



**Universidad Autónoma del Estado de México**  
UAEM



**CENTRO UNIVERSITARIO  
UAEM ZUMPANGO**

**INGENIERÍA EN COMPUTACIÓN**

Unidad de Aprendizaje:

# PROGRAMACIÓN 1



Unidad de Competencia I:

**ELEMENTOS DE UN PROGRAMA Y SU  
REPRESENTACIÓN EN PSEUDOCÓDIGO Y UML.**

**M. en C. Edith Cristina Herrera Luna**



# CLASES Y OBJETOS

PROGRAMACIÓN 1

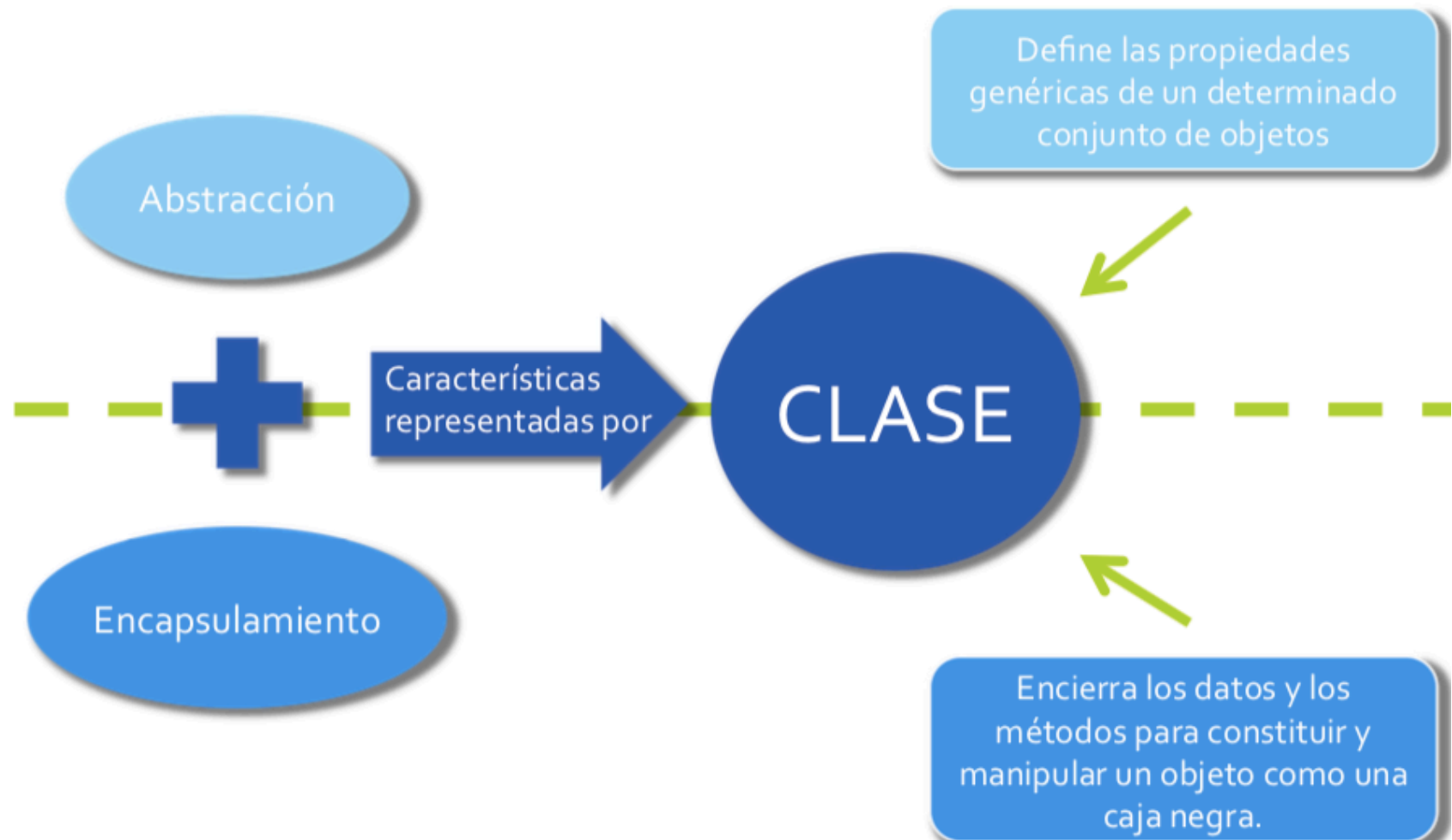
# ABSTRACCIÓN Y ENCAPSULAMIENTO

- + **Abstracción:** Obtener una representación de las características esenciales de algo sin incluir antecedentes o detalles irrelevantes.
- + **Encapsulamiento:** U ocultamiento de información se refiere a la práctica de incluir dentro de un objeto todo lo que necesita, de tal forma que no sea imprescindible conocer su estructura interna.



¿Qué aspectos o características son las mas importantes en un automóvil?... ¿Si es para uso diario, competencia, exhibición?

# ABSTRACCIÓN Y ENCAPSULAMIENTO



# CLASE, OBJETO, ATRIBUTO Y MÉTODO

- + **OBJETO:** Entidad o unidad que representa un concepto específico y contiene la información necesaria para abstraerlo: datos que describen sus propiedades (atributos) y operaciones que definen su comportamiento (métodos).
  - Representa el caso específico de una clase
  - Un objeto es la instancia de una clase.
  
- + **CLASE:** Es la definición o descripción de un objeto por medio de su estado y comportamiento (atributos y métodos). Es la generalización de un tipo de objetos

# CLASE, OBJETO, ATRIBUTO Y MÉTODO

- + Un **objeto** es la *instancia de una clase* y representa un caso individual o específico de dicha clase

Una clase no es un objeto, es la **definición** de un objeto. Describe sus características y su comportamiento.

CLASE



Instancia es otra manera de decir objeto

OBJETO



