

Plan 301 Ing.Tec.Informática de Gestión

Asignatura 16508 PROGRAMACION III

Grupo 1

Presentación

La asignatura tiene como principal objetivo la presentación y estudio del paradigma de programación orientada a objetos.

La primera parte aborda los conceptos básicos del paradigma, clase, clase genérica y objetos, adentrándose, posteriormente, en la programación de diseño bajo contrato.

En la segunda parte se estudia en profundidad el concepto de herencia, incluida herencia múltiple, los diferentes tipos de herencia, así como los elementos primordiales del paradigma: jerarquías de clases, polimorfismo, ligadura dinámica y clases diferidas.

Como aplicación de los conocimientos estudiados, se introduce el concepto de patrones de diseño de programas. Se analizan en detalle un conjunto reducido de patrones representativos abriendo así, el camino para un posterior estudio en profundidad de las técnicas de análisis y diseño orientado a objetos, que serán objeto de estudio de posteriores asignaturas.

Programa Básico

Programación Orientada a Objetos. Diseño bajo contrato. Excepciones. Herencia. Genericidad. Técnicas avanzadas de herencia. Patrones de diseño.

Objetivos

Los objetivos generales de la asignatura en el orden propio de su desarrollo son:

- Conocer el paradigma de la programación orientada a objetos como un lenguaje de programación orientado hacia la estructura del problema.
- Estudiar los conceptos básicos de la programación orientada a objetos: clases, clases genéricas y objetos.
- Analizar el concepto de CLASE como expresión del concepto matemático TAD en la programación orientada a objetos.
- Analizar el concepto de CLASE GENÉRICA como elemento base de reutilización.
- Analizar el concepto de OBJETO como expresión del elemento de información básico principal en ejecución.
- Estudiar la capacidad de las clases para desarrollar programación bajo contrato.
- Profundizar en la capacidad de las clases para definir soluciones genéricas.
- Conocer el concepto de "herencia" como técnica principal y diferencial de la programación orientada a objetos.
- Estudiar el concepto de "jerarquía de clases". Presentar diferentes jerarquías de clases utilizadas por lenguajes de programación visuales para el desarrollo de programas.
- Comprender la capacidad de la herencia y la definición de jerarquías de clases como elementos primordiales de reutilización y mantenimiento de programas.
- Estudiar los conceptos fundamentales del paradigma de objetos: "polimorfismo", "ligadura estática" y "ligadura dinámica".
- Estudiar el concepto de "clases diferidas" como definición de comportamientos base de las jerarquías de clases.
- Analizar y aplicar el mecanismo de control de excepciones.
- Conocer el concepto de "herencia múltiple", sus ventajas e inconvenientes.
- Analizar los conceptos básicos de "Agregación" y "Asociación" y su relación con la herencia.
- Clasificar y diferenciar los tipos de herencia.
- Presentar el concepto y formato de los "patrones de diseño".
- Conocer los patrones de diseño básicos: patrones de creación, patrones estructurales y patrones de comportamiento.
- Estudiar en profundidad un conjunto de patrones de diseño que identifiquen con claridad todos los conceptos definidos en la asignatura: patrón singleton, patrón builder, patrón compuesto, patrón visitador y patrón del observador.

1. Introducción.

1. Factores internos y externos de calidad.
2. Criterios, reglas y principios de modularidad.
3. Reutilización del software. Propuestas.
4. La propuesta de la orientación a objetos.

2. La estructura estática: Clases.

1. El papel de las clases en orientación a objetos.
2. Un sistema de tipos uniforme.
3. El estilo de programación orientado a objetos.
4. Ocultación de la información.

3. La estructura dinámica: Objetos.

1. Objetos.
2. Objetos y referencias.
3. Creación y destrucción de objetos.
4. Operaciones sobre referencias.
5. Objetos compuestos y tipos expandidos.
6. Ligadura por referencia y por valor.
7. Trabajando con referencias, ventajas e inconvenientes.
8. La gestión de la memoria.

