



TECNOLÓGICO
NACIONAL DE MÉXICO



INSTITUTO TECNOLÓGICO DE NUEVO LAREDO

“Con la Ciencia por la Humanidad”

Introducción a la Ingeniería en Sistemas Computacionales y al Diseño Orientado a Objetos

Curso propedéutico

Instructor:

Bruno López Takeyas

bruno.lt@nlaredo.tecnm.mx

10.- DISEÑO ORIENTADO A OBJETOS EN RAPTOR

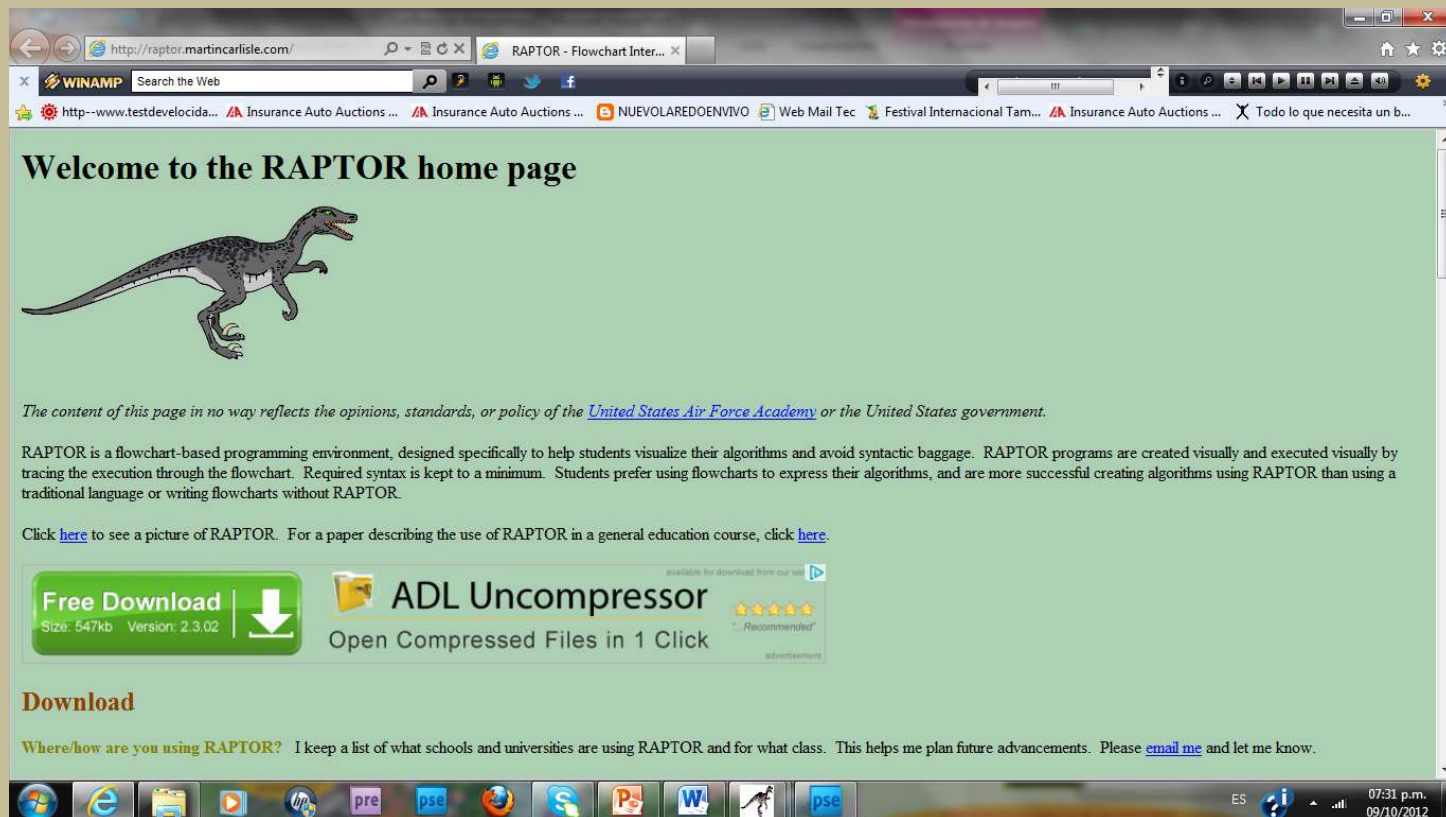
- 10.1 Introducción a Raptor
- 10.3 Símbolos de instrucciones
- 10.4 Variables en Raptor
- 10.5 Entrada de datos
- 10.6 Asignación de datos
- 10.7 Expresiones
- 10.8 Llamadas a métodos
- 10.9 Salida
- 10.10 Estructuras de control

Raptor es un ambiente de diseño de algoritmos basado en la elaboración de diagramas de flujo, diseñado específicamente para que los estudiantes visualicen sus algoritmos evitando el cúmulo de sintaxis.

RAPTOR

Descargar de manera gratuita en:

<http://raptor.martincarlisle.com/>



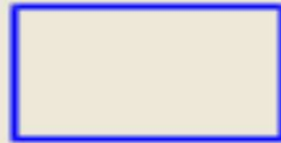
The screenshot shows a web browser window displaying the RAPTOR home page. The browser's address bar shows the URL <http://raptor.martincarlisle.com/>. The page has a light green background and features a large illustration of a dinosaur in the center. Below the dinosaur, there is a disclaimer: "The content of this page in no way reflects the opinions, standards, or policy of the [United States Air Force Academy](#) or the United States government." A paragraph of text explains that RAPTOR is a flowchart-based programming environment designed for students to visualize algorithms. Below this text, there are two links: "Click [here](#) to see a picture of RAPTOR." and "For a paper describing the use of RAPTOR in a general education course, click [here](#)." An advertisement for "ADL Uncompressor" is visible, featuring a green "Free Download" button with a white arrow icon, a file icon, and the text "ADL Uncompressor" and "Open Compressed Files in 1 Click". The advertisement also includes a star rating and the word "Recommended". At the bottom of the page, there is a "Download" section with the heading "Where/how are you using RAPTOR?" and a paragraph: "I keep a list of what schools and universities are using RAPTOR and for what class. This helps me plan future advancements. Please [email me](#) and let me know." The browser's taskbar at the bottom shows various application icons, including Internet Explorer, File Explorer, and several instances of the RAPTOR application (pse). The system tray in the bottom right corner shows the time as 07:31 p.m. on 09/10/2012.

