y pulsa [INTRO] al final de cada línea:

REPITE 4[AV 50 GD 90]

BP

REPITE 6 [AV 40 GD 60]

BP

REPITE 7 [AV 30 GD 360 / 7]

Cambia el siete por nueve y pulsa [INTRO].

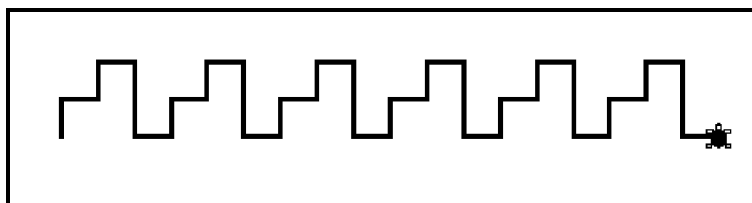
Escribe el siguiente conjunto de órdenes:

BP SL GI 90 AV 200 GD 90 BL

REPITE 10[AV 5 GD 90 AV 20 GD 90 AV 5 GI 90 AV 20
GI 90]

Observa que la última orden del corchete GI 90 deja a la tortuga en la orientación inicial de modo que, al ejecutarse el módulo siguiente, la tortuga esté situada en la misma dirección.

PROPUESTA: Realiza más cenefas como la que aparece en la figura siguiente:



3.- MODO PROCEDIMENTAL

El lenguaje Logo permite trabajar en modo procedimental, característica que comparte con otros lenguajes de ordenador. Programar en modo procedimental consiste en enseñar procedimientos al ordenador que, una vez asimilados, interpreta como si fueran nuevas primitivas.



Icono de Edición

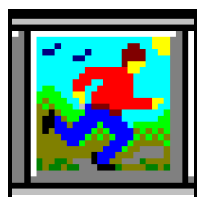
Los procedimientos pueden escribirse en las ventanas de trabajo y edición, aunque es preferible hacerlo en la de edición. Para ello, se pulsa el icono de Edición que aparece en la barra de menú que se superpone a la ventana de edición. Los procedimientos se escriben en la ventana de edición.

Existen dos PRIMITIVAS asociadas a cualquier procedimiento, PARA y FIN. Todos los procedimientos comienzan por la primitiva PARA y terminan con la primitiva FIN.

Escribe el siguiente procedimiento:

```
PARA CUADRADO  
REPITE 4 [ AV 60 GD 90]  
FIN
```

Hay que tener en cuenta que en la primera línea, sólo puede escribirse la palabra que se está definiendo separada de PARA por un espacio y en la última línea sólo puede escribirse la palabra FIN. Siempre debe pulsarse [INTRO] para validar un procedimiento



Intérprete

El procedimiento anterior se llama CUADRADO. Para ejecutar posteriormente este procedimiento, es necesario que el ordenador lo memorice; para ello, basta pulsar el icono de INTERPRETE. En la ventana de Texto aparecerá, si el procedimiento es correcto, el siguiente mensaje: ACABAS DE DEFINIR CUADRADO. Con esta acción, el ordenador ha memorizado el procedimiento, por lo que

puede ser utilizado en el futuro. Cierra la ventana de Edición situando el cursor en el cuadrado superior izquierdo y pulsa dos veces el ratón. Activa la ventana de trabajo. Si escribes CUADRADO y pulsas [INTRO], en la ventana de gráficos se ejecutarán las órdenes asociadas a ese procedimiento (en este caso aparece cuadrado de 60 unidades de lado).

Los procedimientos recién definidos pueden ejecutarse mientras no se salga de Logo. Una vez apagado el sistema, Logo sólo recuerda sus primitivas.

Cómo archivar los procedimientos que se encuentran en el Editor.

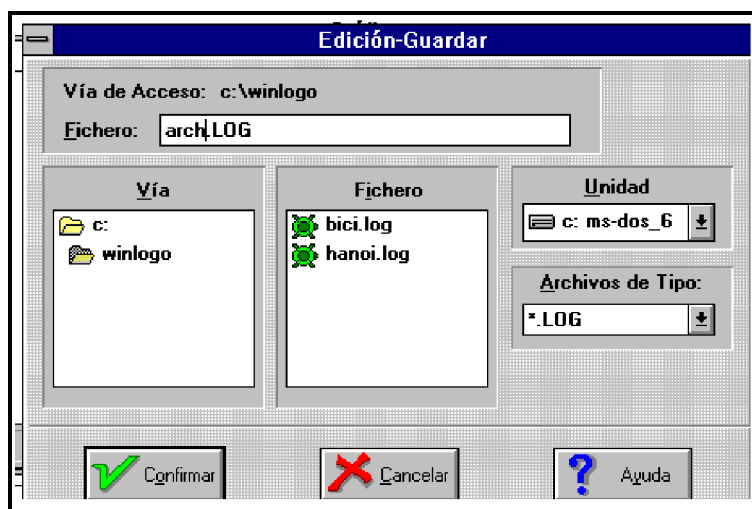
Para no perder los procedimientos diseñados de un día para otro, Logo permite archivar en el disco los procedimientos que se encuentran visibles en la pantalla de edición. Para archivar los procedimientos pueden seguirse los siguientes caminos:

a.- Desde la ventana de Trabajo: Mediante la primitiva GUARDAED se almacenan todos los procedimientos en un determinado archivo. Por ejemplo, con GUARDAED "arch.log, se almacenan los procedimientos en el archivo arch. log.

b.- Desde la barra de menú teniendo activa la ventana de EDICIÓN: Se elige la opción GRABAR dentro del menú ARCHIVO.

Cuando aparece la ventana de diálogo adjunta, se escribirá el nombre con el que se quiera guardar el conjunto de procedimientos. En este caso se elige el nombre ARCH.

Conviene señalar que los archivos de procedimientos, tienen por defecto la extensión LOG.



c.- Teniendo activa la ventana de EDICIÓN, es posible elegir el icono GRABAR (salvar en disco) y seguir los pasos del punto anterior.



Icono de grabar



Si has comprendido los contenidos presentados hasta ahora y has realizado las prácticas sin dificultad, continúa con el estudio de LOGO.

4.- ALGUNAS PRIMITIVAS MÁS DEL LENGUAJE LOGO

PRIMITIVA	ABREVIATURA	ACCIÓN
SUBELAPIZ	SL	Eleva el lápiz de la tortuga activa para que no deje trazo al moverse.
BAJALAPIZ	BL	Activa el lápiz de la tortuga y dibuja un trazo a lo largo de su trayectoria.
GUARDAED "archi		Guarda el contenido del área de edición en un archivo llamado archi.
REPITE		Repite el número de veces indicado la acción situada entre corchetes.
CENTRO		Sitúa las tortugas activas en el centro del gráfico. En su desplazamiento al centro, las tortugas dejan rastro o no, dependiendo de si su posición es bajalapiz o subelapiz.

5.- PROGRAMACIÓN MODULAR

La estructura de Logo, basada como ya se ha visto, en los procedimientos definidos, permite realizar programas de forma modular, es decir, descomponiéndolos en partes que pueden resolverse por separado y que son más simples que el programa inicial. Esta modularización puede hacerse a varios niveles, ya que cada módulo puede descomponerse, también, en partes más sencillas. Este proceso permite dar respuesta a los planteamientos de la programación estructurada. Logo es un lenguaje que estimula la realización de programas claros, en los que cada procedimiento realiza una misión concreta.

Modularidad

La posibilidad que ofrece Logo de dividir un programa complejo en otros más pequeños y fáciles de diseñar que pueden dividirse en otros, que se integran en el programa inicial para dar respuesta al problema planteado, se llama modularidad.

Otra definición de modularidad es la siguiente: modularidad es una característica de Logo que permite que un programa llame a otros más sencillos y que estos, también, puedan llamar a otros.

Uno de los problemas que plantea la modularidad en programas geométricos es la necesidad de tener controlada la posición y orientación de la tortuga entre módulo y módulo. A continuación, se ofrecen dos formas de resolver este problema.

Modularidad con referencia al centro

Se propone, como un ejemplo, un programa cuya misión es dibujar un robot. Para aplicar el concepto de modularidad, se va a dibujar un robot por partes correspondientes a cada uno de sus miembros (cabeza, cuello, cuerpo, brazos, patas y pies).

```
PARA ROBOT
BP BL PONG 2 PONF 3
CABEZA CEN
CUELLO CEN
BRI CEN
CUERPO CEN
BRD CEN
PATAIZ CEN
PIEIZ CEN
PATAD CEN
PIED CEN
FIN
```

La referencia al centro (CEN) es la característica que más destaca en el ejemplo anterior por el hecho de que, después de dibujar cada miembro del robot, la tortuga vuelve al centro de la pantalla para dibujar la siguiente pieza. Este movimiento que se repite al final de cada módulo se realiza por medio del programa CEN.

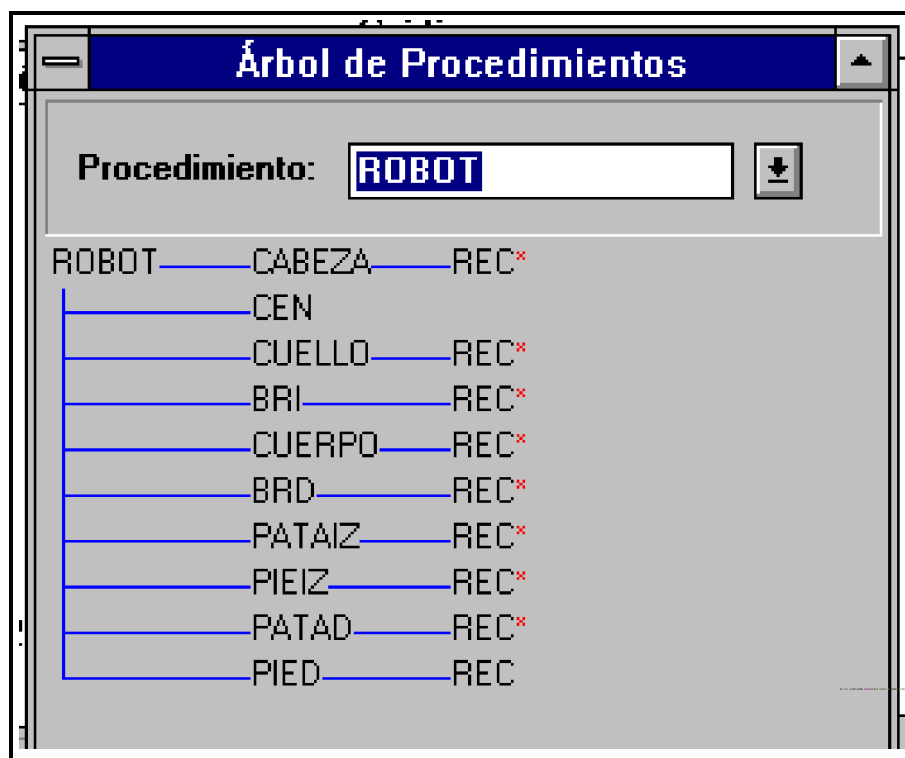
```
PARA CEN
SL PONRUMBO 0 CENTRO
FIN
```

Por otro lado, los módulos encargados de dibujar los miembros trazan rectángulos, por lo que se ha creado un módulo variable, REC que se adapta al tamaño de rectángulo

que interese en cada momento, de forma que cada dibujo comienza y termina en el ángulo inferior izquierdo.

```
PARA REC :L :A
BL
REPITE 2 [ AV :L GD 90 AV :A GD 90]
FIN
```

La estructura de este programa modular es:



Los módulos que dibujan los miembros, CABEZA, CUELLO y el resto, comienzan su ejecución subiendo el lápiz y colocando la tortuga en la posición correspondiente al vértice inferior izquierdo del rectángulo que se dibuje en cada caso. El uso de papel cuadriculado supone una gran ayuda en esta labor.

Modularidad con referencia al módulo anterior.

El problema que se plantea es el de conocer en todo momento la posición de la tortuga cuando termina un módulo, ya que es necesario situar la tortuga en un punto determinado para comenzar el módulo siguiente. Como ejemplo de diseño, se toma un tren dividido en cada una de las unidades que integran el convoy. Consulta el manual de LOGO para conocer las primitivas PONG, PONF y PONCL.

