Paraninfo

UF2404 - Principios de la programación orientada a objetos

Editorial: Paraninfo
Autor: JOSE MANUEL PIÑEIRO GOMEZ
Clasificación: Certificados Profesionales > Informática y Comunicaciones
Tamaño: 17 x 24 cm.
Páginas: 334
ISBN 13: 9788428345224
ISBN 10: 8428345228
Precio sin IVA: $ 105560.00 COP
Precio con IVA: $ 105560.00 COP
Fecha publicacion: 18/05/2022

Sinopsis

Hoy en día utilizamos multitud de programas informáticos que nos permiten realizar muchas de nuestras actividades cotidianas, como utilizar un cajero electrónico o concertar cita con nuestro médico de familia. Debido a esto, la creación de programas o aplicaciones informáticas es una labor que requiere cada vez más personal especializado.

Este manual es una introducción al mundo de la programación informática y presta especial atención a la programación orientada a objetos, que constituye el paradigma de programación empleado masivamente en la actualidad a nivel mundial.

En los primeros temas se explican las bases de la programación orientada a objetos, según la cual un programa está formado por un conjunto de objetos que interactúan entre sí mediante el envío de mensajes. Se estudia la manera de organizar un programa en un conjunto de clases a partir de las cuales se crean objetos. Se analizan las relaciones que se pueden establecer entre clases, conceptos como el polimorfismo, le herencia, y también aspectos fundamentales en todo lenguaje de programación, como las estructuras de control y distintas estructuras de datos para el almacenamiento de la información.

Se ha optado por elegir el lenguaje orientado a objetos Java, en el que se implementan numerosos ejemplos y casos prácticos. Se incluye también un abanico de ejercicios resueltos y otro de ejercicios propuestos, cuya solución es accesible desde www.paraninfo.es.

La obra responde al contenido curricular previsto en el RD 628/2013 de 2 de agosto para la unidad formativa UF2404 *Principios de la programación orientada a objetos,* integrada en el módulo formativo MF0227_3

Programación orientada a objetos, incluido a su vez en el certificado de profesionalidad IFCD0112 *Programación con lenguajes orientados a objetos y bases de datos relacionales,* perteneciente al área de desarrollo de la familia profesional de Informática y Comunicaciones.

Indice

Introducción

1. Introducción al paradigma orientado a objetos
1.1. Ciclo del desarrollo de software bajo el paradigma orientado a objetos: análisis, diseño y programación orientada a objetos
1.2. Análisis del proceso de construcción de software: modularidad
1.3. Distinción del concepto de módulo en el paradigma orientado a objetos
1.4. Identificación de objetos como abstracciones de las entidades del mundo real que se quieren modelar
1.4.1. Descripción de objetos: conjunto de datos que definen un objeto y conjunto de comportamientos que pueden solicitarse a los objetos
1.4.2. Identificación del comportamiento de un objeto: concepto de mensaje

2. Clases y objetos
2.1. Distinguir el concepto de clase y sus atributos, métodos y mecanismo de encapsulación
2.1.1. Relación entre interfaz y clase
2.1.2. Distinción de los tipos de datos y clases
2.2. Análisis de los objetos: estado, comportamiento e identidad
2.2.1. Análisis de mensajes
2.3. Uso de objetos como instancias de clase. Instancia actual (this, self, current)
2.4. Identificación del concepto de programa en el paradigma orientado a objetos. POO = Objetos + Mensajes

3. Generalización/especialización: herencia
3.1. Descripción del concepto de herencia: simple y múltiple
3.1.1. Relación de herencia: características
3.1.2. Reglas y características que definen una relación de herencia: regla «es-un»
3.1.3. Transmisión de atributos y métodos
3.1.4. Regla de especialización de la superclase en la subclase
3.1.5. Acceso a los atributos de una clase y acoplamiento entre las clases
3.1.6. Utilización de objetos this (self, current u otros) y super
3.1.7. Leyes de Demeter
3.2. Distinción de la herencia múltiple
3.2.1. Problemas: conflictos de nombres, herencia repetida
3.2.2. Soluciones
3.3. Creación de objetos en la herencia
3.4. Clasificación jerárquica de las clases
3.4.1. Clase raíz
3.4.2. Clases abstractas
3.4.3. Métodos virtuales
3.4.4. Redefinición de métodos

4. Relaciones entre clases

4.1. Distinción entre agregación/composición
4.2. Distinción entre generalización/especialización
4.3. Identificación de asociaciones

5. Análisis del polimorfismo
5.1. Concepto
5.2. Tipos
5.2.1. Polimorfismo en tiempo de compilación (sobrecarga)
5.2.2. Polimorfismo en tiempo de ejecución (ligadura dinámica)
5.3. Objetos polimórficos
5.4. Comprobación estática y dinámica de tipos

