



Universidad Nacional Autónoma de México
Facultad de Ingeniería
Programación orientada a objetos
Tema 1:
EL PARADIGMA ORIENTADO A OBJETOS

1 El paradigma orientado a objetos.

1 El paradigma orientado a objetos

Objetivo: Interpretar los conceptos de la programación orientada a objetos para aplicarlo a eventos del mundo real.

1 El paradigma orientado a objetos

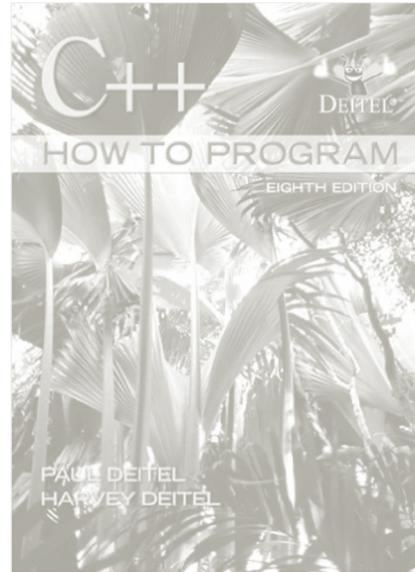
1.1 Elementos básicos del paradigma orientado a objetos.

- 1.1.1 Tipos de datos primitivos y abstractos.**
- 1.1.2 Objetos.**

1.2 Propiedades básicas del paradigma orientado a objetos.

- 1.2.1 Abstracción.**
- 1.2.2 Cohesión.**
- 1.2.3 Encapsulamiento.**
- 1.2.4 Modularidad.**
- 1.2.5 Herencia.**
- 1.2.6 Polimorfismo.**
- 1.2.7 Acoplamiento.**
- 1.2.8 Jerarquía de clases.**

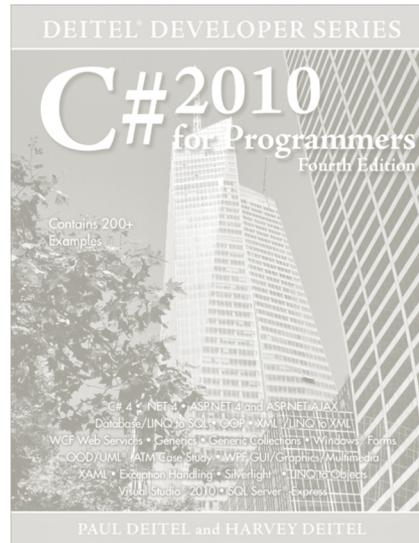
Bibliografía



C++ How to Program. DEITEL, Paul, DEITEL, Harvey, 8th edition, New Jersey, Prentice Hall.

1 El paradigma orientado a objetos.

Bibliografía



**C# 2010 for Programmers. DEITEL, Paul, DEITEL, Harvey,
4th edition, New Jersey, Prentice Hall.**

1 El paradigma orientado a objetos.

1 EL PARADIGMA ORIENTADO A OBJETOS



El concepto de paradigma procede del griego παραδειγμα (paradeigma), que significa modelo o ejemplo.

Un paradigma es un modelo o patrón sostenido en una disciplina. Comprende un grupo de principios fundamentales que producirán una realidad con un propósito particular.

¹ El paradigma orientado a objetos.

En la ciencia, un paradigma es un conjunto de realizaciones científicas universalmente reconocidas que, durante cierto tiempo, proporcionan modelos experimentales que dan solución a otros modelos.

Por ejemplo, el paradigma de Ptolomeo implicaba que la tierra era plana y que el Sol giraba alrededor de la Tierra. Las observaciones de Copérnico probaron lo contrario y generaron el paradigma de que la Tierra es redonda y gira alrededor del Sol.

¹ El paradigma orientado a objetos.

Un paradigma de programación se refiere a la forma, visión o manera que determinan los métodos y las herramientas que un programador usará en la construcción de un software.

Se refiere a una filosofía (o forma) de atacar y analizar problemas, así como diseñar e implementar una solución en una computadora.

¹ El paradigma orientado a objetos.

Un paradigma de programación permite atacar un problema con un enfoque particular para la construcción del software.

