

UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES

1. DEPARTAMENTO: Computación
2. CUATRIMESTRE: Primero de 1997.
3. ASIGNATURA: PARADIGMAS DE PROGRAMACION
4. CARRERA: Licenciatura en Ciencias de la Computación
5. CARACTER DE LA MATERIA: Obligatoria
6. NUMERO DE CODIGO DE CARRERA: 18
7. NUMERO DE CODIGO DE MATERIA:
8. PUNTAJE: ----
9. PLAN DE ESTUDIOS AÑO: 1993.
10. DURACION DE LA MATRERIA: Cuatrimestral
11. HORAS DE CLASE SEMANAL:
 - a) TEORICAS 4 HS.
 - b) LABORATORIO HS.
 - c) PROBLEMAS 3
 - d) SEMINARIOS
12. CARGA HORARIA TOTAL: 7 HORAS
13. ASIGNATURAS CORRELATIVAS: Algoritmos y Estructura de Datos II - Lógica y Computabilidad
14. FORMA DE EVALUACION: Examen Final
15. PROGRAMA Y BIBLIOGRAFIA: Adjuntas a esta hoja

FECHA: 19/7/97

PIA
Firma del Profesor
Lic. Graciela MATICH

LOISEAU
Firma del Director
Lic. Irene LOISEAU

Especificaciones Genéricas

Materia: Paradigmas de Lenguajes de Programación

Carrera: Licenciatura en Ciencias de la Computación

Departamento: Computación

Carácter de la Materia: Obligatoria para Lic. Cs Comp. Plan 93

Puntaje como Optativa: XXXXXXXXXX

Asignaturas correlativas:

Para Lic. en Ciencias de la Computación, plan 1993:

Algoritmos y Estructuras de Datos II

Lógica

Para Lic. en Ciencias de la Computación, otros planes:

Laboratorio IV

Teoría de Lenguajes y Automatos

Objetivos: Analizar algunos paradigmas de lenguajes de programación, introduciendo sus conceptos principales y modalidad si fuese el caso. Evaluar los conceptos de lenguajes de programación en términos de su contribución al proceso de desarrollo de software y al desempeño para determinados propósitos. Desarrollar criterios necesarios para realizar dicha evaluación.

Contenidos: Los diversos paradigmas de lenguajes de programación: imperativo, lógico, funcional, orientado a objetos. Estudio de conceptos comparados en lenguajes de programación como elemento fundamental para el proceso de evaluación de lenguajes.


Duración de la Materia: Cuatrimestral

Modalidad: Teórico - Práctica

Evaluación: Parciales y Final. Proyectos de evaluación de lenguajes.

Carga Horaria Semanal: 7hs

Total de Horas: 112hs


Lic. Graciela H. Matich


Lic. IRENE LOISEAU
DIRECTORA
DEPTO. DE COMPUTACION
F. C. E. y N. UBA

Paradigma Funcional

- Estructura e importancia de los Lenguajes Funcionales.
- Fundamentos: Cálculo Lambda
- Análisis y práctica en algún lenguaje representante de este paradigma (Hope o ML). Sintaxis y conceptos básicos. Primitivas Definición y llamada de funciones.
- Evolución de los lenguajes funcionales (Miranda, Gofer)
- Rasgos de los lenguajes funcionales más modernos.

Paradigma Orientado a Objetos

- Ambito de aplicación: Programación en gran escala
- Conceptos relevantes de Lenguajes Orientados a Objetos (LOOs)
Objetos _ Clases _ Herencia
- Amplio espectro dentro de los llamados LOOs
- Principios y polémicas sobre LOOs
- Algunos representantes: Smalltalk, Eiffel.
Otros representantes. Coincidencias y divergencias
- Evolución y tendencias

Parte II _ Conceptos Comparados en Lenguajes de Programación

Enfoque semántico de los lenguajes de programación

- Unidades de programa
- Ambiente de referencia
- Algunos modelos de implementación

"BINDING" : Amarre o

Resolución de vínculos entre entidades y atributos

- Amarre estático y dinámico ("static vs dynamic binding")
- Lenguajes con estructura de bloques.
Alcance ("scope") y visibilidad
- Declaraciones . Alcance de las declaraciones

Locación de memoria

- Locación de memoria en formas estática y dinámica
- Locación en el "HEAP"
- Sistemas de compactación y recolección de basura


Lic Graciela H. Matich


Lic. IRENE LOISEAU
DIRECTORA
DEPTO. DE COMPUTACION
F. C. E. y N. UBA

Paradigmas de Programación

Introducción

- Criterios para el estudio, análisis y proyecto de lenguajes.
- Importancia de estudiar conceptos comparados en lenguajes de programación para la formación de criterios.
- Peso de propiedades. Ambitos o entornos de aplicación.
- Presentación de los paradigmas a estudiar:
 - Paradigma Imperativo
 - _ Paradigma Lógico
 - _ Paradigma Funcional
 - _ Paradigma Orientado a Objetos

Parte I _ Análisis de algunos Paradigmas en Lenguajes de Programación

Paradigma Imperativo

- Peso histórico y cuantitativo de los lenguajes imperativos.
Orígenes y evolución.
- Puntos relevantes para su análisis: mecanismos de abstracción, tipificación, modularidad.
- Posibilidad de efectos colaterales.
- Programación en gran escala versus programación en pequeña escala
 - _ Programación Modular
 - _ Criterios de reusabilidad
- Panorama de Ada y Modula II

Paradigma Lógico

- Fundamentos y origen: Demostración de Teoremas
- Introducción a la Programación en Lógica
 - _ Sintaxis y semántica de los programas en lógica
 - _ Inferencia lógica
 - _ Unificación
 - _ Representante principal: el lenguaje Prolog
 - Algunos ejemplos
- Intérprete no determinista
- Ambitos de este paradigma: Sistemas Expertos _ Bases de Datos
Procesamiento del Lenguaje Natural



Mecanismos de Abstracción

- Mecanismos de abstracción en control
 - _ Iteradores
- Mecanismos de abstracción en datos

Sistemas de Tipos

- Verificación de tipos en forma estática y dinámica
- Lenguajes con sistemas fuertemente tipificados
("strong typing systems")
- Monomorfismo
- Polimorfismo
- Sobrecarga ("overloading")
- Tipos genéricos ("Generics")

Mecanismos de Herencia

- Herencia basada en moldes
- Compartir atributos en forma dinámica : Delegación

Encapsulamiento

- Ocultamiento de información
- Tipos (abstractos) de datos
- Objetos y clases
- Herencia y encapsulamiento


Excepciones

- Tratamiento de condiciones de excepción ("exceptions") en diferentes lenguajes: CLU, Ada, Eiffel.

Conclusiones

- Balance de rasgos y propiedades en lenguajes de programación
- Aportes oportunos de cada paradigma. Mención de otros paradigmas.
- Nuevos enfoques: Dilución de fronteras entre paradigmas


Lic Graciela H. Matich


Lic. IRÈNE LOISEAU
DIRECTORA
DEPTO. DE COMPUTACION
F. C. E. y N. UBA