Introducción a Raptor

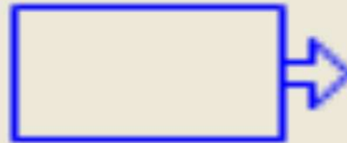
Se recomienda el uso de Raptor por varias razones:

- El ambiente de diseño minimiza la cantidad de sintaxis que se debe aprender al implementar las instrucciones.
- Se diseña en un ambiente visual. Los algoritmos se representan gráficamente mediante diagramas que pueden ser ejecutados símbolo por símbolo, esto ayuda a seguir el flujo del diagrama.
- Facilidad de uso
- Los mensajes de error son fácilmente reconocibles e identificables por diseñadores novatos.
- El principal objetivo es que el estudiante aprenda a diseñar y ejecutar algoritmos sin la necesidad de aprender un lenguaje de programación tal como C++, C# o Java.

Símbolos de instrucciones



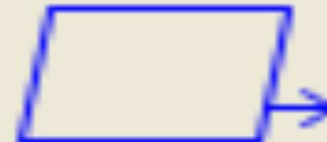
Assignment



Call



Input



Output

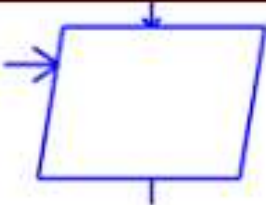
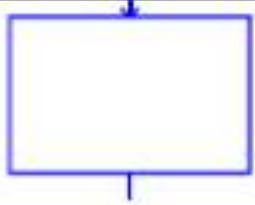
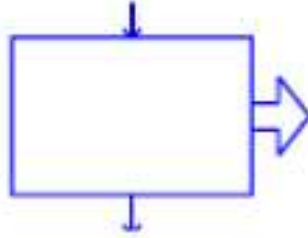
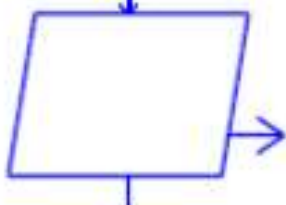


Selection



Loop

Símbolos de instrucciones básicas

<i>Uso</i>	<i>Símbolo</i>	<i>Nombre</i>	<i>Descripción</i>
Entrada de datos		INPUT	Permite al usuario introducir un dato y su valor se almacena en una variable
Asignación		ASSIGNMENT	Cambia el valor de una variable
Llamada a método		CALL	Ejecuta un conjunto de instrucciones definidas en un método
Salida		OUTPUT	Muestra el valor de una variable

Variables en Raptor

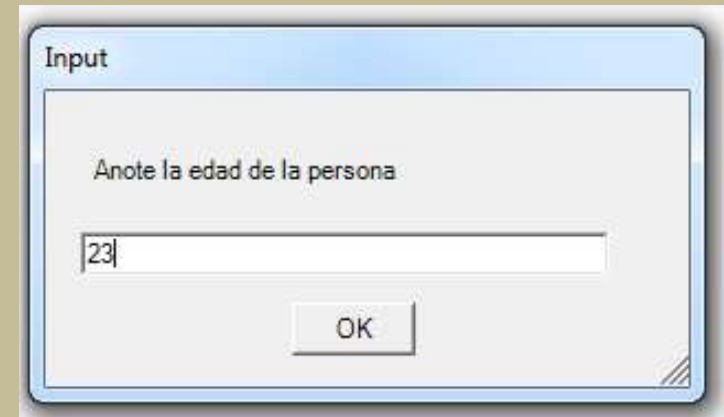
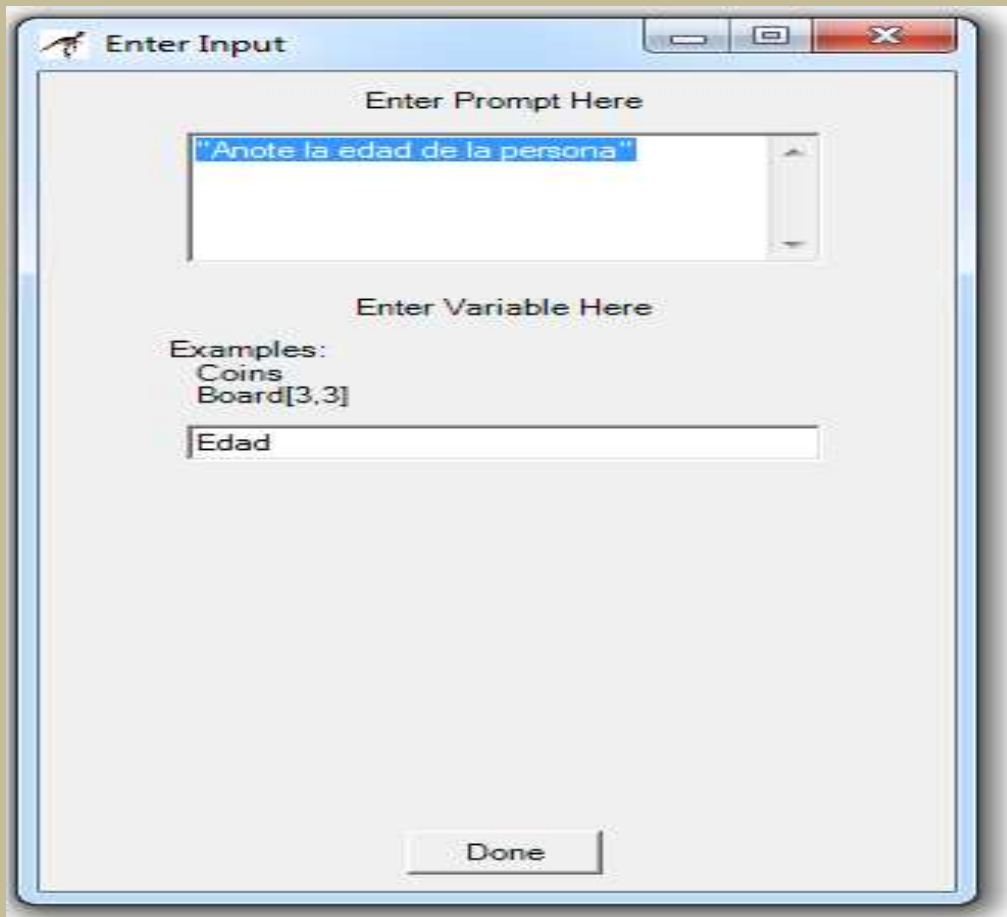
Las variables son localidades de memoria de una computadora para almacenar los valores de los datos