PARA TREN
BP PONG 1 PONF 1 PONCL 4 OT
SL AV 10 GI 90 AV 150 GD 90
MAQUINA
VACARBON
VAGON1
VAGON2
VAGONCOLA
VIA
MT
FIN

Al final de cada módulo, la tortuga dibuja el parachoques y después realiza un giro de noventa grados hacia la izquierda para disponerse a dibujar el siguiente rectángulo.

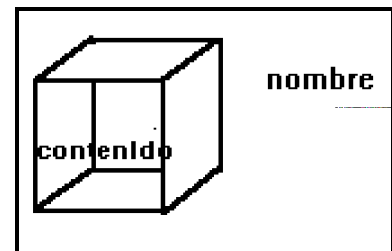


Comunícate con la tutoría si necesitas aclarar conceptos antes de proseguir.

6.- USO DE VARIABLES

Las variables pueden ser comparadas con cajas a las que se les pone una etiqueta (nombre) y en cuyo interior existe un contenido.

Una variable, por ejemplo, puede llamarse HOLA y contener el número 34; otra puede llamarse BARCO y contener la palabra chocolate. De este modo, al ejecutar la orden de escribir el contenido de HOLA, el ordenador escribe 34, de la misma manera que si se le mandase escribir el contenido de BARCO escribiría chocolate.



Las variables en el modo directo:

Sitúate en la ventana de trabajo.

Escribe:

HAZ "BARCO "chocolate

Pulsa [INTRO].

Mediante la instrucción anterior se ha creado la variable BARCO y se le ha asignado como contenido la palabra chocolate.

Escribe :

ESCRIBE : "BARCO

Pulsa [INTRO].

En la ventana de textos, aparece escrito el contenido de la variable BARCO, en este caso chocolate. Conviene destacar que para conocer el contenido de una variable, es necesario escribir la instrucción ESCRIBE : "nombrevar, donde nombrevar es el nombre de la variable:

Prueba el siguiente ejemplo:

```
HAZ "h 66  
ESCRIBE :h + 9
```

En este caso en la ventana de textos aparecerá el número 75, que es el resultado de sumar 66, contenido de la variable h, al número 9.

Las variables locales en los procedimientos .

Retrocedamos al procedimiento CUADRADO

```
PARA CUADRADO  
BP BL PONCL 5 PONG 4 PONF 16  
REPITE 4 [AV 50 GD 90 ]  
FIN
```

Como se ha visto, este procedimiento permite realizar un cuadrado concreto y único de 50 unidades de lado, de modo que si se quisiera realizar otro cuadrado más grande o de otro color, debería diseñarse un procedimiento distinto. Las variables locales permiten solucionar este problema.

Observa el procedimiento que a continuación se analiza:

```
PARA CUAVARILADO :L  
BP BL PONCL 5 PONG 4 PONF 16  
REPITE 4 [AV :L GD 90 ]  
FIN
```

RECUERDA



El nombre del procedimiento que se coloca en la primera línea después de PARA conlleva la creación de la variable :L.

Dentro del corchete, el argumento de la primitiva AVANZA no es una cantidad fija, sino que varía en función del contenido de la variable :L por lo que, dependiendo de dicho contenido, el programa dibuja el cuadrado correspondiente.

Copia el procedimiento CUAVARILADO :L y no olvides pulsar [INTRO] después de FIN; escribe y ejecuta:

```
CUAVARILADO 50
CUAVARILADO 30+5
CUAVARILADO 1
CUAVARILADO
```

Observa que Logo envía un mensaje para que completes los datos.

Las variables locales se llaman así porque su existencia está limitada al tiempo de ejecución del programa. Para comprobar este hecho, escribe la frase:

ESCRIBE :N

Observa el mensaje que envía Logo: "Esa variable ya no existe."



Crea diversos procedimientos que permitan dibujar cuadrados de lado fijo y color, grosor, o fondo variables. Llámalos:

CUAVARICOL :C CUAVARIGRO :G CUAVARIFON :F

Ejecuta dichos procedimientos utilizando distintos parámetros.

Logo puede gestionar en cada procedimiento tantas variables como se definan en la primera línea. Fíjate en el siguiente ejemplo:

```
PARA CUAVARIA :C :G :F
BP BL
PONCL :C
PONG :G
PONF :F
REPITE 4 [ AV 50 GD 90 ]
FIN
```

Una vez definido el procedimiento en el Editor y después de haberlo validado, vuelve al área de trabajo y ejecuta:

```
CUAVARIA 5 4 16
CUAVARIA 1 1 1
CUAVARIA 2 4
```

Observa que en el último caso Logo envía un mensaje para que se completen los datos.



Modifica el procedimiento anterior en el Editor para que incluya como cuarta variable la longitud del lado. (no olvides salir del Editor validando los cambios con el icono intérprete).

7.- VIAJE TOTAL

Es importante notar que el ángulo de giro que recorre la tortuga es el ángulo exterior y no el interior de los polígonos como, en un principio, podría suponerse; de modo que el giro total que realiza la tortuga después de dibujar un polígono es de 360° . Conviene ayudar al alumnado a que descubra que la diferencia entre un polígono y otro no estriba sólo en su número de lados, sino en la amplitud de sus ángulos.

Se va a definir un procedimiento que permite realizar un polígono con un número variable de lados:

```
PARA POLIN :N  
BP BL  
PONCL 1 PONG 2 PONF 3  
REPITE :N [AV 50 GD 360 / :N ]  
FIN
```

Ejecuta el programa desde el área de trabajo con los siguientes parámetros:

```
POLIN 6  
POLIN 3  
POLIN 2  
POLIN 50
```



Crea procedimientos en los que varíen otros parámetros como la longitud de los lados.

Realiza las modificaciones necesarias en el Editor para que este programa dibuje los polígonos apoyados siempre con el lado inferior en posición horizontal.

Diseña un programa que realice circunferencias de tamaño variable.



Envía los procedimientos a la tutoría y recibirás comentarios al respecto

Las variables globales en los procedimientos

Las variables globales, a diferencia de las locales, pueden existir durante todo el tiempo en que Logo está activo; puede decirse que "permanecen suspendidas en el área de trabajo", lo cual permite usarlas en cualquier procedimiento que se cree. Esta característica implica una ocupación de la memoria y la vigilancia del programador para que no haya dos variables definidas con el mismo nombre.

Las variables globales se crean con la primitiva HAZ usada con la siguiente sintaxis:

HAZ "NOMBRE contenido

El nombre que se da a la variable debe ser un conjunto de caracteres sin espacios en blanco y debe ir precedido por comillas. El contenido de la variable puede ser:

- Un número
- El resultado de una operación
- Una palabra que irá precedida de comillas
- Una lista entre corchetes
- Una serie de datos introducidos desde teclado durante la ejecución del programa por medio de ciertas primitivas.

El contenido de una variable aparece cuando se escribe precedida por el símbolo :nombre.

No es posible crear una variable global ni en la primera línea de un procedimiento, ni en la última. Sirva como ejemplo el siguiente procedimiento:

```
PARA CUAGLO
BP BT
ES [ Escribe la longitud del lado de un cuadrado]
HAZ "LADO LEEPALABRA
REPITE 4 [ AV :LADO GD 90]
FIN
```

```
PARA SALUDO
ES [¿Cómo te llamas?]
HAZ "NOMBRE LEEPALABRA
ES FRASE [Encantado de conocerte] :NOMBRE
FIN
```

```
PARA CUBO
BP BT
ES [Este programa calcula el cubo de las cifras arábegas]
ES [¿De qué cifra deseas el cubo?]
HAZ "CIFRA LEECAR
HAZ "CUBO :CIFRA * :CIFRA * :CIFRA
ES (FRASE [El cubo de ] :CIFRA [ es ] :CUBO)
FIN
```

PRIMITIVA	ABREVIATURA	ACCIÓN
HAZ "nombr obj		Asigna el valor indicado en obj a la variable llamada nombr
LEECAR	LC	Lee el primer carácter de un archivo o la primera pulsación desde el teclado sin necesidad de pulsar [INTRO].
LEECARS num	LCS num	Lee los num primeros caracteres pulsados desde el teclado o leídos de un archivo
LEELISTA	LL	Lee una lista de caracteres introducidos desde el teclado o leídos desde un archivo
LEEPALABRA	LP	Lee la primera palabra introducida desde el teclado o leída desde un fichero

Diferencias entre carácter, palabra y lista:

- Carácter es una sola letra, número o pulsación de teclado. (Ej. : a)
- Palabra es un conjunto de caracteres que no tienen espacios entre sí. (Ej. : 123a)
- Lista es un conjunto de palabras separadas por espacios (Ej. : La casa número 18)

Comprueba el comportamiento de estos procedimientos (las frases situadas después de ";" son comentarios, no líneas de programación):

```

PARA lectura_caracteres1
ES [¿Quién descubrió América?:]
HAZ "H LC ; se asigna a la variable H el 1er carácter escrito por teclado
ES :H
FIN

```

Obsérvese que, cuando el nombre de un procedimiento contiene más de una palabra, se usan guiones para unirlos. En el procedimiento anterior no es necesario pulsar [INTRO] para la aceptación del dato.

```

PARA lectura_caracteres2
ES [¿Quién descubrió América?:]
HAZ "H LCS 3 ; se asigna a la variable H los tres primeros caracteres escritos por teclado
ES :H

```

FIN

Tampoco es necesario pulsar [INTRO] para indicar al procedimiento que se ha acabado de introducir los datos, puesto que está esperando tres caracteres y, una vez recibidos, continúa el flujo del programa. Si quieres introducir un número menor de caracteres que los indicados, pulsa [INTRO] cuando decidas no introducir más.

```
PARA lectura_caracteres3
ES [¿Quién descubrió América?:]
HAZ "H LP ; asignamos a la variable H la primer palabra escrita por teclado
ES :H
FIN
```

```
PARA lectura_caracteres4
ES [¿Quién descubrió América?:]
HAZ "H LL ; asignamos a la variable H todo el texto escrito por teclado
ES :H
FIN
```

8.- RECURSIVIDAD

Se llama recursividad a la posibilidad que ofrece un programa diseñado en Logo de llamarse a sí mismo en un bucle sin fin. Un sencillo ejemplo lo constituye el siguiente programa, que dibuja una circunferencia de modo recursivo:

```
PARA CIREC
AV 1
GD 1
CIREC
FIN
```

Un primer problema por resolver en un procedimiento recursivo, es encontrar la forma de abandonar su ejecución infinita de procedimiento sin pulsar la tecla [ESC]. se puede abandonar dicha ejecución mediante la primitiva TECLA? en una condición.

Sentencias condicionales

Logo puede "valorar" el contenido de una condición y "decidir" entre dos opciones. Para dejar claros estos conceptos se utilizará la primitiva TECLA?, que devuelve CIERTO o FALSO según se haya pulsado o no una tecla.

El programa CIREC queda controlado al anteponer el condicional SI a la primitiva TECLA?


```
PARA ARCO
SI TECLA? [ALTO]
AV 1
GD 1
ARCO
FIN
```

Este programa va dibujando un arco con avances y giros hasta que se pulse una tecla.

Es importante insistir sobre la estructura de las sentencias condicionales, ya que son la base del "pensamiento" de los programas en Logo.

La sentencia condicional se expresa con la primitiva SI y tiene varias sintaxis; las dos más usadas son:

```
SI condición [acción 1] [acción 2]
```

Esta sentencia significa que, si se cumple la condición, se ejecutan las acciones contenidas en el primer corchete y, en caso contrario, las incluidas en el segundo. En ambos casos, una vez realizadas todas las acciones incluidas en uno de los corchetes, el programa ejecuta la línea siguiente del programa.

```
SI condición [acción]
```

En este caso, si se cumple la condición, se ejecutan las acciones contenidas en el corchete; después el programa pasa a la línea siguiente; si no se cumple la condición, el programa pasa a la línea siguiente.

Importancia de la recursividad en LOGO

Las estructuras recursivas se utilizan muy a menudo en los programas de robótica. En gran cantidad de ocasiones las máquinas, además de ejecutar acciones habituales, están alertas ante cualquier cambio que pueda producirse gracias a su programación. Dentro de su programa de funcionamiento, se incluyen programas recursivos que se activan debido a los cambios en las condiciones de la máquina.