# CLASE, OBJETO, ATRIBUTO Y MÉTODO

## + Atributo:

O variable de instancia, es un elemento que describe al objeto, son las cosas que los objetos “tienen” o “conocen” sobre sí mismos, representan el **estado** – ó datos – de un objeto.



## + Método:

Son las acciones que puede realizar o se pueden realizar sobre un objeto, representan el **comportamiento** que opera sobre los datos o atributos de la clase.

# CLASE, OBJETO, ATRIBUTO Y MÉTODO

+ De la siguiente lista, indica si se hace referencia a un objeto, método, atributo o clase.



casa		perro
calcularPromedio		Linux
tipoDeFuente		edad
planeta		profesion
Luna		superheroe
correr		programar
alumno		java
calificacion		id

# Ocultamiento de información

Los modificadores de acceso son palabras reservadas que permiten indicar quien o que puede tener acceso a los métodos, clases y variables de instancia.

Generalmente existen los modificadores `public`, `private`, `protected` y por default. Para nuestro estudio, analizaremos:

- + **private:** Los métodos y atributos privados solo pueden ser accedidos por métodos o variables dentro de la misma clase.
- + **public:** Los métodos y variables de instancia publicas pueden ser accedidos desde cualquier clase.



¿Qué modificadores de acceso son validos en el lenguaje de programación estudiado?

# INSTANCIACIÓN DE OBJETOS

Para crear o instanciar un objeto se necesita definir una clase.

## + CLASE:

```
ModificadorDeAcceso NombreClase {  
    //Cuerpo/Ámbito de la clase o Nivel de clase  
    //ATRIBUTOS Y MÉTODOS  
}
```

## + ATRIBUTOS:

Declaración:

```
ModificadorDeAcceso TipoDeDato Identificador [= valor ] ;
```

## + MÉTODOS:

Declaración:

```
ModificadorDeAcceso TipoDeDatoRetornado IdMétodo (Parámetros ) {  
    //Cuerpo del método  
}
```

# INSTANCIACIÓN DE OBJETOS

## + OBJETO:

Declaración:

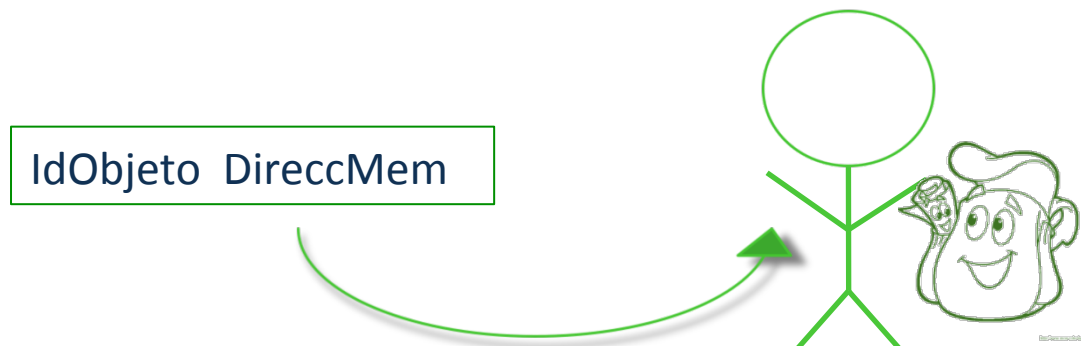
```
ModificadorDeAcceso TipoDeDato IdObjeto;
```

Creación / Reserva de memoria:

```
IdObjeto = new Constructor( );
```

Declaración y creación:

```
ModificadorDeAcceso TipoDeDato IdObjeto = new Constructor( );
```