Una variable puede recibir un valor (o cambiarse) de alguna de las siguientes maneras:

- A través de la entrada de un dato por el usuario usando el símbolo **INPUT**.
- Asignándole un valor por medio del cálculo de una ecuación usando el símbolo **ASSIGNMENT**.
- Recibiendo un valor de retorno de un método por medio del símbolo **CALL**.

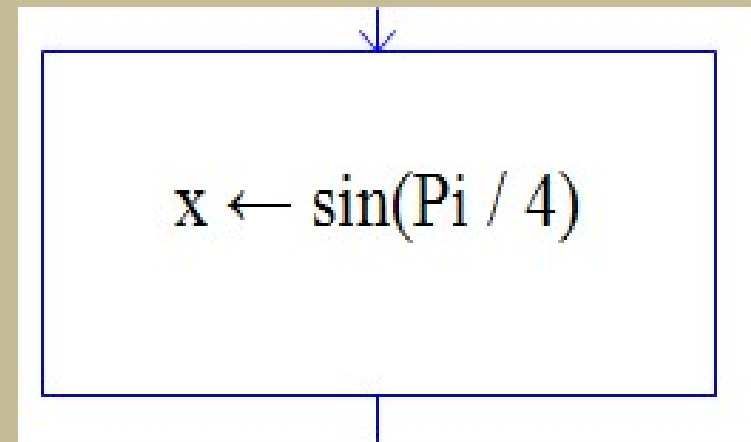
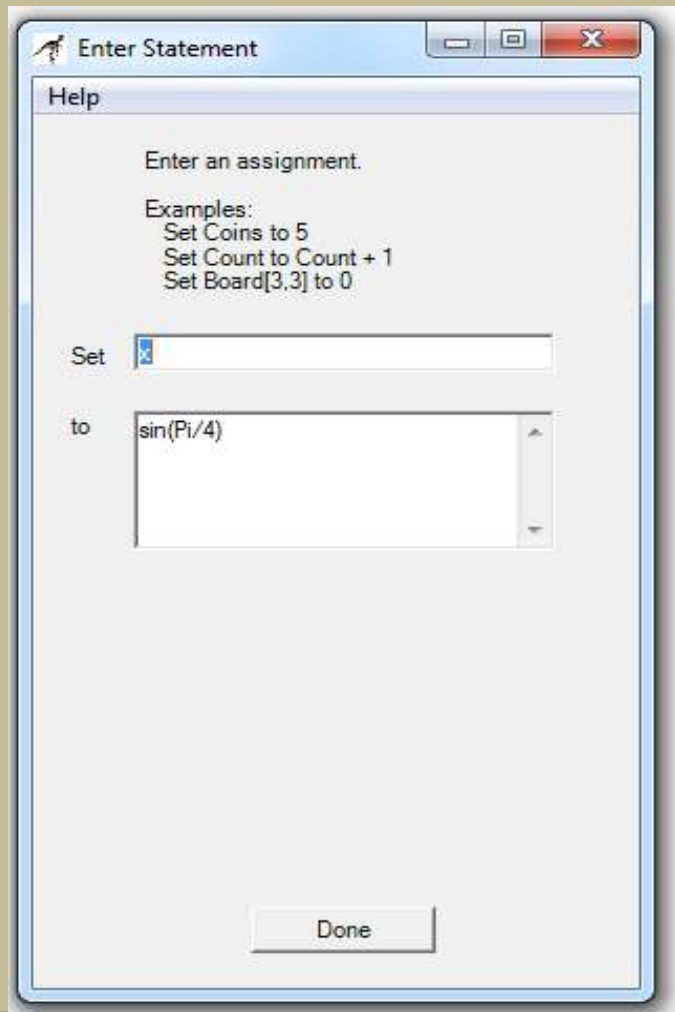
Entrada de datos

El símbolo de entrada (**INPUT**) permite al usuario introducir un valor de un dato y almacenarlo en una variable durante la ejecución de un algoritmo



Asignación de datos

El símbolo de asignación de datos (**ASSIGNMENT**) se utiliza para colocar el valor de un dato en una variable o para realizar un cálculo cuyo resultado se almacenará en la variable definida