Conviene tener en cuenta que la recursividad es más que un bucle, una llamada al propio programa para que vuelva a realizar las mismas acciones en un ciclo sin fin. Tal vez, la etiqueta de un conocido anís español pueda ejemplificar mejor el fenómeno recursivo. En ella, un mono con un cierto toque "darwiniano", sostiene una botella del mismo anís, la cual contiene a su vez otra etiqueta más pequeña que contiene la representación de un mono que sostiene una botella. Conviene aclarar, por tanto, que cada vez que Logo se llama a sí mismo, genera un nuevo programa que debe

ejecutarse en su totalidad. Esta última afirmación, que puede parecer una evidencia incuestionable, sirve como eficaz sistema de control de los programas recursivos y debe tenerse siempre presente para no recargar los programas con la "basura de las colas" que se van generando.

Control de la recursividad por medio de un contador

Ejemplo:

```
PARA CONTADOR :N
SI :N<0 [ESCRIBE "GOL ALTO]
ES :N
CONTADOR :N-1
FIN
```

El programa contiene una variable local :N que sólo sirve para escribir su contenido cada vez que el programa se ejecuta. El programa se llama a sí mismo y hace disminuir el valor de la variable en cada paso. La sentencia condicional de la primera línea controla la ruptura del bucle en el momento en que se llega a un valor negativo, se escribe la palabra GOL y se detiene el programa. Obsérvese que la comparación numérica entre el contenido de la variable y la constante no se ha hecho con el signo = sino con el signo < ya que, si el incremento se hiciera con decimales (CONTADOR :N-0, 3), nunca se cumpliría la condición y el bucle sería indefinido.



- Construye otro contador que aumente el valor de la variable :N
- Construye otro contador con dos variables, el número de inicio y el descenso.
- Construye un programa recursivo que dibuje espirales utilizando un avance y un giro, e incrementando uno o ambos.
- Modifica el programa anterior con otras variables para cada incremento



No avances sin antes aclarar tus dudas. Envía los procedimientos del último ejercicio a tu tutoría.

Las colas en la recursividad.

En un párrafo anterior, se afirma que cada vez que un programa se llama a sí mismo genera un nuevo programa que debe terminarse. Es importante no cerrar el tema de la recursividad sin presentar los "extraños efectos" que la recursividad produce en un programa. La capacidad de Logo para evitar dichos efectos pone de manifiesto cuando se evitan, a través de una correcta programación.

Observa este programa:

```
PARA CAPICUA :N
SI :N<1 [ALTO]
ES FRASE [CABEZA] :N
CAPICUA :N-1
ES FRASE [COLA] :N
FIN
```

Antes de ejecutar CAPICUA 9, te proponemos la apuesta de intentar averiguar lo que escribirá Logo cuando se ejecute CAPICUA 9. Una vez hayas formulado tu apuesta, observa que CAPICUA 9, después de pasar por la primera línea de control escribe CABEZA 9 e inmediatamente después se convierte en CAPICUA 8 sin haber terminado el programa, dejando pendiente para otro momento la ejecución de la frase COLA 9. El programa CAPICUA 8 hará lo mismo dejando su última orden COLA 8 para otro momento, y así sucesivamente hasta que :n llegue a valer 0, momento en que, en lugar de terminar, ejecutará todas las colas empezando por la del último programa ejecutado, CAPICUA 1 hasta llegar al primero.



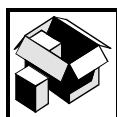
Llama a la tutoría cuando pienses que has comprendido los conceptos básicos de la programación en LOGO y sepas resolver problemas con la programación correspondiente. Solicita realizar un ejercicio de evaluación que te remitirá la tutoría.

III.- MANOS A LA OBRA

III.- MANOS A LA OBRA

Como preámbulo al control y a la programación por ordenador es conveniente, para no incurrir en falsas concepciones, hacer referencia a varios conceptos relacionados con el mundo de la automatización y el control.

Deben diferenciarse los matices grandes o pequeños que distinguen los distintos procesos y grados de control como son la mecanización, la automatización, el control numérico, el servocontrol y la robotización.



Observa detenidamente el vídeo que acompaña a la unidad.

1.- PROCESOS PARA LA MEJORA Y EL CONTROL DE LA PRODUCCIÓN

Se llama mecanización a la incorporación de máquinas en la realización de determinadas tareas. Así, se habla de la mecanización del campo, cuando a las tareas agrícolas se han incorporado máquinas por todos conocidas como el tractor, el arado o la cosechadora. También se habla del mecanizado de piezas en un taller, cuando para su fabricación se utilizan máquinas como tornos o fresadoras, es decir, cuando se abandona la fabricación manual y se sustituye por procesos mecanizados que permiten mejores acabados y mayor rapidez en la confección de elementos.

Un paso más allá es la automatización, considerada como la supresión parcial o total de la intervención de las personas en la realización de tareas productivas, como las tareas agrícolas, industriales o administrativas.

Los autómatas son un caso muy conocido de control, ya tradicional, que se ha venido aplicando a aquella clase de máquinas en las que una fuente de energía acciona un mecanismo, que permite imitar los movimientos de los seres animados. Se conocen autómatas que fueron contruidos por los griegos en el templo de Dédalo; sin embargo, uno de los casos más difundidos es el del Pato de Vaucanson (Grenoble 1709-París 1782) construido en 1738, que era un pato artificial capaz de batir las alas, zambullirse, nadar, tragar grano e incluso expeler una sustancia parecida al excremento. Otro caso de autómatas célebres aunque más próximo a nuestro tiempo es el del jugador de ajedrez de Torres Quevedo, construido en 1912, capaz de jugar finales de partida (rey contra rey y torre). Pero debe ponerse de manifiesto que los autómatas siempre repiten el mismo modelo de actuación, no son reprogramables y tampoco son capaces de variar sus acciones en función del entorno o la situación.

El control numérico es la supervisión y regulación de determinadas tareas mecánicas de precisión, realizadas por una máquina herramienta. El control de estas tareas se realiza de forma automática para evitar, de este modo, que el control se lleve a cabo por un operario que, en ocasiones, podría verse sometido a ciertos riesgos en un proceso donde es imposible erradicar los errores humanos. De esta forma, se ajusta al máximo la precisión en la confección de piezas estandarizadas y se libera al operario de su control, mejorando la calidad y la cantidad del trabajo realizado. Un ejemplo de control automático es el control de la velocidad de giro de un taladro o la velocidad y control de avance de un torno o fresadora.

Otro sistema de control automático es el servocontrol. Consiste en controlar de forma automática las acciones de una máquina en función de unos parámetros definidos y sus variaciones. Por ejemplo, el servocontrol se puede utilizar para controlar la velocidad de giro de un torno, de forma que se mantenga fija dentro de unos límites. Otro ejemplo de servocontrol podría ser el del freno de algunos vehículos en los que la fuerza transmitida a las ruedas es proporcional a la fuerza aplicada por el conductor sobre el pedal del freno; de esta función se encarga un mecanismo servocontrolado que se llama servofreno.

La robotización es también una automatización de procesos sin la intervención humana, pero en este caso se da un paso más; hay desplazamiento de cargas, manipulación de objetos y un fuerte componente de realimentación. Es decir, este tipo de automatización permite la manipulación automática y programable de acciones y objetos.

La realimentación es un proceso imprescindible en la robotización, ya que dota a un proceso de capacidad para captar información que, una vez procesada por la máquina, permite modificar su comportamiento (sus acciones). Una máquina que posea la capacidad de realimentación es capaz de modificar sus respuestas en función de las variaciones de su entorno.

Como curiosidad, la palabra Robot fue creada por el escritor checo Karel Capek, en 1920, en su obra de teatro titulada "Rusum's universal robots", para dar nombre a un androide, construido por un científico, que era capaz de realizar todas las tareas y trabajos propios de las personas.

Centrando el análisis en las diferencias que existen entre automatización y robotización, puede decirse que una máquina automatizada (autómata) responde siempre de igual manera ante sucesos de idéntica naturaleza. Mientras que por el contrario un robot, es decir, una máquina robotizada, se caracteriza porque puede manejar objetos y, lo más interesante, es un dispositivo multifuncional y reprogramable. Una máquina robotizada es capaz de hacer trabajos totalmente diferentes y adaptarse al medio, ya que puede tomar decisiones en función de las condiciones exteriores.



Piensa en una relación de máquinas habituales en tu vida cotidiana. Distingue, para cada una de ellas, si pueden ser consideradas o no como robots e indica a qué categoría de las anteriormente descritas pueden adscribirse las que no sean robots.
Envía tu opinión a la tutoría.

La totalidad de los procesos de mejora y control de la producción pueden sintetizarse en tres fundamentales: Procesos de mecanización, procesos de automatización y procesos de robotización. Las principales características de cada uno de estos tres grupos se exponen en el cuadro adjunto:

PROCESO	CARACTERÍSTICAS
MECANIZACIÓN	<ul style="list-style-type: none">* Incorporación de máquinas:<ul style="list-style-type: none">- Realizan procesos repetitivos.- Sustituyen el esfuerzo humano.- Manejadas por operador humano.
AUTOMATIZACIÓN	<ul style="list-style-type: none">* Sustitución parcial o total de intervención humana.* Sistemas capaces de autorregulación.
ROBOTIZACIÓN	<ul style="list-style-type: none">* Coordinación de automatismos.* Reprogramación informática.* Adaptabilidad a diversas tareas (polivalencia)

2.- EL ORDENADOR EN LOS PROCESOS DE ROBOTIZACIÓN

El ordenador juega un papel imprescindible en los procesos de robotización. Los sistemas controlados por ordenador ofrecen importantes ventajas sobre los sistemas que no lo están, ya que en ellos se puede prescindir de determinados operadores como los relés o temporizadores, a la vez el ordenador aporta una gran flexibilidad al permitir cambiar el funcionamiento de un sistema técnico sin necesidad de rediseñarlo. Para cambiar el funcionamiento basta modificar el programa que gestiona el sistema técnico.

Para que una máquina robotizada pueda ser reprogramada es necesario que alguna persona intervenga en esa reprogramación; es imprescindible, por tanto, un sistema de comunicación entre la máquina y el hombre; el proceso de comunicación se realiza mediante un lenguaje de programación que permite al hombre introducir modificaciones en la máquina dentro de sus procesos de análisis y, por tanto, en sus respuestas.

En este curso el ordenador se va a utilizar para robotizar máquinas; para definir los procesos que ha de realizar y las decisiones que la máquina controlada debe tomar, se empleará el lenguaje Logo.

Antes de continuar, conviene detenerse y reflexionar sobre el componente más importante que debe poseer cualquier máquina que se pretenda robotizar. Si, como se ha dicho, un robot puede variar su comportamiento según se modifique su entorno, es deducible que para que una máquina pueda ser programada ha de poseer sensores que informen al robot de dichas variaciones. Así modificará su actuación según la respuesta elaborada por un programa "inteligente"; esta es la cualidad más importante que diferencia las máquinas programables de las que no lo son.

A continuación se incluye un ejemplo de máquina robotizada en el que se pone de manifiesto la gran ventaja que para una máquina supone la reprogramación. Se trata de controlar una máquina de forma que sea capaz de clasificar manzanas Golden en función de su color y tamaño, de forma que todas las manzanas que no sean amarillas vayan a una caja de rechazos y, respecto al tamaño, se definen los siguientes intervalos de clasificación:

- Manzanas cuyo diámetro sea superior a 7 cm.
- Manzanas cuyo tamaño diámetro esté comprendido entre 5, 5 cm y 7 cm.
- Manzanas cuyo diámetro sea inferior a 5, 5 cm.

La máquina debe estar dotada de unos sensores capaces de detectar el color y el tamaño de cada manzana para depositar después cada una de ellas en la caja correspondiente en función de los intervalos definidos.

Si quisiera emplearse esta misma máquina para clasificar manzanas de tipo Starkin, que son rojas, la máquina se limitaría a enviar todas las manzanas a la caja de rechazos pues ninguna de ellas es amarilla. Este problema puede solucionarse de forma sencilla si la máquina es reprogramable. Para ello, bastaría que la persona encargada del mantenimiento del sistema modificara los parámetros correspondientes al color para que, así, la máquina pudiese clasificar las manzanas Starkin por su tamaño. Gracias a esta posibilidad se dice que un robot es una máquina multifuncional

y reprogramable, características que lo diferencian de cualquier otra máquina y lo hacen más potente y versátil.

Como ya ha podido observarse hay dos acciones importantísimas en todo proceso de control, las entradas y las salidas. Se denominan entradas los datos aportados por los sensores de la máquina y se llaman salidas las acciones o efectos que se ejecutan por parte de los actuadores.