6. Técnicas de programación estructurada
6.1. Identificación de elementos básicos: constantes, variables, operadores y expresiones
6.2. Análisis de estructuras de control: secuencial, condicional y de repetición
6.2.1. Estructura secuencial
6.2.2. Estructura condicional
6.2.3. Estructura repetitiva
6.3. Distinción entre funciones y procedimientos
6.3.1. Interfaz
6.3.2. Paso de parámetros: por valor y por referencia
6.3.3. Parámetros actuales y formales
6.3.4. Funciones: valor de retorno
6.3.5. Procedimientos
6.3.6. Ámbito de las variables
6.3.7. Almacenamiento de las variables
6.4. Demostración de llamadas a funciones y procedimientos
6.5. Empleo de llamadas a funciones y procedimientos incluidos en las clases
6.5.1. Llamadas calificadas y no calificadas (instancia actual)
6.5.2. Paso de parámetros
6.5.3. Los atributos de la clase
Ejercicios resueltos
Ejercicios propuestos

7. Estructura de la información
7.1. Enumeración de datos simples: numéricos (enteros y reales), lógicos, carácter, cadena de caracteres, puntero o referencia a memoria
7.2. Datos estructurados: arrays
7.2.1. Listas enlazadas, pilas y colas
7.2.2. Estructuras
7.2.3. Ficheros
7.2.4. Otras estructuras complejas: tablas hash e introducción a los árboles y grafos
7.3. Mecanismos de gestión de memoria
7.3.1. Uso de la gestión automática de memoria
7.3.2. Construcción y destrucción de objetos
7.3.3. Objetos inalcanzables
7.3.4. Recolección de «basura»
7.3.5. Métodos constructores y destructores

Ejercicios resueltos
Ejercicios propuestos

8. Lenguajes de programación orientados a objetos
8.1. Análisis del lenguaje de programación orientado a objetos y paradigma orientado a objetos
8.1.1. Lenguajes de programación orientados a objetos
8.1.2. Lenguajes de programación basados en objetos
8.1.3. Lenguajes de programación que utilizan objetos
8.2. Comparación entre los lenguajes de programación orientados a objetos más habituales. Características esenciales
8.3. Librerías de clases
8.3.1. Definición de su estructura
8.3.2. Creación y utilización

9. Implementación del paradigma utilizando un lenguaje de programación orientado a objetos
9.1. Elección del lenguaje
9.2. Enumeración de los tipos de aplicaciones
9.3. Herramientas de desarrollo
9.3.1. Instalación del IDE Eclipse
9.3.2. Creación de programas en el IDE Eclipse
9.3.3. Ejecución de programas en el IDE Eclipse
9.4. Tipos de datos y elementos básicos característicos del lenguaje. Instrucciones
9.4.1. Variables y constantes en Java
9.4.3. El método main
9.4.4. Salida de datos en Java
9.4.5. Entrada de datos en Java
9.4.6. Estructuras de control en Java
9.5. Estudio y utilización de las clases básicas incluidas en la librería de clases
9.5.1. Clases envoltorio (wrapper)
9.5.2. Clase Math
9.5.3. Clases para fechas y horas
9.5.4. Clase Object
9.5.5. Clase Class
9.5.6. Clase Field
9.5.7. Clase Method
9.5.8. Clase Random
9.6. Definición de clases
9.6.1. Construcción de métodos. Sobrecarga
9.6.2. Construcción de atributos
9.6.3. Construcción de la interfaz de la clase
9.6.4. Construcción de clases incluyendo relaciones de agregación/composición y asociación
9.6.5. Construcción de clases con herencia
9.6.6. Construcción de clases con herencia múltiple
9.6.7. Definición de clases abstractas
9.6.8. Construcción de clases con herencia incluyendo polimorfismo
9.7. Empleo de excepciones
9.8. Gestión de eventos
9.8.1. Eventos, fuentes y auditores de eventos

9.8.2. Tipos de eventos. Mecanismos de gestión de eventos
9.8.3. Librerías de clases asociadas
9.9. Empleo de hilos
9.9.1. Fundamentos
9.9.2. Creación
9.9.3. Prioridad
9.9.4. Comunicación
9.9.5. Sincronización
9.9.6. Estados
9.9.7. Creación y ejecución de hilos en el lenguaje
9.9.8. Librerías de clases asociadas
9.9.9. Programación multihilo
9.10. Definición y análisis de programación en red
9.10.1. Aplicaciones cliente servidor
9.10.2. Sockets
9.11. Acceso a bases de datos desde las aplicaciones. Librerías de clases asociadas
Ejercicios resueltos

Bibliografía

Apéndice A: palabras reservadas

Paraninfo Colombia Calle José Abascal, 56 (Utopicus). Oficina 217. 28003 Madrid (España)
Tel. (+34) 914 463 350 Fax
clientes@paraninfo.co www.paraninfo.mx