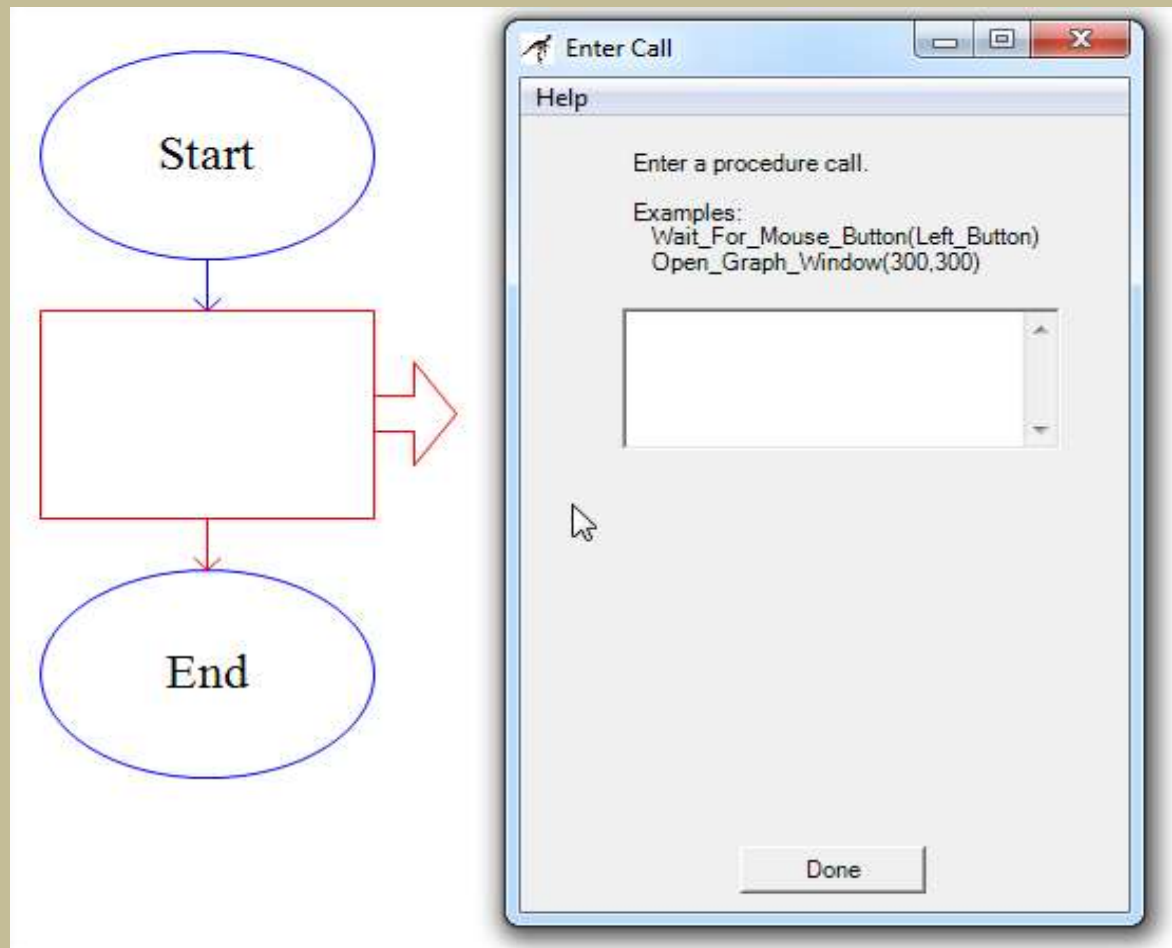


Expresiones

Operación	Descripción	Ejemplo	Resultado
+	Suma	3 + 4	7
-	Resta	3 - 4	-1
*	Multiplicación	3 * 4	12
/	División	3 / 4	0.75
^ o **	Exponenciación	3^4 3**4	81
rem o mod	Residuo	10 rem 3 10 mod 3	1
sqrt	Raíz cuadrada	sqrt(4)	2
log	Logaritmo natural (base e)	log(e)	1
abs	Valor absoluto	abs(-9)	9
ceiling	Redondeo	ceiling(3.1)	4
floor	Truncamiento	floor(9.8)	9
sin	Seno (ángulo en radianes)	sin(pi/6)	0.5
cos	Coseno (ángulo en radianes)	cos(pi/3)	0.5
tan	Tangente (ángulo en radianes)	tan(pi/4)	1.0
cot	Cotangente (ángulo en radianes)	cot(pi/4)	1.0
arcsin	Sin ⁻¹ (devuelve radianes)	arcsin(0.5)	pi/6
arccos	Cos ⁻¹ (devuelve radianes)	arccos(0.5)	pi/3
arctan	Tan ⁻¹ (y,x) (devuelve radianes)	arctan(10,3)	1.2793
arccot	Cot ⁻¹ (y,x) (devuelve radianes)	arccot(10,3)	0.29145
random	Genera un número aleatorio	Random*100	Algún valor entre 0 y 99.999
pi	Valor de pi	pi	3.141592741012 57
Length_of	Devuelve la cantidad de caracteres de una cadena	x←"Hoy es" Length_of(x)	6