En este curso el ordenador se va a utilizar como herramienta de control de las entradas y las salidas y, además, el ordenador será siempre el depositario de los programas de control de las máquinas. Los programas de control deben diseñarse en alguno de los lenguajes de programación existentes; en este curso, como ya se ha señalado, el lenguaje que se utiliza es Logo. En esta misma unidad didáctica se ha insertado una sección cuyo objetivo es introducir los principios de la programación en Logo. Conviene tener presente que un lenguaje de programación, en principio, está diseñado para un propósito general, aunque partiendo de los lenguajes que existen se pueden conformar conjuntos de instrucciones destinadas a un objetivo específico. Un ejemplo de ello es un conjunto de órdenes cuya ejecución permita hacer un dibujo. En Logo el conjunto de instrucciones y comandos referentes al control de las máquinas se conoce como Micromundo de Control; en ese micromundo no se utilizan todas las órdenes que componen el lenguaje Logo, sino que el uso se limita a aquellas que son necesarias para regular el funcionamiento de los operadores y los sistemas técnicos. Por ello, el apartado II, "Fundamentos de Logo" no es un manual del lenguaje de programación sino una introducción al método de programación y al estudio de las posibilidades del lenguaje, mediante el uso de las órdenes que son imprescindibles para la comprensión de Logo.

El objetivo que se persigue en esta unidad es la adquisición de conocimientos suficientes para poder crear procesos capaces de controlar máquinas y sistemas y diseñar máquinas que puedan ser controladas mediante el ordenador.

El análisis de cualquier máquina a la que pretenda aplicar un sistema de control permite establecer que debe cumplir determinadas características que permitan su robotización y que serán comunes al resto de las máquinas controlables por robots.

Estas características son:

- a.- Realizan las acciones para las que están diseñadas a partir de una información que puede denominarse de "entrada".
- b.- De todas ellas, se espera que produzcan algún efecto: trasladar objetos, desplazar cargas o actuar sobre circuitos. Efecto que puede denominarse "salida".

Piénsese en cualquiera de las máquinas construidas hasta el momento en el curso y se podrá comprobar que lo anteriormente expuesto es cierto. Deteniéndose aún más puede pensarse en máquinas cualesquiera que siempre cumplen las condiciones de una máquina robotizable. Una lavadora tiene unas entradas (mandos y programador) y realiza unas determinadas acciones (toma agua, aclara y centrifuga). El más simple de los circuitos eléctricos puede servir de modelo, donde el interruptor representa la entrada y la bombilla encendida representa la salida.

Controlar una máquina supone identificar las entradas y salidas necesarias que le permitan cumplir la función para la que fue diseñada. Ahora bien, entre las entradas y las salidas de una máquina robotizada, debe disponerse de un sistema lógico que tome las decisiones oportunas; por ejemplo, un ordenador dotado con un programa adecuado. El uso del ordenador no es estrictamente necesario, si bien puede simplificar las operaciones a realizar.

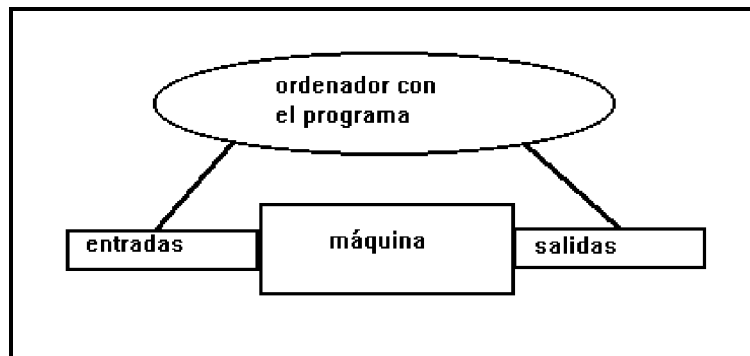


Si no has tenido dificultades, sigue adelante. En caso contrario, ponte en comunicación con la tutoría y comenta tus dificultades.

3.- NECESIDAD DE UN LENGUAJE DE PROGRAMACIÓN

Falta pensar cómo puede el ser humano comunicarse con el ordenador para introducir las órdenes necesarias que permitan controlar una máquina; para ello se utilizará un lenguaje de programación. En las unidades didácticas se utiliza Logo. Aprovechando las características del lenguaje Logo, se puede desarrollar un algoritmo que permita controlar una máquina. En este lenguaje, como en cualquier otro, es necesario definir o usar instrucciones o comandos que permitan elaborar un programa capaz de controlar una máquina. Para poner en marcha un motor, habrá que usar una serie de instrucciones, teniendo en cuenta que los movimientos de un motor son el giro en un sentido y en el contrario y, por supuesto, la parada. El conjunto de órdenes que permiten gobernar las máquinas se denomina "Micromundo de Control". La denominación "Micromundo" obedece a que, para controlar una máquina, basta con utilizar un reducido número de órdenes del lenguaje de programación; tan sólo son necesarias aquellas que regulen el funcionamiento de los operadores y sistemas técnicos, sobre todo las que controlan las entradas y salidas; por lo tanto, un micromundo será el subconjunto de órdenes o comandos propios del lenguaje de programación que son utilizados para definir nuevas órdenes.

El cuadro adjunto muestra una aproximación a la interrelación entre el ordenador y las entradas y las salidas de una máquina.



Las salidas son, generalmente, motores que mueven la máquina a las que están conectados. Otros operadores que, habitualmente, constituyen una salida del ordenador son las bombillas, los electroimanes, los relés y las electroválvulas.

Las entradas tienen por objeto captar la información que el ordenador necesita para conocer el estado de la máquina. A partir de los datos de entrada, el programa toma decisiones referentes a las próximas actuaciones de la máquina. Las entradas están constituidas por sensores como los interruptores fin de carrera, los potenciómetros o las fotorresistencias.

En el lenguaje informático es necesario ponerse de acuerdo en un vocabulario básico para hacer posible la comunicación, como ocurre en cualquier otro tipo de lenguaje humano; por ejemplo, si se desea actuar sobre un motor, una de las palabras utilizadas para controlarlo podría ser MOTOR y, si además fuese necesario definir su sentido de giro, podría emplearse la palabra DERECHA o IZQUIERDA. En el "micromundo" de control se usa el término MOTOR1 "I" para hacer que el motor gire en un sentido y MOTOR1 "D" para que lo haga en sentido contrario; para parar el motor se ha elegido la expresión MOTOR1 "P".

Antes de profundizar en el lenguaje de programación, se van a definir algunos conceptos importantes. La diferencia entre entradas y salidas debería estar clara con lo expuesto, pero conviene señalar que existen dos tipos de entradas, las digitales y las analógicas. Son entradas digitales aquellas que admiten sólo los valores 0 y 1; por ejemplo, un interruptor es un modelo típico de entrada digital, pues sólo tiene dos estados o posiciones, abierto y cerrado; a la posición abierto se le asigna el valor 0 y a la de cerrado el 1. Las entradas analógicas admiten varios valores. Un potenciómetro es un ejemplo de sensor analógico ya que tiene múltiples posiciones y, por lo tanto, suministra al ordenador múltiples valores.

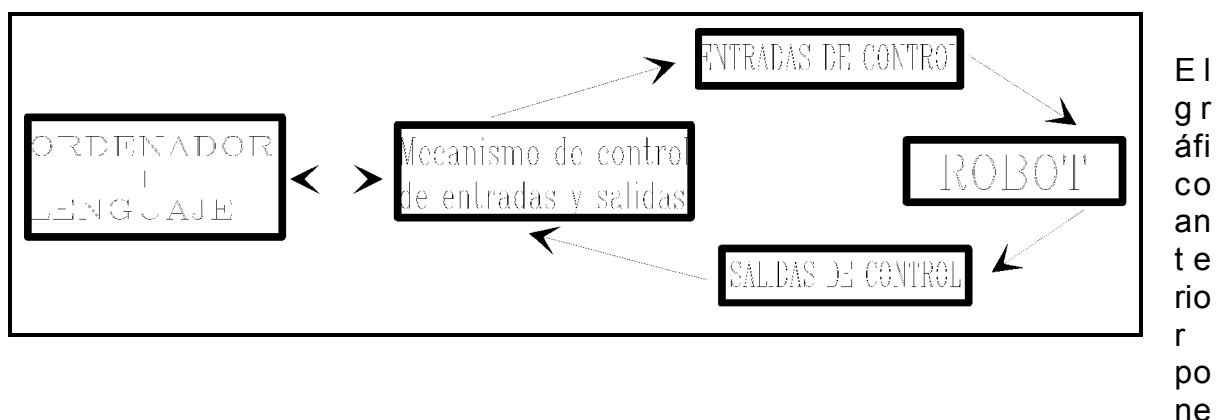
También las salidas pueden ser analógicas o digitales; una bombilla constituye un ejemplo de salida digital, ya que sólo tiene dos estados, encendida o apagada; los motores pueden estar conectados a una salida digital que sólo controlará el giro y la

parada pero, si se conecta a una salida analógica, también podrá controlarse su velocidad de giro. En este último caso, si se opta por una salida analógica, se utiliza la palabra ACELERA seguida de un parámetro para indicar la magnitud de la aceleración, por ejemplo, ACELERA 3.

En el siguiente cuadro se presenta un resumen de las entradas y salidas tanto analógicas como digitales, que más se usan en el control de máquinas y su sintaxis en Logo.

TIPO	VALOR	SALIDAS	ENTRADAS
DIGITALES	Tienen dos valores 0, 1	M1 I D P M2 I D P M4 I D P	SD 1 SD 2 SD 8
ANALÓGICAS	Tienen múltiples valores	ACELERA 3 ACELERA 7 ACELERA 16	SAW SAX ... SAZ

La exposición presentada ha expuesto los conceptos sobre la relación entre el ordenador, el lenguaje de programación y el robot. El esquema definitivo de esta relación es el siguiente:



de manifiesto las relaciones entre los distintos componentes de un sistema robotizado mediante ordenador. Por un lado está el ordenador con un lenguaje y el programa, para tomar datos, decisiones y acciones, que controla todo el funcionamiento del sistema. El ordenador se comunica con la máquina (robot) mediante una tarjeta (controladora) que hace de intérprete de las señales recibidas del robot y traslada

hasta él las órdenes. El robot se comunica con la tarjeta mediante su sistema de entradas y salidas. El ordenador junto con el lenguaje y el programa, la tarjeta controladora y el robot con su sistema de entradas y salidas, conforman un sistema controlado. Un sistema de estas características se denomina robot.

4.- EVOLUCIÓN Y TIPOS DE ROBOT

El desarrollo de los robots ha pasado por distintas fases todas ellas basadas en los elementos de control que intervienen en la máquina, cuya integración ha ido produciéndose a medida que se han mejorado los elementos de control (entradas, salidas, ordenadores y programas). Debido a esas mejoras, las posibilidades y características de los robots han ido variando. A continuación, se presenta una clasificación de los robots según su menor o mayor grado de adaptación al medio, es decir, de su "inteligencia" robótica.

a.- Manipuladores. Son sistemas mecánicos multifuncionales que poseen un sistema de control sencillo, que puede ser manual, y les permite repetir una secuencia de trabajo que en general es fija aunque, en ocasiones, pueda ser variable.

b.- Robots de aprendizaje. Son robots que efectúan una secuencia de movimientos por repetición de los previamente ejecutados por un operador humano. Mediante un sistema mecánico el operador ejecuta los movimientos y el robot los memoriza para ejecutarlos posteriormente de forma repetitiva.

c.- Robots con control. Son robots controlados por un ordenador que transmite las órdenes contenidas en un programa, para que las ejecute.

d.- Robots "inteligentes". Son semejantes a los anteriores, pero además poseen sensores que les permite recibir información sobre el estado del proceso que controlan por lo que pueden tomar decisiones de tipo "inteligente", y realizar un control denominado en tiempo real.

"Tiempo real" es una expresión que se usa para indicar que el tiempo de respuesta de un robot ante la modificación de su entorno de trabajo es mínima; esto es posible gracias a la velocidad de proceso de los ordenadores. Cuando se habla de robots inteligentes, se usa también la expresión "realimentación". Para comprender este concepto se incluye el siguiente ejemplo. Imaginemos que una máquina debe desempeñar su tarea al aire libre, pero que debe desconectarse en caso de agua como medida de protección; para conseguir desconectarla automáticamente, debe disponer de sensores adecuados que detecten la presencia de lluvia y puedan desconectar la máquina hasta que la lluvia cesa, en cuyo caso, la máquina deberá conectarse de nuevo de forma automática. La capacidad de adaptación al medio es posible gracias a la información suministrada por los sensores y a la capacidad que posee el

ordenador para tomar decisiones. Este proceso se llama "realimentación". Un robot dotado de un proceso de realimentación es una máquina capaz de adaptarse a las variaciones del entorno.