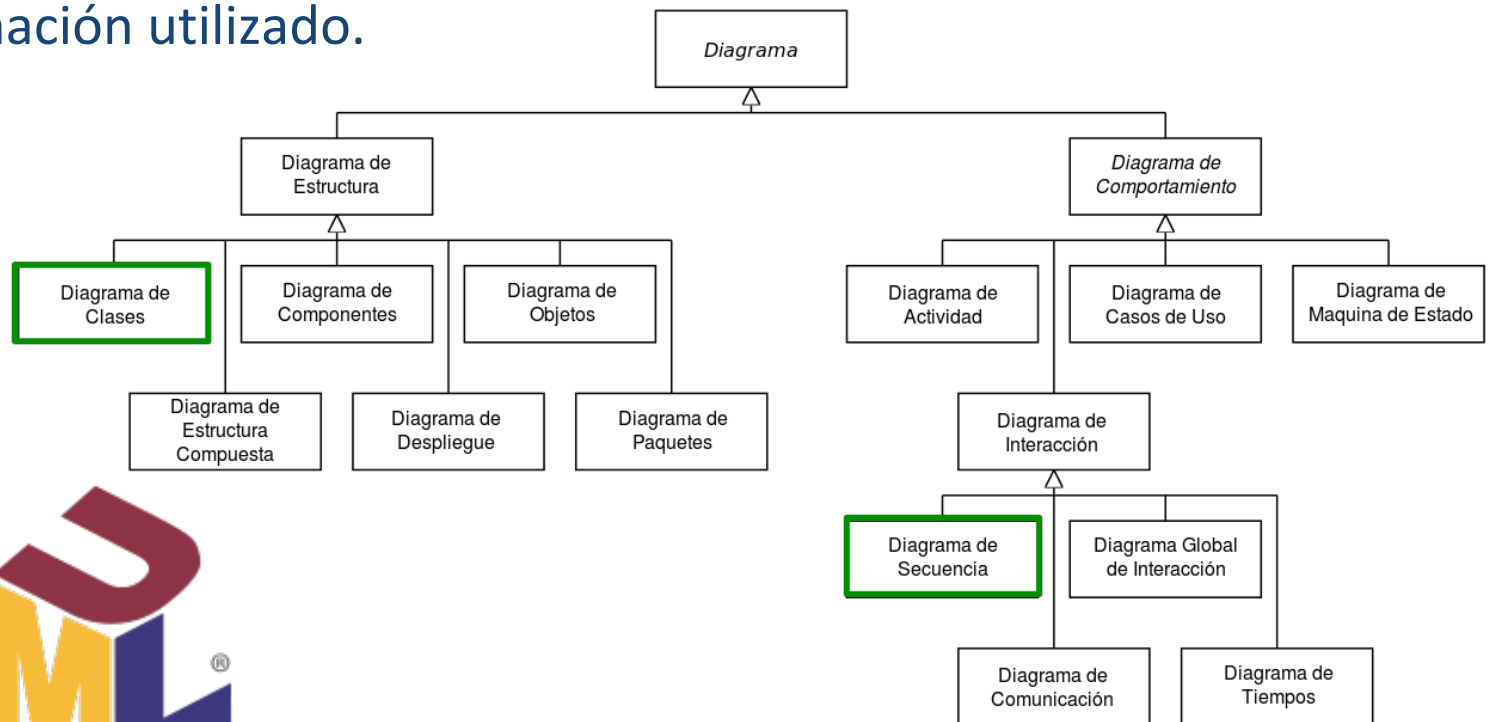
# DIAGRAMAS UML

PROGRAMACIÓN 1

# DIAGRAMAS UML

## + UML: LENGUAJE DE MODELADO UNIFICADO

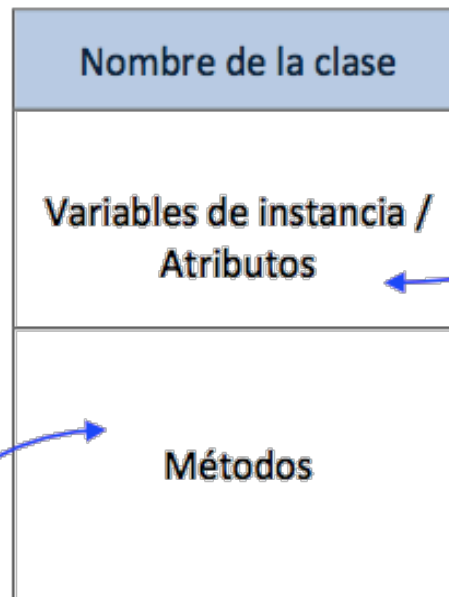
Es un lenguaje de modelado gráfico para visualizar, especificar, construir y documentar un sistema. Es independiente al lenguaje de programación utilizado.



# DIAGRAMAS UML: DIAGRAMA DE CLASE

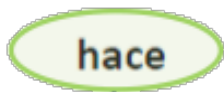
Usa **DIAGRAMAS DE CLASE** de UML (Lenguaje de Modelado Unificado) para definir una clase.

serán creados a partir de esa clase:



• Cosas que el objeto **conoce ó tiene**  
↓  
**Variables de instancia**  
(estado)