4. Introducción a la herencia.

1. Importancia de la herencia en el paradigma de orientación a objetos.
2. Polimorfismo.
3. Ligadura estática y dinámica.
4. Clases y características diferidas: Frameworks.
5. Técnicas de redeclaración.
6. Herencia y aserciones.
7. Herencia y ocultación de la información.
8. El significado de la herencia.
9. El problema de los tipos.

5. Genericidad.

1. Generalización horizontal y vertical.
2. La necesidad de parametrización de tipos.
3. Clases genéricas.

6. Herencia múltiple.

1. Ejemplos de Herencia múltiple.
2. Problemática en la herencia múltiple.
3. Renombrado de características.
4. Estructura global de la herencia.
5. Herencia repetida.
6. Conflictos
7. Genericidad restringida.
8. Tipos y redeclaración.

7. Aspectos metodológicos de la O.O.

1. Introducción.
 2. Cómo encontrar las clases.
-

3. Principios de diseño de clases.
4. Tipos de herencia y su clasificación.
5. Metaclases.

8. Diseño bajo contrato.

1. Mecanismos básicos de fiabilidad.
2. La corrección del software.
3. Aserciones: Pre y postcondiciones.
4. Invariantes de clase.
5. Invariantes y variantes de bucles.
6. Manejo de excepciones.

9. Técnicas de diseño: aplicaciones de la herencia

1. Introducción.
2. Patrones de diseño: formato y clasificación.
3. Ejemplos de patrones de diseño.
4. Pruebas específicas O.O.

Programa Práctico

Las sesiones prácticas se desarrollan paralelamente a las sesiones teóricas, complementando a las mismas. El programa de prácticas se subdivide en dos tipos de actividades paralelas:

1. Prácticas en laboratorio bajo la dirección, control y supervisión del profesor. Durante cada sesión de laboratorio se desarrollará una práctica intrínsecamente relacionada con las sesiones de teoría desarrolladas en las fechas más inmediatas, con objeto de que el alumno sea capaz de relacionar los conceptos teóricos con la implementación práctica, empleando un lenguaje de programación orientado a objetos.
2. Prácticas evaluables a desarrollar por los alumnos con entrega obligatoria en la fechas propuestas. Se propondrán tres prácticas obligatorias que abarcarán los conceptos que para cada una se indican, y se desarrollarán con posterioridad a la explicación teórica de los conceptos involucrados así como al desarrollo de prácticas indicadas en el apartado anterior.

- Práctica 1ª.- Desarrollo de una aplicación simple que estudie los conceptos de clases y objetos.
- Práctica 2ª.- Desarrollo de una aplicación basada en los conceptos de herencia, jerarquías de clases, polimorfismo, ligadura estática y dinámica.
- Práctica 3ª.- Desarrollo de una aplicación que utilice al menos uno de los patrones de diseño presentados en las sesiones teóricas.

Evaluación

La evaluación de la asignatura se realiza en los aspectos práctico y teórico. La evaluación práctica se desarrolla analizando y valorando los conceptos siguientes:

1. Reglas de estilo: Evalúa la normativa de entrega de prácticas desarrollada para la asignatura consistente principalmente en un claro, concreto y preciso desarrollo de la documentación subdividido en los apartados siguientes:
 - Diagrama modular.
 - Diagrama de clases.
 - Diseño y justificación de clases.
 - Diseño y especificación de pruebas.
 - Listado autodocumentado de módulos fuente.
2. Diseño de clases: Evalúa el diseño de las clases declaradas por el alumno para resolver el problema propuesto en los conceptos de:
 - Diseño concreto de cada clase como T.A.D. (tipo abstracto de datos) en sus acepciones de atributos y métodos.
 - Asociaciones y relaciones de clases para conformar la solución del problema.
3. Diseño de pruebas: Evalúa el diseño de las pruebas para verificar la corrección de la solución propuesta en sus apartados:
 - Pruebas de clase: que verifiquen la corrección del TAD propuesto por la clase.
 - Pruebas de módulo: que verifiquen la corrección de la propuesta de diseño arquitectónico propuesto.
 - Pruebas de aplicación: que verifiquen la corrección de la integración modular para resolución completa del problema.
4. Diseño de módulos: Evalúa el diseño de la solución arquitectónica propuesta:

-
- Correspondencia directa entre la estructura del problema y la estructura de la solución.
 - Número y estructura de las interfaces propuestas.
 - Ocultación de la información.
5. Diseño y desarrollo de métodos: Evalúa la programación concreta de cada uno de los métodos diseñados para cada clase considerando principalmente:
- Sencillez.
 - Legibilidad.
 - Reusabilidad.
 - Eficiencia.
6. Corrección de la ejecución: Evalúa la presentación y correcta ejecución de la solución proporcionada por el alumno. Para cada práctica propuesta se entrega al alumno un conjunto de ficheros de prueba que permiten verificar la corrección de la ejecución.

La evaluación teórica se desarrolla valorando los conceptos siguientes:

1. Clases: En cuestiones breves y ejercicios propuestos.
2. Objetos: En cuestiones breves y ejercicios propuestos.
3. Herencia: En cuestiones breves y ejercicios propuestos.
4. Patrones de diseño: En ejercicios propuestos.

Los criterios de evaluación utilizados para los aspectos teóricos y prácticos es el siguiente:

- Prácticas: Cada uno de los conceptos de evaluación práctica citados anteriormente deben superarse suficientemente para considerar aprobada la práctica, en otro caso la práctica debe ser realizada nuevamente por el alumno. La escala de calificaciones de cada uno de los conceptos se subdivide en:
 - a. MB: El concepto ha sido muy satisfactoriamente superado con una calificación de 10.
 - b. B: El concepto ha sido satisfactoriamente superado con una calificación de 8.
 - c. R: El concepto ha sido suficientemente superado con una calificación de 5.
 - d. M: El concepto no ha sido superado con una calificación de suspenso.

La calificación final de una práctica se realiza como la media aritmética de las calificaciones de cada uno de los conceptos evaluados.

La entrega de prácticas tiene asociada una penalización por entrega fuera de plazo que modifica el valor máximo de la calificación 10 para las entregas en plazo, a una calificación máxima de 7 para entregas con una semana de retraso y una calificación máxima de 5 para entregas posteriores.

La calificación final de prácticas se obtiene asignando a cada práctica un valor ponderado según su nivel y grado de dificultad. La valoración de prácticas del curso actual ha sido: $(P1 * 0.2 + P2 * 0.4 + P3 * 0.8) / 1.4$.

- Teoría: La evaluación teórica se desarrolla con los criterios siguientes:
 - a. Cada pregunta o ejercicio del examen teórico lleva asociado su calificación máxima.
 - b. Las cuestiones breves se califican con el criterio de bien o mal obteniendo una calificación máxima o cero.
 - c. Los problemas propuestos se califican tomando como base la solución propuesta por el profesor con los criterios:
 - i. Si la solución propuesta por el alumno coincide con la solución propuesta por el profesor o se ajusta a la metodología desarrollada en la asignatura se obtendrá la calificación máxima.
 - ii. Si la solución propuesta por el alumno es válida para resolver el problema pero no se ajusta a la metodología desarrollada en la asignatura se obtendrá una calificación máxima del 50% del valor del ejercicio.
 - iii. En otro caso la calificación será cero.

La calificación final de la asignatura se realiza con los criterios siguientes:

- La asignatura se aprueba superando con al menos un cinco en la parte teórica y la parte práctica de la asignatura, siendo condición necesaria tener aprobada la parte práctica con anterioridad a la evaluación teórica.
- La nota final se determina como la media ponderada de la calificación teórica con la calificación práctica con un criterio de 60% - 40% respectivamente.

Bibliografía