5.- ENTRADAS Y SALIDAS

LAS ENTRADAS O SENSORES

De los conceptos presentados, se deduce que la parte más importante de un robot, en su estructura mecánica son sus sensores o entrada. Debido a la importancia de estos elementos, se analizan a continuación sus clases y en qué se fundamentan.

Una máquina realimentada es capaz de detectar su estado general de forma que, a través de sus sensores, el robot puede detectar sobrecargas en su propia estructura o en los motores que lo dotan de movimiento, y reaccionar para impedir deformaciones en la estructura o el recalentamiento de sus circuitos. En las distintas articulaciones y piezas sometidas a tensión y torsión se sitúan sensores que detectan la intensidad e importancia de la deformación de la máquina, por variación en la conductividad de las piezas, y avisan cuando se produce alguna anomalía para que ésta pueda ser corregida.

En ocasiones se dispone de sensores que posibilitan que la máquina detecte su propia posición, hecho éste importantísimo para determinar nuevas trayectorias y desplazamientos. Existen dos tipos de sensores de posición: los ópticos y los magnéticos. Los ópticos están formados por un disco que posee perforaciones cada cierta distancia y dos células fotoeléctricas que son excitadas por rayos luminosos entre los que se interpone el disco; a medida que gira el disco solidario a la pieza móvil, las células fotoeléctricas se van activando, lo cual permite conocer el número de giros realizados por la pieza, o bien conocer cuál ha sido el desplazamiento desde la posición inicial. Los sensores de posición magnéticos también están formados por dos piezas, una de ellas es fija y la otra solidaria a la pieza móvil. Las dos piezas están constituidas por bobinas en las que, al ser excitadas eléctricamente, se produce un desfase de una de ellas sobre la otra y consiguientemente, el desplazamiento de las partes del robot. Dicho desfase puede medirse. En las presentes unidades didácticas se utiliza un potenciómetro para cumplir la función de este tipo de sensor. El eje del potenciómetro estará sujeto a la pieza que gire, con lo que la resistencia del potenciómetro variará en función del giro de la misma y será posible determinar el ángulo de desplazamiento entre ambas piezas.

Dentro del grupo de sensores que permiten determinar el estado o posición de las distintas piezas de un robot, son importantes los que informan acerca de la velocidad de actuación de cada una de las partes móviles. En este caso no es necesario

disponer un sensor específico, ya que basta con hacer los cálculos oportunos a partir de los datos aportados por los sensores de posición ópticos o magnéticos.

Un grupo muy importante de sensores es el de los sensores de entorno, los cuales le permiten al robot determinar su distancia respecto a otros objetos, así como conocer sus formas y volúmenes, es decir, adaptarse al medio cambiante. Los sensores encargados de medir la distancia a la que se halla el robot frente a otros objetos están muy desarrollados; tienen una concepción similar a la del sonar, aunque se ha sustituido la función de las ondas sonoras características del sonar, por la de rayos láser. La posición del robot respecto a un objeto se calcula mediante el método de triangulación. Se han desarrollado con éxito, además de este tipo de sensores, otros que determinan la presencia de objetos próximos, tanto sólidos como líquidos; este tipo de sensores se utiliza para evitar la colisión del robot con objetos próximos a él.

LAS SALIDAS O ACTUADORES

El conjunto de dispositivos que permiten la actuación del robot, son las salidas. Estos dispositivos permiten al robot realizar esfuerzos y movimientos; son, en general, motores, bombillas, relés o electroválvulas.

Los motores hacen posible actuar directamente sobre las distintas partes móviles de la máquina. Los relés actúan sobre dispositivos que requieren, para su funcionamiento, una tensión más elevada que la suministrada por el ordenador a través de su circuito secundario. Las electroválvulas abren, cierran o regulan el flujo de sistemas neumáticos o hidráulicos, que actúan sobre partes móviles. Se pueden utilizar, en vez de electroválvulas, simples electroimanes para atraer o repeler cuerpos ferrosos. Todos los actuadores del robot se conectan a las salidas del ordenador.

RECUERDA



- Los sensores del robot se conectan a las entradas del ordenador.
- Los actuadores del robot se conectan a las salidas del ordenador.

IV.- CON NUESTROS ALUMNOS Y ALUMNAS

IV.- CON NUESTROS ALUMNOS Y ALUMNAS

PLANTEAMIENTO DIDÁCTICO DEL TEMA "EL PROCESO DE CONTROL"

1.- PLANTEAMIENTO INTERDISCIPLINAR Y GLOBALIZADO

El tema "procesos de control" debe plantearse en el aula de forma interdisciplinar, es decir, no centrándose de forma exclusiva en la parte tecnológica que conlleva la cuestión del control robótico, sino abriéndose y hasta esforzándose por encontrar las conexiones existentes que tienen los conceptos que se irán presentado, con otros ya conocidos por el alumnado y que ya han sido desarrollados en otros niveles y áreas de la enseñanza.

Es conveniente que se aprovechen los contactos estructurales entre los distintos seminarios del centro educativo para asegurarse de que en áreas como las de Ciencias Sociales, Filosofía y, sobre todo, Ciencias Naturales, se han expuesto cuestiones referentes a la industrialización, los medios de comunicación, los sistemas de gobierno, las facultades del ser humano, los sentidos, el aparato locomotor o el sistema nervioso entre otros temas. Dichas cuestiones van a ser comentadas con el grupo de alumnos y alumnas cuando se introduzca el tema del control.

La exposición del tema del proceso de control se referirá permanentemente a la estructura interna del ser humano en lo referente a sus actos reflejos y voluntarios. No debe perderse de vista que, si se compara el comportamiento humano que implica procesamiento de información con los procesos de control de dispositivos físicos, se tiene el propósito de facilitar al alumnado la comprensión de procesos complejos que presentan analogía -no identidad- con los comportamientos propios del hombre.

Podrá comenzarse a plantear este tema con los alumnos observando cómo los descubrimientos del hombre han partido, en la mayor parte de los casos, de la observación de la Naturaleza de la que el hombre es un componente fundamental y de donde ha tomado algunos modelos para "reinventar" soluciones a diversos problemas que la Naturaleza ya había dado. Los medios de locomoción, el vestido, e incluso inventos modernos como el radar, tienen su base en la observación y reflexión sobre fenómenos naturales.

En esta ocasión, la observación versará sobre procesos de la propia realidad humana, tanto físicos (el funcionamiento de la musculatura y el esqueleto, que forman el aparato locomotor y ejecutan movimientos similares a los de las palancas y los cilindros compresores; los sentidos perciben cambios en el exterior) como psíquicos, ya que el

sistema nervioso central y periférico reciben información, la elaboran y, por medio de ciertas conexiones, envían ordenes al aparato locomotor.

El planteamiento inicial del proceso de control debe ser globalizador e interdisciplinar, aunque para desarrollar este tema debe seguirse un procedimiento analítico y reflexionar sobre las fases del proceso:

- 1.- La recepción de información.
- 2.- El procesamiento de información.
- 3.- Actuación.

Es necesario ayudar al alumnado para que comprenda que, para que en un ser humano actúe ante un estímulo cualquiera, ha debido producirse un proceso con las siguientes fases:

- 1.- Ha recibido información por medio de los sentidos.
- 2.- Ha interpretado correctamente y analizado esta información por medio del sistema nervioso central, lo que le ha permitido tomar decisiones basadas en esos datos y en sus facultades y limitaciones.
- 3.- Su aparato locomotor realiza determinados movimientos como respuesta a la situación que los sentidos le han planteado y el sujeto es consciente de las acciones que está ejecutando mientras se producen. Ello le permite modificar sus respuestas si cambia alguno de los datos que sucesivamente van aportando los sentidos.

Es importante transmitir al alumnado que el análisis que acaba de hacerse muestra sólo un esquema de nuestros comportamientos, donde se han omitido consideraciones en favor de la claridad con que se quiere presentar el análisis. Pero no debe olvidarse que la simplificación que se ha introducido en procesos tan complejos, si no se tiene presente puede llevar también a una comprensión inadecuada del comportamiento motor.

Estos mismos pasos que acaban de señalarse han sido imitados por el ser humano en el diseño y construcción de un robot, de forma que las fases de actuación de éste son las siguientes:

- 1.- Los sensores captan los cambios que se producen en el exterior de la máquina.
- 2.- El ordenador recibe esta información y, con un programa idóneo, prepara una respuesta teniendo en cuenta todos los datos recibidos.

- 3.- Por medio de conexiones, pone en marcha los actuadores que mueven una estructura. Estos movimientos no son "ciegos", sino que el ordenador sigue recibiendo información de los sensores mientras se ejecuta la orden dada de forma que, si se produce algún cambio durante la ejecución que modifique la situación inicial, se puede modificar dicha orden y reconducir la actuación llegándose hasta la paralización del sistema de no tener alternativas válidas para lograr el objetivo pretendido.

2.- La recepción de información.

Siempre que hablemos de control tenemos que plantear al alumnado la necesidad de poseer información de lo que pasa en el entorno.

En el caso de los animales, (los tropismos vegetales no son ejemplos válidos por su extremada lentitud), los sentidos van aportando información sobre los cambios que se producen en el exterior (o también en el interior por medio del dolor), de una forma especializada; así, el ojo sólo capta sensaciones luminosas que el oído, por ejemplo, no puede captar. Sólo en fenómenos muy especiales, un sentido puede captar sensaciones producidas por estímulos no idóneos (sensaciones táctiles que producen efectos luminosos, como un golpe en el ojo que produce la sensación de "ver las estrellas").

Durante el desarrollo del tema habrá que buscar el recurso que estimule la participación del alumnado por medio de preguntas sobre fenómenos ya conocidos, de forma que las conclusiones del análisis se acerquen a los planteamientos aquí expuestos. Ejemplos sobre los electrodomésticos de uso habitual en el hogar pondrán al descubierto sistemas de control cercanos a la experiencia de los alumnos.

Para asegurar la comprensión de las funciones que cumplen los sensores de un proceso robótico, habrá que ayudar al alumnado a deslindar la frontera que separa un "sensor" de un contacto tipo interruptor. La cuestión es fundamental y no siempre fácil de distinguir. La función de un interruptor supone la puesta en comunicación o la ruptura de un flujo que no puede detenerse, como la apertura del caudal de una acequia que permite la salida del agua de forma natural e irreversible. El efecto es inmediato y único, sin posibilidad de modificación en el momento en que se produce ni tampoco a lo largo de la ejecución.

La actuación de un sensor en un robot supone la captación de una información que no produce un efecto por sí misma, sino que aporta tan sólo un dato que, junto con otros, se incluye en un programa que lo valora y produce la acción. Acción que puede ser revisada continuamente mientras se ejecuta y que puede variarse en función de las sucesivas informaciones que vayan apareciendo.

Deberemos, por tanto, distinguir entre el interruptor de puesta en marcha de un transistor, las sucesivas fases del programa de una lavadora, el sensor de fin de recorrido de una cinta magnetofónica y la célula fotoeléctrica del circuito de alumbrado público.

Continuando con el símil entre el funcionamiento de los sentidos y los datos aportados por los sensores en un robot, convendría analizar, con los alumnos y alumnas, el hecho de que todos nuestros sentidos pueden estar aportando datos al cerebro en un momento determinado, pero solamente se analizan los datos relacionados con la acción que se debe realizar para tomar una decisión. Si tenemos que hacer una suma, no influirá en el resultado el color del bolígrafo usado ni el tamaño del cuaderno en el que se realiza la operación.

En el caso de las máquinas robotizadas, habrá que asegurar que los sensores integrados en ellos sean idóneos y fiables. Un móvil dirigido por la luz reflejada en una línea pintada en el suelo modificará su recorrido por haber entrado en una zona donde hay bajas temperaturas, a no ser que en el programa diseñado para controlar el móvil, además de una célula fotoeléctrica, se coloque una NTC que capte las diferencias de temperaturas producidas en su entorno.