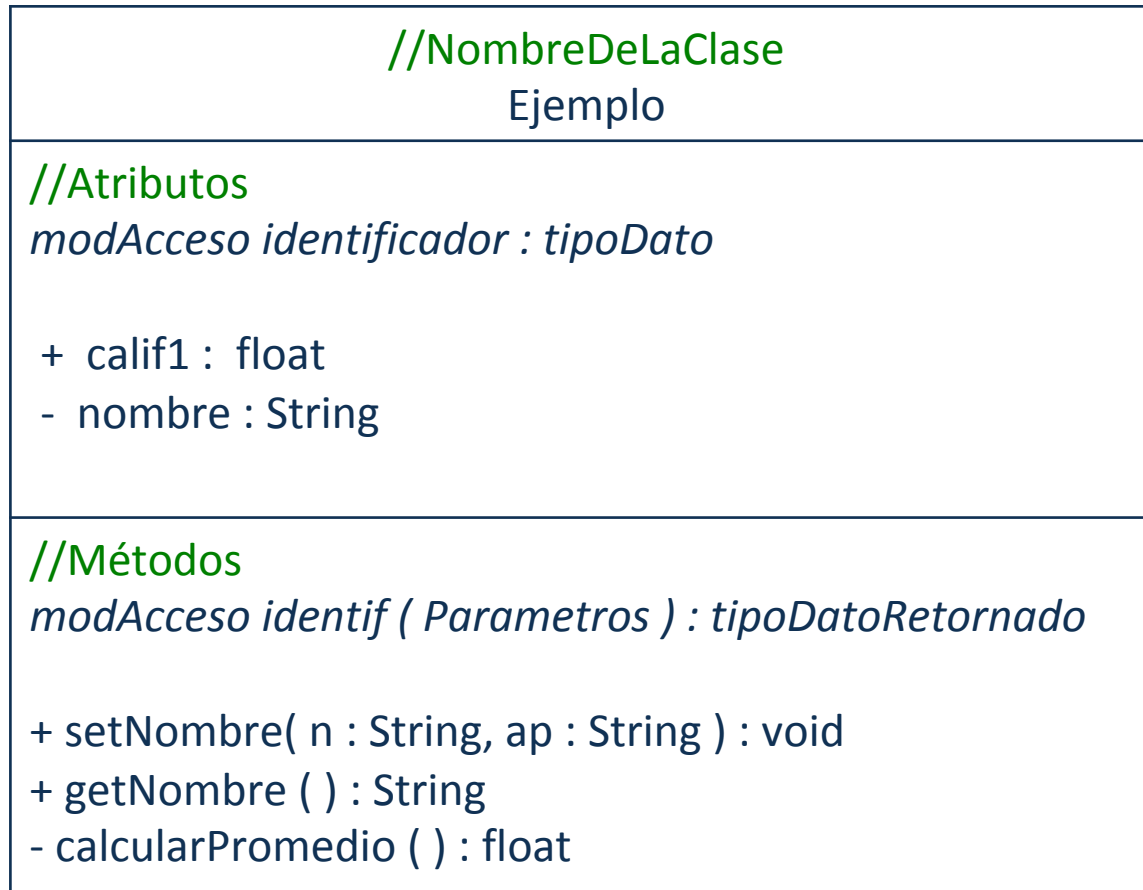
• Cosas que el objeto



↓  
**Métodos**

(comportamiento)

# DIAGRAMAS UML: DIAGRAMA DE CLASE



# DIAGRAMAS UML: DIAGRAMA DE CLASE

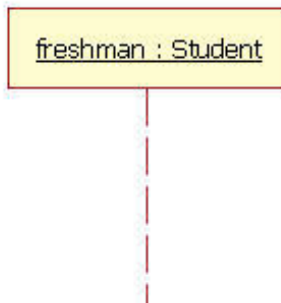
- + Crea los siguientes diagramas de clase, cada uno debe contener entre 3 y 5 atributos y métodos.



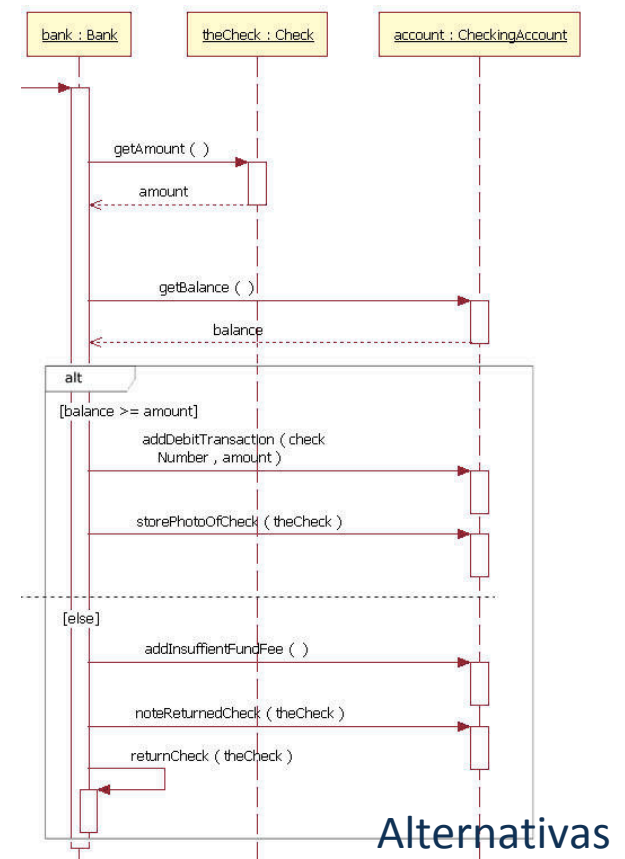
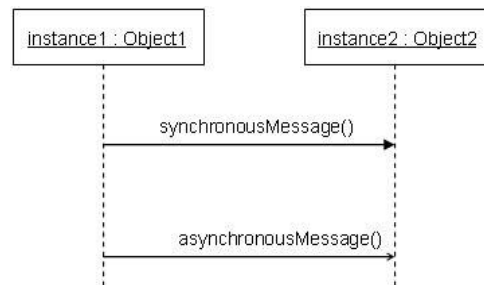
# DIAGRAMAS UML: DIAGRAMA DE SECUENCIA