Dentro de la programación existen diferentes tipos de paradigmas de programación:

- **Paradigma imperativo**
- **Paradigma lógico**
- **Paradigma funcional**
- **Paradigma lógico**
- **Paradigma declarativo**
- **Paradigma estructurado**
- **Paradigma orientado a objetos**

¹ El paradigma orientado a objetos.

A pesar de que existen diferentes tipos de paradigmas de programación, no se puede decir que uno es mejor que otro sino que cada uno tiene ventajas y desventajas y se van a utilizar dependiendo del contexto del problema.

Los lenguajes de programación están basados en uno o más paradigmas de programación.

¹ El paradigma orientado a objetos.



1.1 ELEMENTOS BÁSICOS DEL PARADIGMA ORIENTADO A OBJETOS.

122

El paradigma orientado a objetos se basa en el hecho de dividir la lógica en modelos de objetos físicos o simulados. Por lo tanto, la solución se expresa como un conjunto de objetos que colaboran entre sí con un fin único (para realizar una tarea específica).

1 El paradigma orientado a objetos.

1.1.1 Tipos de datos primitivos y abstractos.

Los tipos de datos hacen referencia al tipo de información con la que se trabaja, donde la unidad mínima de almacenamiento es el dato. Existen dos tipos de datos: los primitivos y los abstractos.

Un tipo de dato primitivo permite almacenar la información básica de cualquier objeto. Todos la información que se almacena en un objeto cae, necesariamente, en un tipo de dato primitivo.

Los tipos de datos primitivos básicos son:

- **Booleanos**
- **Caracteres**
- **Enteros**
- **Reales**

¹ El paradigma orientado a objetos.

Un tipo de dato abstracto permite definir un conjunto de datos primitivos, así como un conjunto de operaciones agrupadas bajo un mismo ente. Estos tipos de datos se les conoce como objetos.

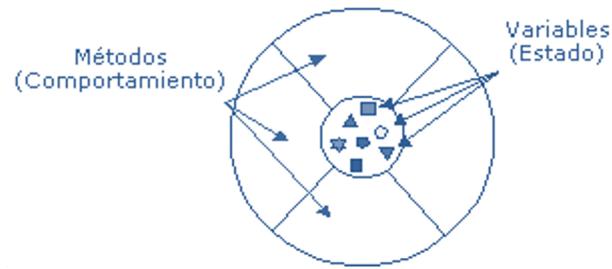
1 El paradigma orientado a objetos.

1.1.2 Objetos.

Un objeto es la representación en un ente o concepto. Contiene toda la información necesaria para describirlo de manera abstracta: datos primitivos que describen sus atributos y operaciones que pueden realizarse sobre los mismos.

Los objetos son la base del paradigma orientado a objetos.

Un objeto encapsula de manera genérica los datos y de los métodos que es capaz de tener y manipular.



El objeto es el centro de la programación orientada a objetos. Es un ente que se visualiza, se utiliza y que juega un papel o un rol preponderante en el dominio del problema del programa.

La estructura interna y el comportamiento de un objeto, en consecuencia, no es prioritario durante el modelado del problema.

¹ El paradigma orientado a objetos.

1.2 PROPIEDADES BÁSICAS DEL PARADIGMA ORIENTADO A OBJETOS.

20

1.2 Propiedades básicas del paradigma orientado a objetos.

Los cimientos del paradigma orientado a objetos son el encapsulamiento, la herencia, el polimorfismo y la abstracción.

Adicionalmente, un objeto posee propiedades de cohesión y acoplamiento, bajo un funcionamiento modular.

Así mismo, un conjunto de clases que se relacionan entre sí a través de la herencia se conoce como jerarquía de clases.

1.2.1 Abstracción.

La abstracción es la propiedad que considera los aspectos más significativos o notables de un problema y expresar una solución en esos términos.

La abstracción posee diversos grados, lo que se conoce como niveles de abstracción, los cuales ayudan a estructurar la complejidad intrínseca que poseen los sistemas del mundo real.

¹ El paradigma orientado a objetos.



EJEMPLO

23

1 El paradigma orientado a objetos.

1.2.2 Cohesión.

El término cohesión en ingeniería de software se refiere al grado en el cual los elementos de un módulo pertenecen entre sí, es decir, qué es lo que el módulo puede hacer.