La aportación de ideas sobre distintos tipos de máquinas puede ir acompañada de actividades que ayuden al alumnado a fijar estas ideas y crear una dinámica de equipos más participativa, como las que se proponen a continuación.

Confección de murales conteniendo máquinas automáticas, programables, "ciegas", con sensores...

Elaboración de informes sobre robots con pies de foto, que incluyan explicaciones sobre las acciones que realizan y los sensores utilizados.

Elaboración de comentarios críticos sobre textos literarios o guiones cinematográficos que aborden temas sobre androides o máquinas "inteligentes".

Pase comentado del vídeo del National Geographic que trata del tema de ROBÓTICA. Trabajo en grupos y elaboración de resoluciones sobre alguno de los variados aspectos que se tratan en dicho video sobre la ROBÓTICA.

3.- El procesamiento de información

Para plantear esta fase del proceso de control, se puede comentar con el alumnado el conjunto de actos voluntarios que realizamos a lo largo del día y que requieren la recogida de datos y la toma de decisiones, a la que se denomina procesamiento de información.

Se insistirá en la necesidad de que los datos sean fiables, de que los sentidos proporcionen datos correctos y precisos como base imprescindible para poder tomar decisiones adecuadas.

Una vez superada esa primera fase, se deberá disponer de criterios adecuados para que el procesamiento de los datos lleve a la mejor decisión posible puesto que, en el mundo de lo humanamente posible, la decisión correcta está en función de factores ambientales y culturales que la hacen relativa (por ejemplo, descalzarse para entrar en una iglesia es una decisión incorrecta, pero en una mezquita es la conducta institucionalizada por las normas propias de la sociedad musulmana).

En términos informáticos, dicha relatividad no aparece reflejada, pues los criterios que incluiremos en el programa de control serán algoritmos para dar respuestas preestablecidas en cada una de las encrucijadas que se irán produciendo conforme los datos recibidos por los sensores se vayan valorando. En unos casos será un Sí o un No lo que hará que el programa vaya en una dirección u otra; en otros, será el valor numérico de un dato, mayor o menor que una cantidad utilizada como referencia; pero, en todo caso, será siempre la comparación de los datos obtenidos con las condiciones incluidas en el programa lo que hará que se realice una acción u otra.

En este punto conviene proponer a la clase que establezca la distinción entre las actuaciones de los animales, los actos reflejos, los automatizados y los actos humanos voluntarios fruto de una reflexión consciente, para establecer los paralelismos oportunos con los programas informáticos que deciden las actuaciones de los robots.

La irracionalidad de las actuaciones de los animales, dirigidas únicamente por el instinto innato e inmutable, tiene su parecido con la estructura fija de un programa informático. La rapidez de un acto reflejo es similar tanto a la velocidad en la llegada de los datos al ordenador como a la de la ejecución del programa que inmediatamente prepara una orden para un actuador. Muchos actos realizados en nuestra vida diaria están automatizados y memorizados en el cerebelo; de este modo, actuaciones más complejas que las de un acto reflejo como escribir, leer, andar o conducir un vehículo, no suponen un desgaste y una atención continuada y permiten realizar simultáneamente otras tareas; la similitud con la repetición de los automatismos de un robot es evidente. Por último, la necesidad de recibir información fiable, de valorarla

en función de unos parámetros para llegar posteriormente a una actuación, que puede ser la de no moverse, es comparable al proceso interno de un programa informático.

Para afianzar la adquisición de los conceptos básicos vinculados con el de control robótico, pondremos al alumnado en situaciones de su vida diaria en las que suele tomar decisiones: ir al cine.... Les haremos reflexionar tanto sobre la dificultad de decidir sin disponer de los datos adecuados: relación de películas en cartel, como de hacerlo sin disponer de un método adecuado para obtenerlos: consultar el periódico, o el teletexto; se insistirá en la necesidad de analizar y valorar la información disponible: si es día de la semana festivo, laborable, o día del espectador; si se tiene tiempo libre, o unos ahorrillos.

Una reflexión de este tipo tiene que llevar a los alumnos y alumnas a captar la complejidad de actos tan sencillos, aparentemente, como ir al cine, puesto que al proceso anteriormente expuesto se seguirán sumando datos y variables como: el clima. la distancia al cine, los medios de transporte disponibles, etc.

Se aprovecharán estas situaciones para hacer notar la necesidad de analizar los problemas complejos y dividirlos en partes más sencillas hasta llegar a resolverlas una por una de modo que, sin perder la conexión con el todo, la suma de las respuestas a cada uno de los aspectos parciales de estos problemas se integren en la solución del problema planteado y, por tanto, la ejecución de la decisión correcta.

4.- Actuación

Las actuaciones llevadas a cabo por un robot controlado por un programa se asemejan también a los actos volitivos en que no llevan tan sólo a un acto puntual y breve en el tiempo, sino que pueden ser complejos y dilatados en el tiempo de su ejecución.

Ejemplos como el anteriormente expuesto de ir al cine requieren de una serie de fases no sólo para la toma de la decisión, sino para su ejecución: vestirse con ropa adecuada, salir de casa a la calle, elegir el medio de transporte adecuado, comprar la entrada, etc. Fases en las que pueden aparecer variables que modifiquen nuestra actuación final con la aparición de hechos como: no tengo ninguna blusa planchada, se ha estropeado el ascensor, hay huelga de autobuses urbanos, sólo hay localidades de la fila uno; son datos que la experiencia aporta a lo largo de las actuaciones parciales y que nos deben hacer modificar nuestras actuaciones parciales con el fin de conseguir el objetivo final.

El alumnado aportará soluciones a cada uno de estos problemas de modo que surjan opciones como: pido prestada una camisa a mi hermano, bajo por las escaleras, paro un taxi; cada una de las cuales puede ir acompañada de las alternativas que se quiera, incluyendo la de no ir al cine.

El análisis precedente sirve de base para asimilar los conceptos vinculados con la toma de decisiones y el flujo de actuaciones que aquella genera.

Un programa de control actuará de modo similar. Su trabajo no termina después de analizar los datos obtenidos por los sensores y elegir la respuesta precisa; durante la ejecución del movimiento decidido, estará analizando las variaciones que se vayan produciendo en el proceso de ejecución de modo que, si se produjera algún cambio en la información de entrada aportada por los sensores, no debe proseguir sin modificaciones. Si fuera preciso el programa podría detener su ejecución y mandar un mensaje para que quien estuviera a cargo de la máquina modificara las condiciones externas o internas que impiden la realización del acto programado.

La comprensión de estos procesos exige capacidad analítica en niveles altos de abstracción. Por ello, conviene abordar su tratamiento en el aula analizando los procesos en la concreción de las máquinas familiares en el entorno cultural y técnico del alumnado.

Máquinas como las soldadoras de una cadena de montaje de coches son muy conocidas y pueden ilustrar la estructura del proceso de actuación en las máquinas controladas por ordenadores. Comentaremos su funcionamiento con el alumnado apoyándonos en fotografías o vídeos donde pueden aparecer sensores que localizan la chapa y no la sueldan si no detectan su correcta colocación; sensores que captan la dureza del metal antes de soltar la descarga e impiden accidentes ante la presencia de los operarios que trabajan cerca de estos robots; sistemas de control de la longitud de electrodo disponible, que no sueldan cuando éste se ha gastado y exigen, por medio de una alarma o un piloto, la presencia del operario que alimente la máquina.

Podemos también proponer la reflexión con máquinas más sencillas pero más cercanas a la experiencia de los alumnos, en las que los sensores paralizan el proceso iniciado cuando el entorno o el interior de la máquina ha variado y el mantenimiento de la orden dada podría causar una avería. Los sensores para detectar el nivel de aceite en el motor de un coche, el atasco de una hoja en una fotocopidora o el recalentamiento del motor de arrastre de un vídeo por el excesivo rozamiento de una cinta averiada, pueden ser ejemplos sencillos que ayuden a comprender y valorar la necesidad de que los programas de robótica tengan un sistema de observación y vigilancia del interior y exterior de la máquina que actúe durante el proceso de ejecución.

5.- Transferencia de los conceptos propios del proceso de control al ámbito del trabajo grupal

La reflexión sobre las fases necesarias en el diseño del flujo de las acciones de control:

TOMA DE DATOS ---> PROCESAMIENTO DE LA INFORMACIÓN ---> ACTUACIÓN

debería servir de motivación para valorar la dinámica de trabajo en los grupos del aula, por la conveniencia de su utilización en áreas como la tecnológica.

En los trabajos que el profesor o profesora vaya proponiendo y que el alumnado irá realizando a lo largo de sucesivos cursos lectivos, se analizarán los proyectos antes de ejecutarlos, tanto si son cerrados, abiertos o libres. Esto requerirá que los alumnos estén dispuestos para recibir la información necesaria desechando datos innecesarios y tomando datos relevantes. Posteriormente se organizarán en grupos para diseñar el proceso que permitirá llevar a cabo el proyecto, analizar las distintas posibilidades de solución, buscar salidas a las dificultades que encuentren, en suma, tomar decisiones entre todos después de haberlas analizado y valorado. Como las actuaciones serán complejas y se realizarán dentro de un proceso, conforme vayan encontrando dificultades, los grupos irán recabando más información: consultarán libros, al profesorado, a los propios compañeros, y volverán a procesar estas informaciones para llevarlas por último a la práctica y alcanzar el objetivo propuesto.

Como puede observarse, los aspectos básicos del proceso de control se encuentra también en algunas formas de interacción de los grupos de pares. Por ello, conviene que el profesor haga explícita la semejanza antes apuntada para mejorar el proceso interno del trabajo grupal, haciendo meditar a sus alumnos y alumnas sobre las ventajas de controlar los propios procesos de actuación del grupo y crear hábitos de práctica reflexiva.

Conviene plantear cualquier trabajo sobre control pensando en las tres fases citadas anteriormente.



Diseña una sesión de trabajo para tus alumnos y alumnas, donde se plantee el tema del control. Selecciona cuidadosamente el sistema técnico cuyo análisis propondrás para que el alumnado reconozca y asimile los conceptos básicos de control. Envía el diseño a la tutoría.



El planteamiento didáctico del tema del control robótico supone proponer al alumnado proyectos que le exigen:

- Determinar con claridad el proceso del sistema técnico: el objetivo que se propone.
- Establecer cuáles son los datos necesarios.
- Decidir cómo procesar la información y plantear las alternativas del flujo de acciones, es decir, crear procedimientos.

V.- ENTRE MÁQUINAS Y HERRAMIENTAS

V.- ENTRE MÁQUINAS Y HERRAMIENTAS

MICROMUNDOS DE CONTROL

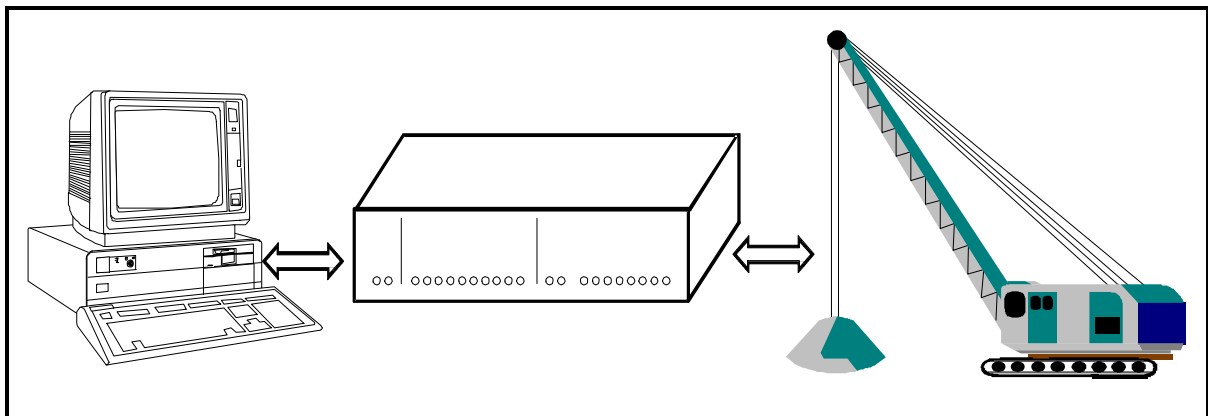
1.- Control externo de una máquina mediante ordenador

RECUERDA



Para controlar una máquina son necesarios:

- Un sistema programable que gestione las señales que provienen de una máquina y actúan sobre ella (ORDENADOR + PROGRAMA).
- Una interfaz o dispositivo que adecue las señales de la máquina a los dispositivos de entrada y salida del ordenador (TARJETA CONTROLADORA).
- Una máquina que haya que controlar con sus sensores y actuadores.



a.- El ordenador. Procesado de la información.