- + Son diagramas que muestran la forma en que un grupo de objetos interactúan entre sí, consta de objetos, mensajes entre objetos y una línea de vida del objeto.
- + Se centran específicamente en líneas de vida o en los procesos y objetos que coexisten simultáneamente.

Línea de vida

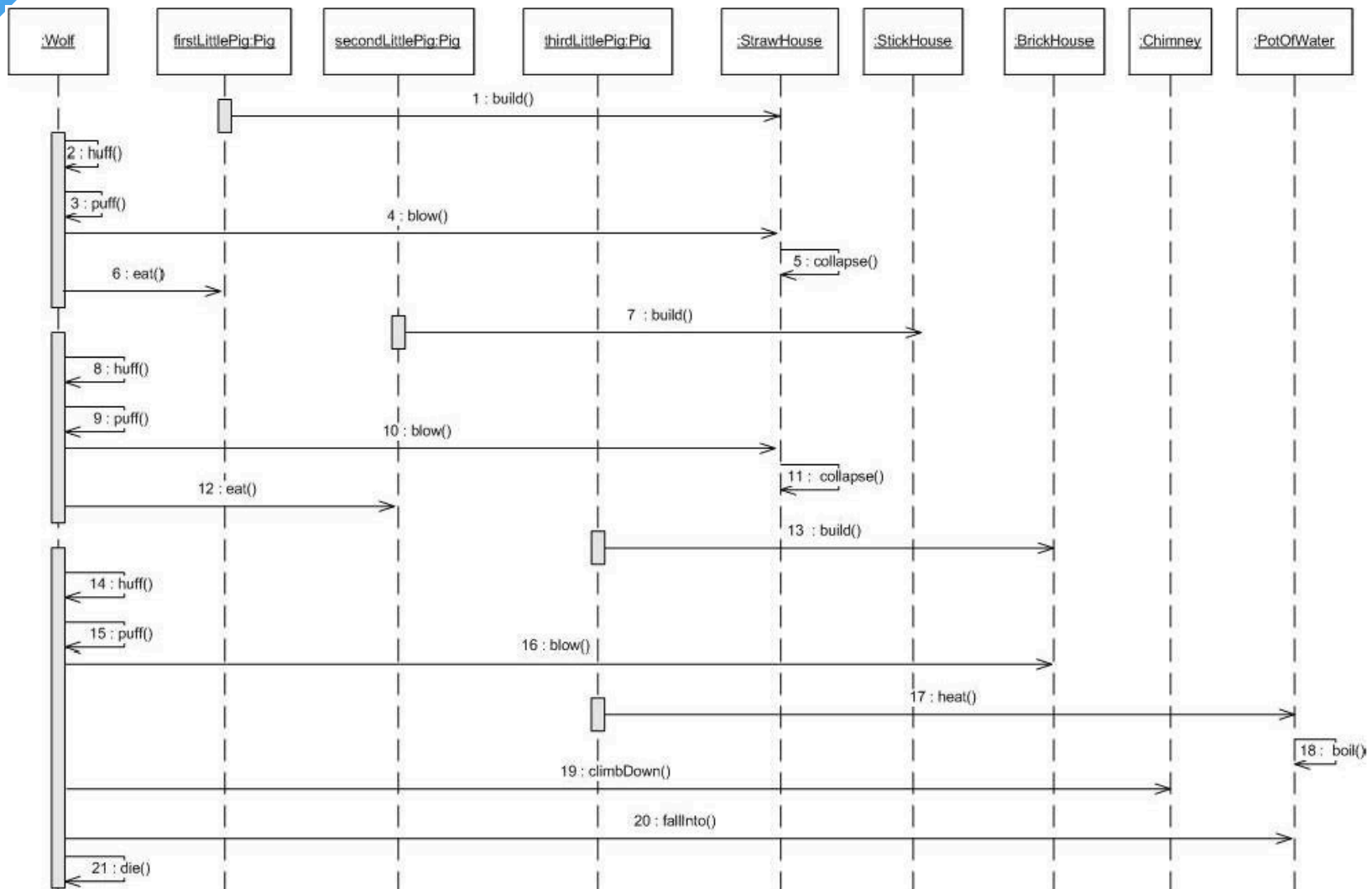


Mensajes



Alternativas

# + ¿Qué secuencia muestra el siguiente diagrama?





# ESTRUCTURAS DE CONTROL

PROGRAMACIÓN 1