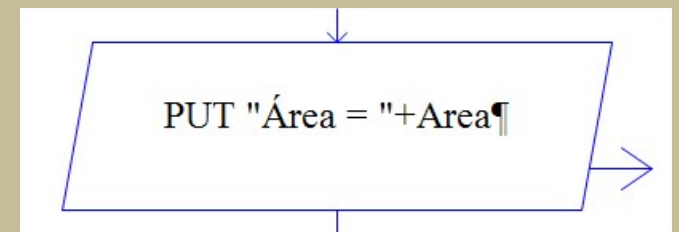
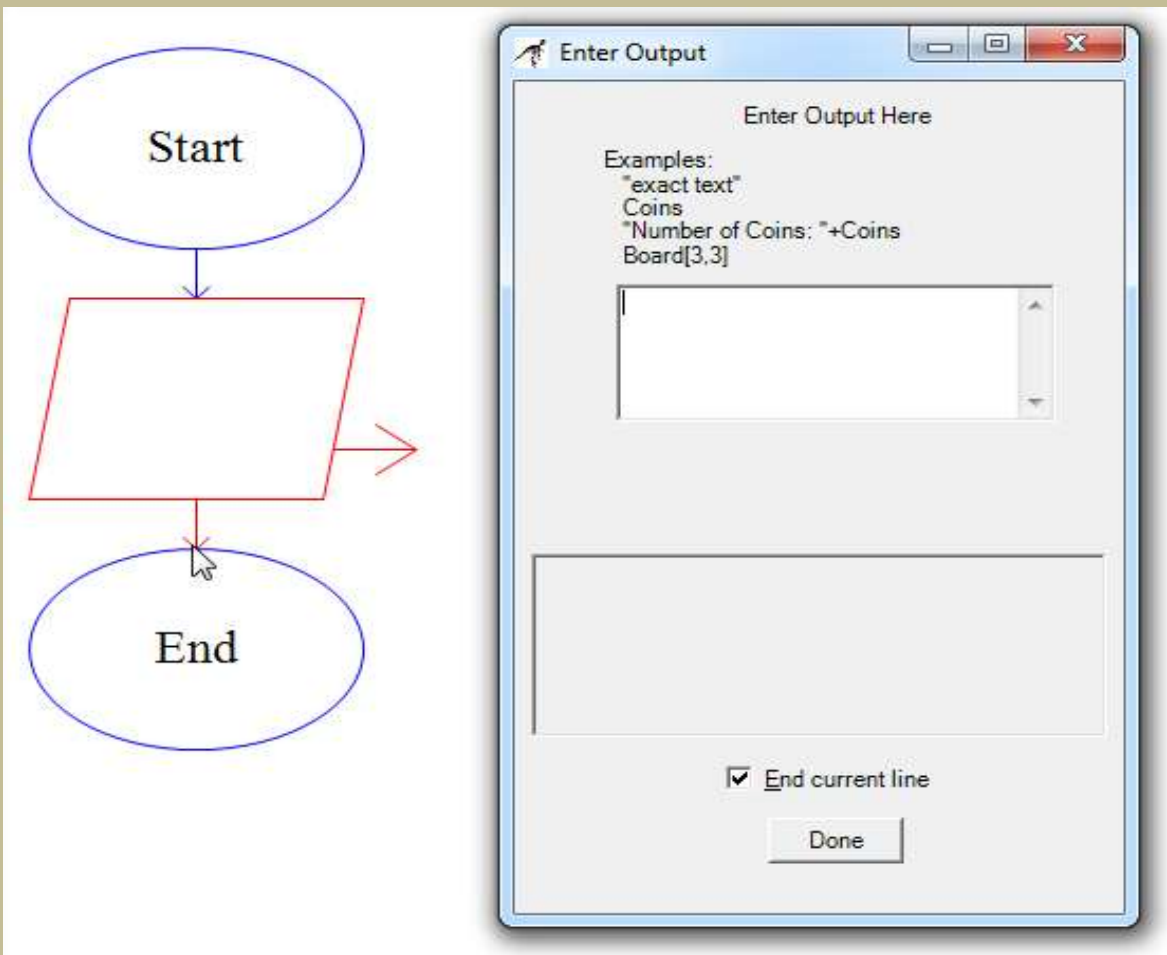
Llamadas a métodos

Para invocar un método basta con colocar un símbolo identificado con la etiqueta “Call” y escribir dentro el nombre del método solicitado



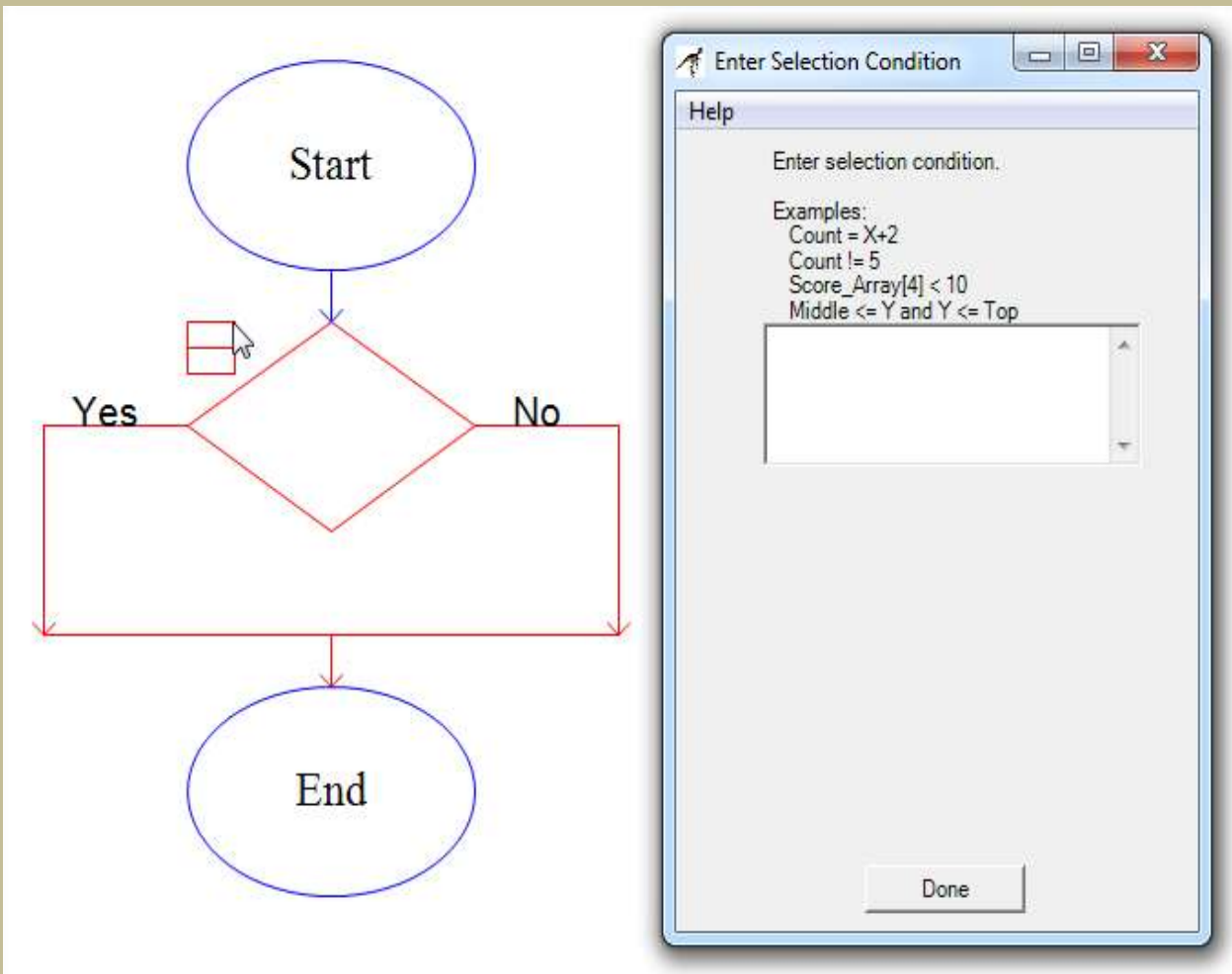
Salida

En Raptor un símbolo de salida despliega un valor y/o un mensaje en la pantalla y se identifica con una flecha que apunta hacia afuera de él