Un ordenador es una máquina de propósito general para procesar información.

La información se introduce en el ordenador por los puertos de entrada, mediante periféricos como el teclado, el ratón o determinados dispositivos, que actúan como interfaz o adaptador entre el ser humano (u otra máquina) y el ordenador. Estos sistemas de entrada se denominan unidades de entrada.

La adaptación de la información por medio de la interfaz es necesaria para adecuar a los impulsos digitales del ordenador los formatos de información inteligibles por el ser humano o para adecuar las señales que proporcionan los sensores de una máquina a los impulsos digitales.

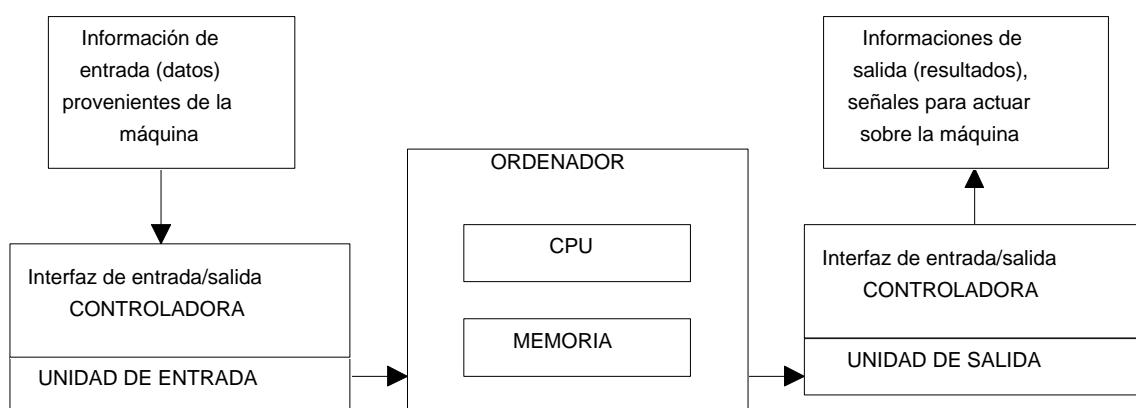
La información con la que trabaja el ordenador está constituida por datos e instrucciones (programas) y es almacenada temporalmente, mientras dura su procesamiento, en un soporte de almacenamiento denominado memoria.

El ordenador, mediante las instrucciones de un programa, procesa los datos y elabora unos resultados. Esta operación es realizada por el microprocesador o CPU (unidad central de proceso).

Los resultados del procesamiento se obtienen por los puertos de salida, mediante periféricos como la pantalla, la impresora, o determinados dispositivos que actúan de interfaz o adaptador entre el ordenador y el ser humano u otra máquina. Estos sistemas de salida se denominan unidades de salida.

En muchos casos, los sistemas que actúan de interfaz pueden realizar las dos funciones de adaptación existentes: Desde el hombre o una máquina, hacia el ordenador y viceversa.

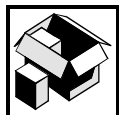
En el siguiente gráfico se resumen las ideas expuestas; las flechas marcan las direcciones que son recorridas por las señales o informaciones.



b.- El equipo de control

Como ya se ha dicho, el ordenador necesita un programa para procesar específicamente determinadas informaciones.

Los programas se elaboran mediante un lenguaje de programación; en este curso se ha usado el lenguaje WinLogo por su sencillez, porque las instrucciones se escriben en español, y por permitir la construcción de un vocabulario específico -denominado micromundo de control- que permite recibir y enviar órdenes a las máquinas de forma natural mediante la controladora.



La controladora se describe en la unidad didáctica XIII de este curso.

Para poder trabajar con WinLogo, como funciona bajo el sistema operativo Windows, se necesita un ordenador que disponga de un microprocesador 80386 u otro superior, disco duro, pantalla y adaptador gráfico VGA en color, 4 Megabytes o más de memoria RAM y ratón.

El ordenador recibe los datos o señales a través de los canales de entrada (puertos de entrada) mediante periféricos de entrada, y emite los resultados o señales a través de canales o puertos de salida mediante periféricos de salida, una vez que ha sido procesada la información.

Los puertos de entrada y salida a través de los que actúa la controladora utilizada en este curso, son los puertos LPT1 y LPT2, denominados también puertos paralelos. Las impresoras se conectan, generalmente, a estos puertos mediante conectores denominados DB25 (de 25 patillas). La tarjeta controladora se conectará también con el DB25. El puerto paralelo que se desea utilizar (LPT1 o LPT2) puede seleccionarse a través de un programa, normalmente se usa LPT1, aunque también puede seleccionarse el LPT2.

2.- La automatización y la robotización

El conjunto constituido por una máquina y su sistema de control constituye un autómeta; si el sistema de control es programable se denomina autómeta

programable y esto permite que la máquina realice diferentes tareas según el programa empleado.

Se denominan autómatas programables secuenciales a los autómatas programables que disponen de memoria y de sensores que informan del estado interno de la máquina, por lo que puede utilizarse dicha información para modificar su estado de acuerdo con el programa almacenado en la memoria y realizar así tareas más complejas. En estos autómatas, los estados de la máquina son función temporal de los estados anteriores, por lo que puede decirse que son autómatas retroalimentados.

Si este tipo de autómatas dispone, además, de sensores que controlen el entorno de la máquina, entonces podrá actuar en función de los estados anteriores y del entorno y sus variaciones; en este caso sería más correcto denominar a estos autómatas con el término robot.



En las próximas unidades se realizan y establecen diversas propuestas de automatización y robotización; clasifícalas de acuerdo con las definiciones anteriores.

3.- Actividad: control de un motor. El problema de la interfaz.

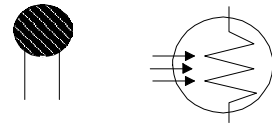
En esta sección se propone una actividad práctica. Se desea usar el ordenador para controlar un motor, haciendo que éste funcione de acuerdo con las cadencias que se establezcan en el programa de control.

¿Qué interfaz se podría usar?. Una posible solución es la de construir una interfaz capaz de leer en la pantalla la información necesaria para controlar el motor. Para ello, puede optarse por leer con una fotorresistencia la intensidad de iluminación de la zona de la pantalla que contenga la información. La débil señal de la fotocélula puede ser amplificada mediante un transistor y utilizada para disparar un relé que cierre o abra el circuito al que está conectado el motor.

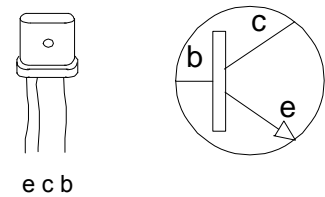
La solución propuesta es algo rebuscada, aunque perfectamente realizable. Por otra parte, se pone de manifiesto la conveniencia de tener interfaces universales, que permitan una conexión cómoda entre el ordenador y las máquinas en diferentes situaciones.

a.- Materiales necesarios para preparar el circuito

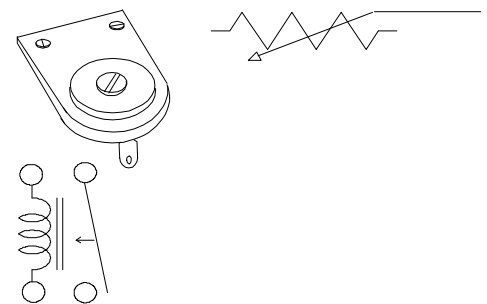
1 Célula fotoeléctrica LDR



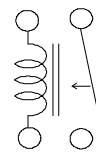
1 Transistor BD175 o equivalente.



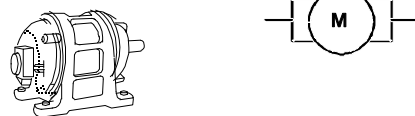
1 Resistencia variable de $1K\Omega$
(para ajustar la sensibilidad de la célula)



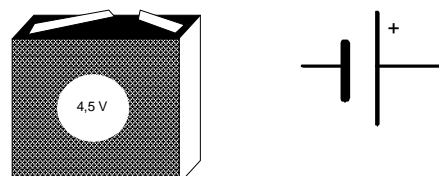
1 Relé (6V-280 Ω) o similar



1 Motor (a 4'5 V)

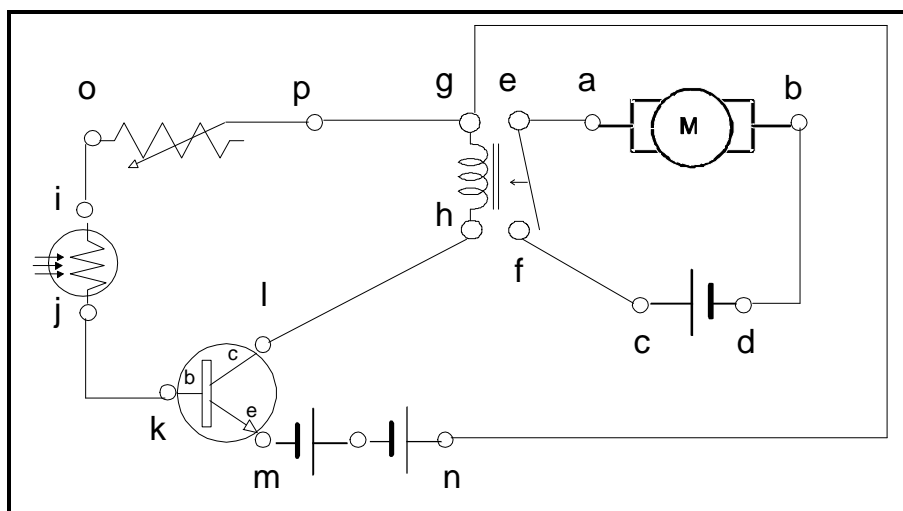


3 Pilas de 4, 5 V



y cable rígido de poca sección.

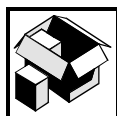
Con estos componentes se puede montar el circuito como el que aparece en la figura.



b.- Montaje y explicación del circuito

Se conecta el motor M con la pila de 4, 5 V, utilizando como interruptor la salida (e f) del relé. Se construye, a continuación, el circuito (a b d c f e).

Se completa el montaje con el circuito (g p o i j k l h) y se conectan las pilas (m n) con la polaridad indicada en la figura anterior.



Consulta el apartado V de la unidad didáctica XIX sobre el transistor para comprender mejor la explicación sobre el circuito que aparece a continuación.

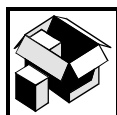
La fotorresistencia (i j) produce una cierta intensidad de corriente cuando la zona de pantalla está iluminada, pero esta intensidad es demasiado pequeña para que el campo que se genera en la bobina del relé (g h) cierre el interruptor (e f).

En cambio la intensidad de la fotorresistencia es suficiente para desencadenar en la base del transistor una corriente amplificada del colector al emisor que abre circuito (l m n g h) alimentado con las pilas (m n), con lo que se consigue que la bobina (g h) del relé cierre su interruptor de salida (b d).

No obstante, para impedir que la intensidad de la fotorresistencia cuando la pantalla esté apagada sea suficiente para abrir el circuito, se coloca la resistencia variable (op) con la que se puede limitar la corriente que llega a la base del transistor.

c.- Programa para controlar la iluminación de una zona de la pantalla del ordenador

La activación de la fotorresistencia tiene lugar cuando la luz de la pantalla incide sobre ella; se puede controlar esta operación con la ayuda del ordenador y el lenguaje WinLogo.



Es conveniente complementar las explicaciones que siguen, con el apartado II de WinLogo, que figura en esta unidad didáctica. Revisa el vídeo de introducción para familiarizarse con WinLogo y su tortuga.