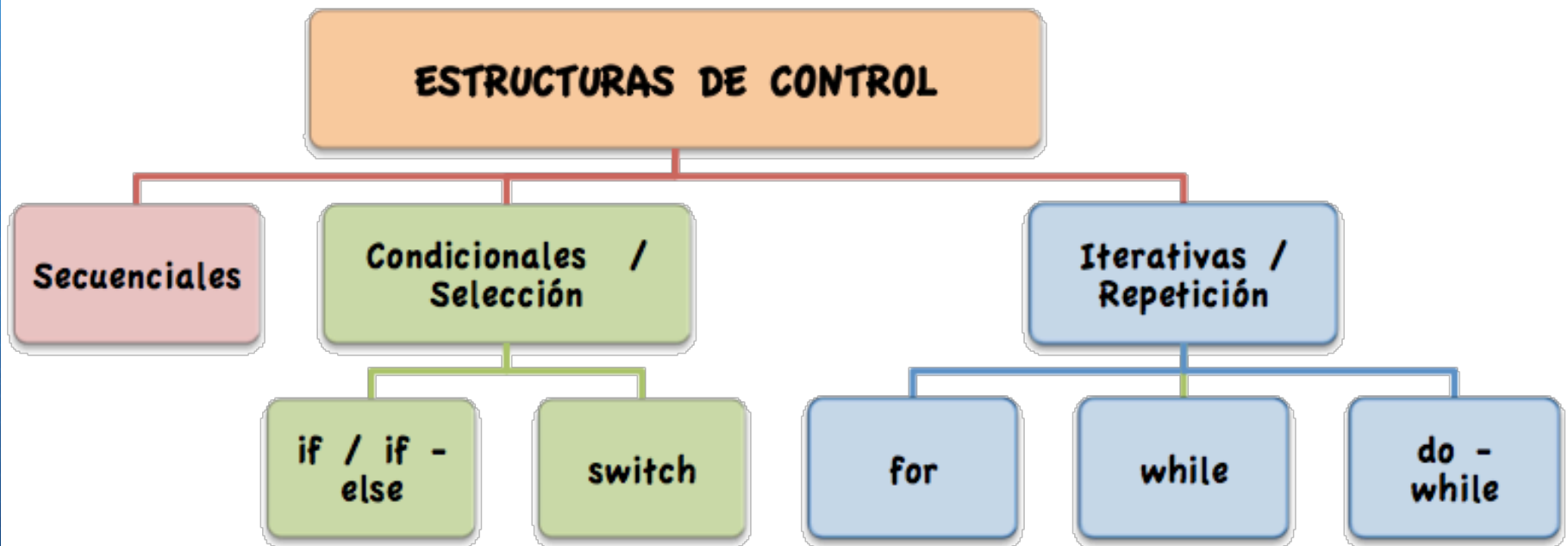
# ESTRUCTURAS DE CONTROL

En un programa, el flujo de información no siempre debe ejecutarse conforme aparece en el código: una secuencia detrás de otra. Para resolver problemas generales es necesario que el programador pueda controlar que sentencias se debe de ejecutar en función de los datos y expresiones evaluadas.