Estructura selectiva

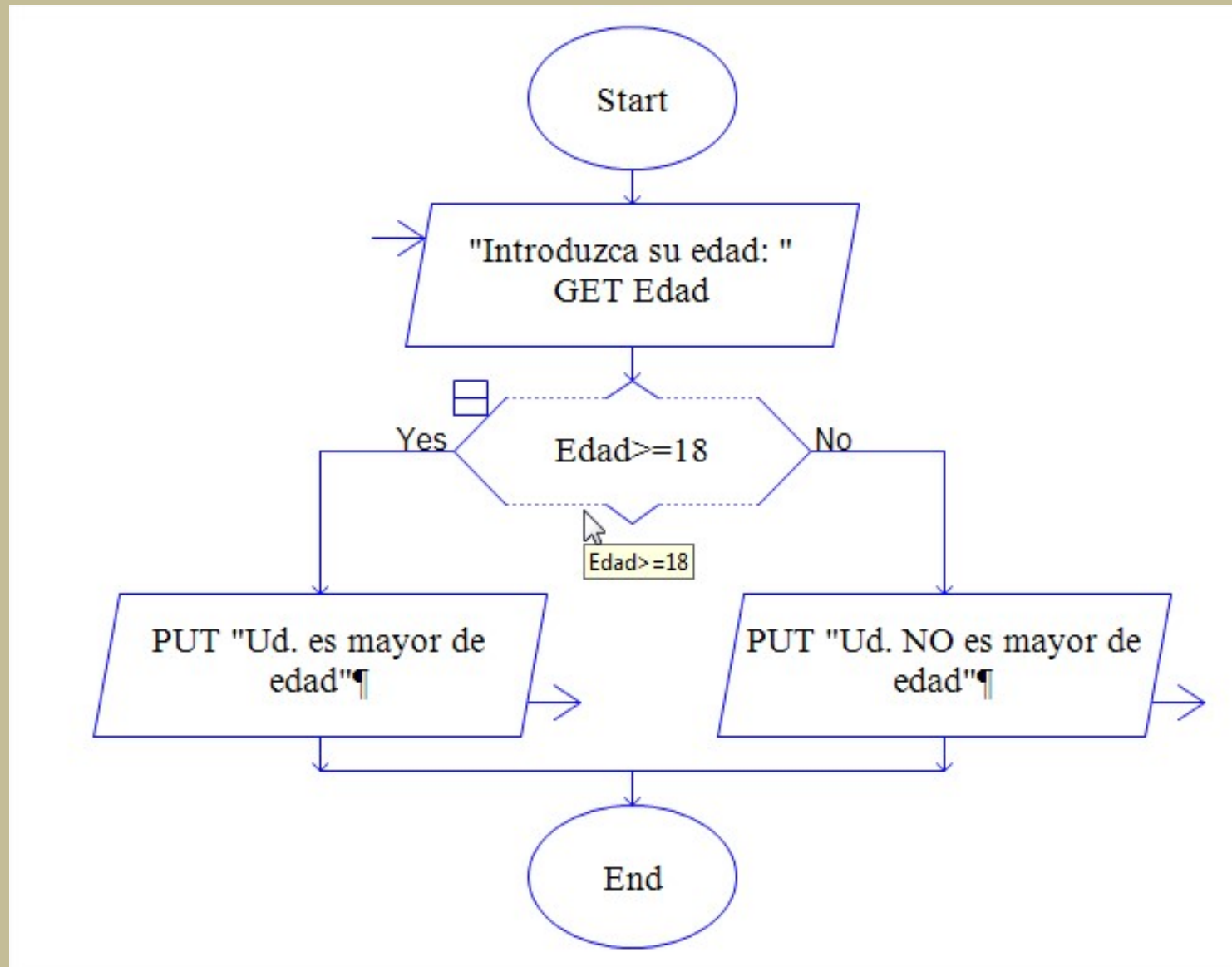
Se utiliza este tipo de estructura de control de flujo para que el algoritmo tome una decisión y determine cuáles sentencias deben ejecutarse y cuáles no.