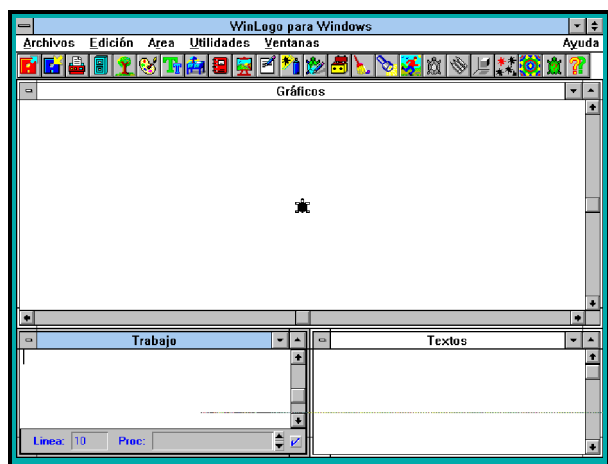
4.- LOGO

Para trabajar con LOGO deben realizarse las siguientes acciones:

- Encender el ordenador
- Cargar Windows
- Ejecutar WinLogo

a.- Ventanas iniciales

Se observan tres ventanas:



Ventana de gráficos: La tortuga puede moverse por ella con órdenes directas o mediante la ejecución de procedimientos (programas). En su movimiento puede dejar tras de sí un trazo, con lo que puede realizarse dibujos.

Esta ventana tiene un sistema de coordenadas euclídeas. El origen de coordenadas (Centro) es el centro de la pantalla; en él se encuentra inicialmente la tortuga. Su eje de abscisas es

horizontal (positivo a la derecha) y el de ordenadas vertical (positivo hacia arriba); las unidades están representadas por los pasos de la tortuga. En cada instante se conoce la posición de la tortuga: rumbo (ángulo, expresado en grados sexagesimales, que forma el eje de la tortuga con el semieje positivo de ordenadas, de signo positivo si el giro se realiza en el sentido de las agujas del reloj), y coordenadas X e Y.

Ventana de trabajo: En ella se ejecutan las ordenes y se definen los procedimientos. Para escribir y moverse por la ventana de trabajo se utilizan las teclas de edición habituales ([Supr], [Del], movimientos de cursores, etc.) y las acciones de edición (cortar y pegar habituales de Windows).

Ventana de textos: En ella, WinLogo informa de los errores o resultados de las operaciones de escritura.

b.- Primeras primitivas gráficas de la tortuga:

Para realizar el control es necesario conocer tan sólo algunas primitivas (comandos de acción del lenguaje).

Borrado de pantalla:

BP Borra la ventana gráfica e inicializa la tortuga en (0, 0) y rumbo 0°.

Desplazamientos:

AV n La tortuga avanza n pasos de acuerdo con la dirección de su eje.

RE n La tortuga retrocede n pasos de acuerdo con la dirección de su eje.

GD n La tortuga gira hacia la derecha n grados sexagesimales modificando su rumbo.

GD n La tortuga gira hacia la izquierda n grados sexagesimales modificando su rumbo

Estado de la tortuga:

OT Oculta tortuga, La tortuga se hace invisible pero sigue allí.

MT La tortuga es visible.

SL La tortuga sube el lápiz por lo que no deja trazo al caminar.

BL La tortuga baja el lápiz por lo que deja trazo al caminar.

Llenado de una figura cerrada:

RELLENA Se introduce la tortuga dentro de una figura cerrada con el lápiz subido para que no deje rastro, se le da la orden BL RELLENA y la figura se rellena con el color del lápiz de la tortuga.

Otras primitivas de control son:

REPITE n [órdenes] Se ejecutan n veces las órdenes (primitivas) que figuran en la lista, entre corchetes.

ESPERA n Transcurren n centésimas de segundo antes de ejecutar la siguiente primitiva.

Los ejercicios siguientes te familiarizan con las primitivas anteriores.

ACTIVIDAD 1



Averigua los pasos que puede dar la tortuga sin salirse de la ventana.

ACTIVIDAD 2

Dibuja un cuadrado de lado 80 y rellénalo de tinta negra.

ACTIVIDAD 3

Dibuja un triángulo equilátero y un pentágono regular.

Sugerencia:

- (¿Cuál es el valor del ángulo de giro que efectúa la tortuga en cada caso al iniciar el trazado de un nuevo lado?)

c.- Modo procedimental:

Para crear un procedimiento se debe escribir en la ventana de trabajo la primitiva PARA (que indica: para definir...) seguida del nombre que se quiera dar al procedimiento.

Se escriben las órdenes (primitivas de la tarea a realizar) y se concluye con la primitiva FIN para indicar el fin de la definición.

ACTIVIDAD 4



Realiza un procedimiento que genere un cuadrado de lado 100 y se rellene de negro. Llama a este procedimiento CUADRONEGRO.

d.- El procedimiento de control

Construido el circuito de acuerdo con las instrucciones del punto b de este mismo apartado 3, se propone que el motor realice un ciclo ininterrumpido en el que esté en marcha 4 segundos y parado 2.

Para ello, se coloca la célula fotoeléctrica del circuito sobre la pantalla con un poco de cinta adhesiva, centrándola en la posición del cuadrado del ejercicio anterior. Cuando la tortuga rellene el cuadrado de negro la célula estará inactiva y el motor parado; al borrar la pantalla, la ventana mantendrá su fondo blanco, la célula se activará y, en consecuencia, el motor entrará en funcionamiento.

Los tiempos de rellenado pueden acelerarse ocultando la tortuga lo que, además, puede influir en la recepción de luz por la célula fotoeléctrica.

Para temporizar cada estado se utilizará la primitiva ESPERA ya descrita: ESPERA 400 y ESPERA 200.

Se denominará al procedimiento CONTROL1. Para que el ciclo se realice ininterrumpidamente se utilizará la recursividad. Se colocará como última orden de este procedimiento la orden CONTROL1, con lo que se inicia de nuevo el procedimiento al llegar a este punto. Se denomina recursividad a la introducción, en la definición de un procedimiento de una llamada a sí mismo.

Definido el procedimiento CUADRONEGRO propuesto por la actividad 4, en la ventana de trabajo se escribe:

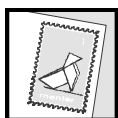
```
PARA CONTROL1
BP ESPERA 400
CUADRONEGRO ESPERA 200
CONTROL1
FIN
```

Al ejecutar CONTROL1 se controla exteriormente el circuito de acuerdo con las pautas pedidas. No debe olvidarse que en WinLogo, para parar un procedimiento recursivo como este, debe pulsarse la tecla [ESC] (Escape), de lo contrario el procedimiento se repite indefinidamente.

ACTIVIDAD RECOMENDADA



Siguiendo el apartado II de esta unidad repasa cómo se reedita un procedimiento, ya que es muy útil para corregir errores de programación. Ejercita también las acciones de almacenamiento y carga de un conjunto de procedimientos. Aplica estos conocimientos para guardar y recuperar desde el disco los procedimientos CUADRONEGRO y CONTROL1.



Envía los procedimientos de la actividad recomendada a la tutoría.



Diseña otros procedimientos para controlar el motor con diferentes secuencias, por ejemplo:

- Para que después de cada segundo parado este encendido durante 1, 2, 3, 4, 5, 6,... s, respectivamente.
- Para que realice exactamente 4 encendidos y 4 apagados de 1 s.
- Para que realice ininterrumpidamente encendidos y apagados de 1 centésima de segundo. (¿Que sucederá?).

VI.- GLOSARIO

VI.- GLOSARIO

- **ALGORITMO:** Es un conjunto de reglas operativas cuya aplicación permite resolver un problema formulado mediante un número finito de operaciones. Un algoritmo debe tener tres propiedades: ser finito, definible y generalizable. Lo característico del algoritmo es que transforma unas cantidades, denominadas magnitudes de entrada, en otras cantidades, denominadas magnitudes de salida, a partir de un conjunto bien definido de instrucciones de transformación. Un algoritmo puede traducirse, merced a un lenguaje de programación, en un programa ejecutable por un ordenador.
- **CONTROL DE PROCESOS:** Consiste en la utilización de un ordenador especializado, denominado controlador de procesos, para controlar un proceso industrial o de fabricación.
- **INTERFAZ:** En informática, es la unión de dos equipos informáticos o de dos aplicaciones de "software" que les permite intercambiar información mediante la adopción de reglas físicas o lógicas, comunes. Permite la comunicación física y lógica de un sistema informático con elementos exteriores.
- **LDR:** Es una resistencia que varía en función de la luminosidad.
- **MICROMUNDO:** Es un conjunto de comandos y órdenes del lenguaje LOGO relativos a una actividad informática determinada. Por ejemplo, el "Micromundo de la Tortuga" es el conjunto de órdenes y comandos del lenguaje LOGO relacionadas con el uso de la Tortuga; el "Micromundo del control" es el conjunto de órdenes relativas a las entradas y salidas de control aplicables a cualquier sistema técnico.
- **NTC:** Es una resistencia que varía en función de la temperatura.
- **PARÁMETRO:** Información transmitida a un procedimiento o asociada a una primitiva. Por ejemplo en "Avanza 40", el número cuarenta es el parámetro asociado a la primitiva "Avanza".
- **PRIMITIVA:** Es una orden o comando del lenguaje LOGO, definido por el diseñador del lenguaje. Ejecuta una acción de manera inmediata y puede necesitar parámetros o no. Por ejemplo, "Borrapantalla" es una primitiva que no necesita ningún parámetro; "Avanza 67" y "Escribe "gato"" son primitivas que necesitan parámetros para poder ser ejecutadas.
- **PROCEDIMIENTO:** En informática, es la sección de un programa que realiza acciones (cálculos fijos, control de datos y acciones repetitivas) definidas

completamente con los datos especificados por parámetros. El término "procedimiento" se emplea en el contexto de lenguajes informáticos de alto nivel.

- ROBÓTICA: Conjunto de técnicas utilizadas para el diseño y construcción de robots y la puesta en marcha de sus aplicaciones.

VII.- BIBLIOGRAFÍA

VII.- BIBLIOGRAFÍA

TÍTULO: Inteligencia artificial.

AUTOR: J. M^a Angulo, A. del Moral

EDITORIAL: Paraninfo

ISBN:84-283-1451-9

Aunque se trata de un libro editado ya hace tiempo, en 1986, no por eso deja de ser un buen instrumento para acercarse al mundo de la inteligencia artificial y su relación con la robotización y los lenguajes naturales.

En este libro se ofrece una perspectiva general de los apartados más significativos de la Inteligencia Artificial, como las posibilidades de ella en campos como el de la comprensión del lenguaje natural por parte de máquinas y ordenadores; el de la visión artificial; el de la construcción de robots con un cierto grado de "inteligencia" y el de la creación de sistemas expertos, capaces de sustituir a expertos humanos en áreas concretas de trabajo. En definitiva, se trata de un buen libro para iniciarse en la comprensión de la Inteligencia Artificial y los robots, que ayudará en el desempeño de la función docente.

VIII.- SOLUCIONES

VIII.- SOLUCIONES

ACTIVIDAD 1

Solución:

- Haz que la tortuga avance 100 pasos: Escribe en la ventana de trabajo: AV 100. Pulsa [Intro].
- Borra la pantalla: BP [INTRO].
- Retrocede 100 pasos. RE 100 Pulsa [INTRO].
- Borra la pantalla y gira 90º: BP GD 90. Pulsa [INTRO].
- Avanza 280 pasos: AV 280...

ACTIVIDAD 2

Solución:

Para hacer el cuadrado:

- BP AV 80 GD 90 AV 80 GD 90 AV 80 GD 90 AV 80 GD 90
(El último giro no sería necesario, pero observa que de este modo se repite 4 veces AV 80 GD 90)

Con la primitiva REPITE podrías escribir:

- BP REPITE 4 [AV 80 GD 90]

Para rellenarlo:

- SL GD 45 AV 40 BL RELLENA (¿De acuerdo?)

ACTIVIDAD 3

PARA TRIANGULO
REPITE 3 [AV 60 GD 120]
FIN

PARA PENTAGONO
REPITE 5 [AV 60 GD 72]
FIN

ACTIVIDAD 4

Solución

Escribe en la ventana de trabajo:

```
PARA CUADRONEGRO  
REPITE 4 [AV 100 GD 90]  
SL GD 45 AV 50 BL  
RELLENA  
FIN
```

Si todo ha ido bien en la ventana de texto se leerá el mensaje: Acabas de definir cuadrado. La tortuga no ha realizado ninguna acción, pero WinLogo ya conoce el significado de cuadrado. Para ordenar a la tortuga que lo realice basta escribir en la ventana de trabajo: BP CUADRONEGRO y luego pulsar [INTRO]. Se puede esta palabra como cualquier otra primitiva, incluso dentro de cualquier procedimiento.