Una baja cohesión significa que módulo tiene una gran variedad de acciones y no se enfoca en lo que debería hacer. Mientras que una alta cohesión significa que el módulo se especializa en lo que debería hacer.

¹ El paradigma orientado a objetos.

1.2.3 Encapsulamiento.

La encapsulación o encapsulamiento significa reunir en una estructura a todos los elementos que en un nivel de abstracción se pueden considerar pertenecientes a una misma entidad. Consiste en agrupar de datos y operaciones relacionadas bajo una misma unidad, lo que permite aumentar la cohesión de los componentes del sistema.

El encapsulamiento oculta lo que hace un objeto de lo que hacen otros objetos y del mundo exterior por lo que se denomina también ocultación de datos.

Un objeto tiene que presentar “una cara” al mundo exterior de modo que se puedan invocar sus operaciones.

¹ El paradigma orientado a objetos.

1.2.4 Modularidad.

Se refiere proceso de descomposición de un problema para reducir la complejidad del mismo. Una vez que se ha descompuesto el problema en múltiples módulos, éstos se pueden ligar y combinar para formar la solución integral del problema.

La modularidad hace posible que un recurso (módulo) sea reusable y minimiza el grado de duplicidad.

¹ El paradigma orientado a objetos.

1.2.5 Herencia.

La herencia es una propiedad que permite crear nuevos objetos que asumen las propiedades de objetos existentes. Utilizando herencia la relación jerárquica queda establecida de manera implícita, partiendo de la clase más general (clase base) a la clase más específica (clase derivada).

¹ El paradigma orientado a objetos.

Las dos razones más comunes para utilizar herencia son:

- **Para promover la reutilización de código.**
- **Para usar polimorfismo.**

1 El paradigma orientado a objetos.

1.2.6 Polimorfismo.

El término polimorfismo es una palabra de origen griego que significa muchas formas. Es la propiedad por la que se pueden enviar mensajes sintácticamente iguales a objetos de tipos distintos.

Permite que un objeto de una clase se comporte como un objeto de cualquiera de sus subclases.

¹ El paradigma orientado a objetos.

El polimorfismo se puede clasificar en dos grandes grupos:

- **Polimorfismo dinámico (o paramétrico):** es aquel en el que no se especifica el tipo de datos sobre el que se trabaja y, por ende, se puede recibir utilizar todo tipo de datos compatible (programación genérica).
- **Polimorfismo estático (o ad hoc):** es aquel en el que los tipos de datos que se pueden utilizar deben ser especificados de manera explícita antes de ser utilizados.

¹ El paradigma orientado a objetos.

1.2.7 Acoplamiento.

En la ingeniería de software, se refiere a qué tan fuerte está relacionado un elemento de software con otro.

Un elemento de software puede ser una clase, un paquete, un componente o un sistema. Al diseñar un sistema se recomienda tener elementos con alta cohesión y bajo acoplamiento.

1.2.8 Jerarquía de clases.

Cuando se utiliza la herencia se establece de manera implícita un grafo relacional, partiendo del elemento más general (base) al elemento más específico (derivado).

Dicha distribución se conoce como jerarquía de clases y permite identificar los objetos abstractos, los objetos internos y los objetos finales.

“Programming languages, like pizzas, come in only two sizes: too big and too small.”

**Richards E. Pattis
(He is a senior lecturer at the University
of California, Irvine's Donald Bren School
of Information and Computer Sciences.)**

¹ El paradigma orientado a objetos.

1 El paradigma orientado a objetos

Objetivo: Interpretar los conceptos de la programación orientada a objetos para aplicarlo a eventos del mundo real.

1.1 Elementos básicos del paradigma orientado a objetos.

- 1.1.1 Tipos de datos primitivos y abstractos.
- 1.1.2 Objetos.

1.2 Propiedades básicas del paradigma orientado a objetos.

- 1.2.1 Abstracción.
- 1.2.2 Cohesión.
- 1.2.3 Encapsulamiento.
- 1.2.4 Modularidad.
- 1.2.5 Herencia.
- 1.2.6 Polimorfismo.
- 1.2.7 Acoplamiento.
- 1.2.8 Jerarquía de clases.