Operadores relacionales y operadores lógicos

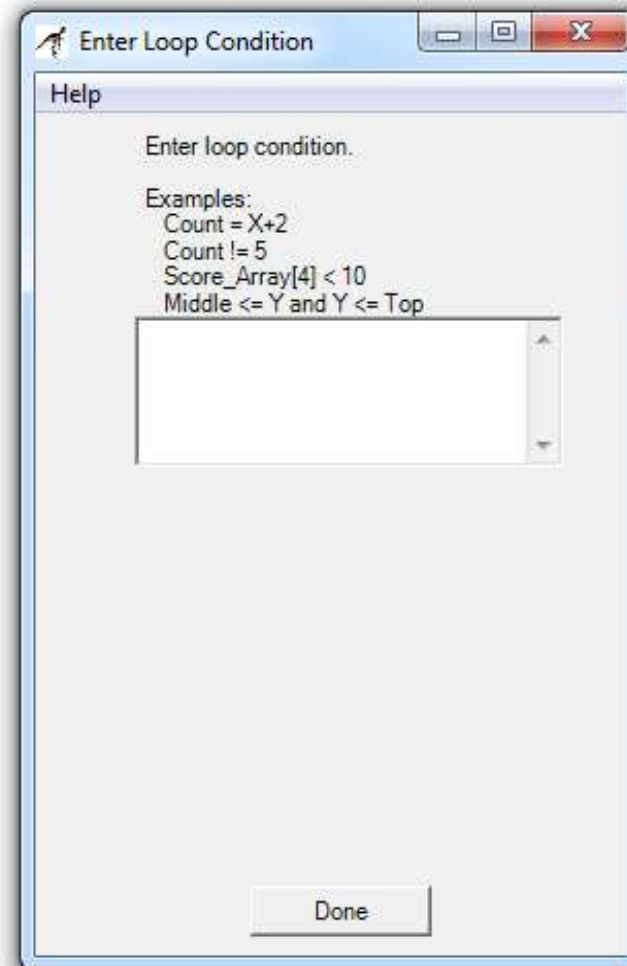
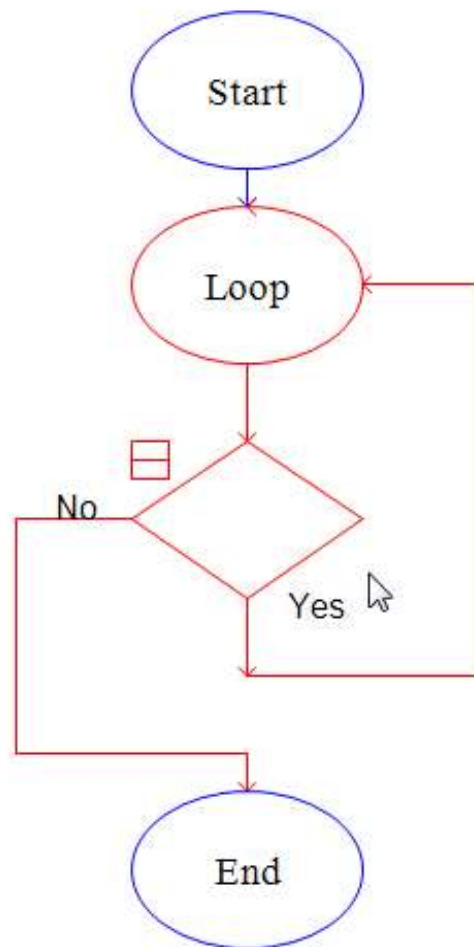
Operador	Descripción	Ejemplo	Resultado
=	“es igual a”	$5 = 4$	No
!= o /=	“no es igual a “	$7 != 8$ $7 /= 8$	Yes
<	“es menor que”	$3 < 2$	No
<=	“es menor o igual que”	$5 <= 7$	Yes
>	“es mayor que”	$9 > 8$	Yes
>=	“es mayor o igual que”	$2 >= 3$	No
and	“es verdadero si todos son verdaderos”	$(5 > 3) \text{ and } (3 > 2)$	Yes
or	“es verdadero si alguno es verdadero”	$(3 > 8) \text{ or } (6 > 4)$	Yes
not	“no es cierto que”	$\text{not } (3 > 5)$	Yes

Ejemplo

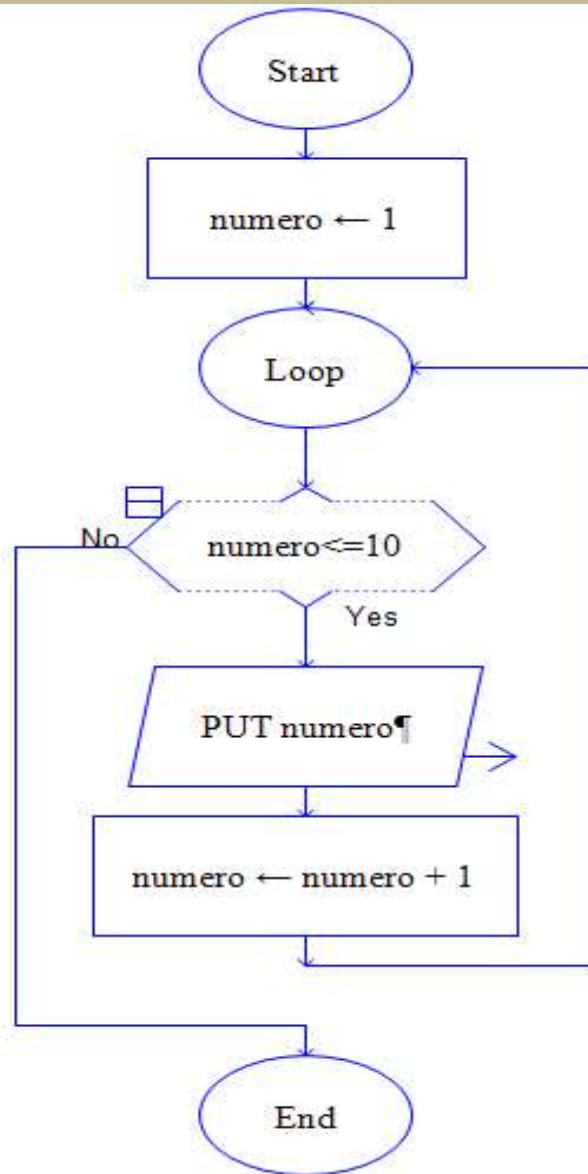


Estructuras iterativas (ciclos)

Una estructura iterativa, repetitiva o cíclica (también conocida como ciclo) se utiliza para que un algoritmo ejecute en varias ocasiones una instrucción, un conjunto de sentencias o un mismo proceso.