- + Las estructuras de control son instrucciones que permiten que el flujo de ejecución de un programa sea controlado por el programador.

Existen tres tipos de estructuras de control: *Secuenciales*, *Selectivas* o *Condicionales* y *Repetitivas* o *Iterativas*.

# ESTRUCTURAS DE CONTROL



Investigar el artículo científico en el que se definen por primera vez las estructuras de control



# REFERENCIAS

PROGRAMACIÓN 1

# REFERENCIAS


- + Osvaldo Cairo. *Metodología de la programación*. 3ª. Edición. Alfaomega. ISBN: 970-15-1100-X. (2005)
- + José R. García-Bermejo. *Programación estructurada en C*. Giner. Pearson Educación, España 2008. ISBN: 978-84-8322-423-6.
- + Stephen G. Kochan. *Programming in C*. Tercera Edición, Sams Publishing. ISBN: 0-672-32666-3
- + Albarrán, T. S. E. & Salgado, G. M. (2008). *Programación Estructurada*. Universidad Autónoma del Estado de México.
- + Cairó, Osvaldo y Guardati, Silvia. (2006). *Estructuras de datos* (3a. Edición). McGraw-Hill.
- + Walter Savitch. *Absolute C++*.
- + Joyanes, A. L., & Zahonero, M. I. (2003). *Programación en C, Metodología, Estructura de Datos y Objetos*. México. McGraw-Hill.
- + Koffman, Elliot y Wolfgang, Paul. (2008). *Estructura de datos con C++*. Objetos, abstracciones y diseño. McGraw-Hill.
- + García, B., & Giner, J. R. (2008). *Programación Estructurada en C*. Madrid. Pearson Educación.
- + Norton, P. (2006). *Introducción a la Computación* (6a ed.). México. McGraw-Hill.
- + Ken Howard. Fuente: <http://www.tracemodeler.com/articles/pimp-my-diagram-three-little-pigs/img/original-sequence-diagram-by-ken-howard.png>
- + Sierra, Katy; Bates, Bert. *Head First Java*. O'Reily Media. (2005)
- + The Sequence Diagram. Donald Bell. IBM. <https://developer.ibm.com/articles/the-sequence-diagram/>



# GRACIAS

**PROGRAMACIÓN 1**  
**Continúa Unidad 2: Proceso de Programación**

# GUÍA PARA EL PROFESOR

- + Las primeras diapositivas muestran el propósito, justificación y objetivos de la unidad de aprendizaje. Se presentan para que el alumno identifique dichos elementos.
- + El contenido, conforme a la unidad de aprendizaje, maneja los temas de un menor a mayor grado de dificultad.
- + Se hacen notas respecto a diversas actividades sugeridas conforme se avanza con el tema, se pueden identificar claramente con el icono: 
- + Es conveniente que antes de dar algún tema, se pida al alumno investigue sobre algunos conceptos básicos. De igual forma, se debe combinar el uso de diapositivas con ejemplos y ejercicios de codificación para poner en práctica la sintaxis marcada.
- + Considere que el tema de UML es muy extenso, sin embargo, para esta primera asignatura de programación se abordan los temas de manera sencilla y precisa.

# GUÍA PARA EL PROFESOR

- + Existe poca sintaxis de un lenguaje en específico, debido a que al ser una unidad de introducción a la programación, los conceptos básicos pueden ser implementados para diversos lenguajes, como C, C++ y Java.
- + Debido a la facilidad y naturaleza del lenguaje Java para la programación orientada a objetos, en la sintaxis referente a los elementos de la programación (clase, objeto, método y atributo) se hace uso del mismo.
- + En cuanto al tema de estructuras de control, solo se hace mención de que es una estructura y se pide investigación del tema porque en la Unidad 3, se trata a fondo el tema definiendo claramente cada una de las estructuras.