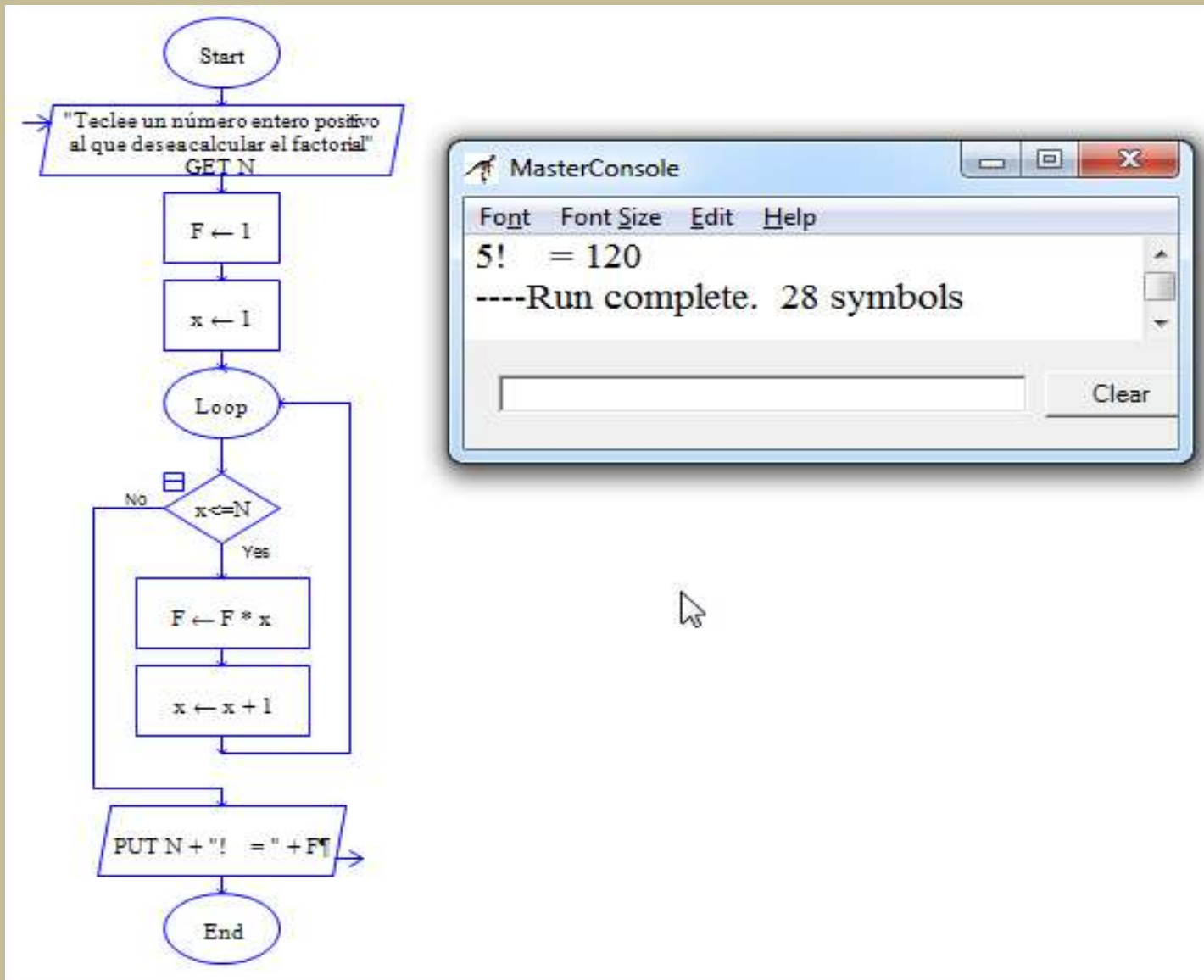


Ejemplo



The screenshot shows a window titled 'MasterConsole' with a menu bar containing 'Font', 'Font Size', 'Edit', and 'Help'. The main area displays the output of the program: a list of numbers from 1 to 10, followed by the message '----Run complete. 45 symbols evaluated.----'. At the bottom of the window, there is an input field and a 'Clear' button.

Algoritmo que calcula el factorial de un número con un ciclo



Prácticas

■ Descargue del sitio web:

<https://nlaredo.tecnm.mx/takeyas/LibroISC>

■ *Práctica 10.1.- Ejercicios con Raptor*



Tarea

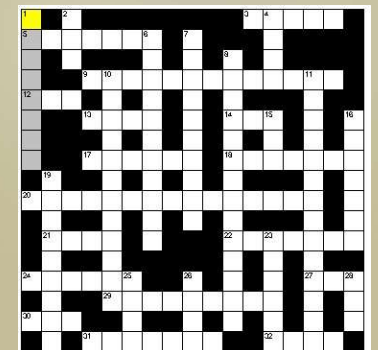
Resuelva en el sitio web

<https://nlaredo.tecnm.mx/takeyas/LibroISC>

Cuestionario 10.1



Crucigrama 10.1

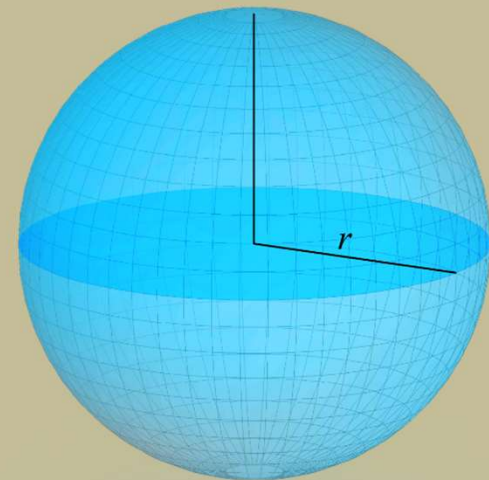


Ejercicio resuelto

Consulte en el sitio web

<https://nlaredo.tecnm.mx/takeyas/LibroISC>

- ❑ Ejercicio resuelto 10.11.- Análisis y diseño orientado a objetos para calcular el volumen de una esfera



Fuentes de información:

- López Takeyas, Bruno. (2019) “Introducción a la Ingeniería en Sistemas Computacionales y al diseño orientado a objetos”. Editorial Pearson.
- <https://nlaredo.tecnm.mx/takeyas/LibroISC/>

