Facultad de Ingeniería - Universidad Nacional de Cuyo P1- PROGRAMA DE ASIGNATURA			
Asignatura:	Paradigmas de Programación		
Profesor Titular:	Dr. Pablo Javier Vidal		
Carrera:	Licenciatura en Ciencias de la computación		
Año: 2022	**Semestre: 4to**	**Horas Semestre: 112**	**Horas Semana: 7**

OBJETIVOS

- Reconocer los fundamentos, características y diferencias de los distintos paradigmas de programación que son utilizados por los lenguajes de programación actuales.
- Identificar el paradigma que resulta más adecuado de aplicar para resolver diferentes tipos de problemas.
- Reconocer el modelo formal o semiformal subyacente de cada paradigma y la forma en que el mismo es incorporado en un lenguaje de programación correcto
- Adquirir las nociones básicas de los modelos de programación orientada a objetos, lógico y funcional.
- Aplicar los diferentes paradigmas en la resolución de problemas.

CONTENIDOS

UNIDAD 1: PARADIGMAS DE PROGRAMACIÓN

Concepto de paradigma de programación. Paradigmas fundamentales. Comparación Programa: definiciones generales y específicas. Abstracción y modularización. Traductores: compilador e interprete. Tipo: representación e importancia de los tipos en los diferentes lenguajes de programación. Chequeo de tipos. Ubicación de los mecanismos de control de flujo en un programa: definición y mecanismos de implementación. Concepto de ligaduras. Tipos de Ligaduras. Utilización de las variantes del polimorfismo en los diferentes paradigmas. Comparación entre paradigmas de programación.

UNIDAD 2: PARADIGMA LÓGICO

Programación declarativa y la programación lógica. Fundamentos Teóricos. Lógica de Primer Orden y formas restringidas. Cláusulas de Horn. Unificación. Aritmética y Operadores. Principio de resolución SLD. Tipos de Derivaciones SLD. Árbol de derivación SLD. Retroceso Cronológico. Operador de corte. Entidades y Ligaduras.

Lenguaje asociado: Implementación de programas en Prolog. Sintaxis y semántica del lenguaje. Realización de consultas. Trabajo con listas.

UNIDAD 3: PARADIGMA FUNCIONAL

Paradigma funcional. Concepto de función. Transparencia referencial. La función como bloque

de construcción de programas. Concepto de programa en el paradigma funcional. Efecto de lado. Concepto de variable. Definición de tipo y valor. Funciones definidas por ramas. Pattern matching. Funciones recursivas. Inferencia de tipos. Sistemas de tipos. Manejo de listas. Listas por comprensión/extensión. Funciones de orden superior. Currificación y aplicación parcial de funciones. Evaluación diferida y listas infinitas. Polimorfismo y tipos genéricos. Tuplas. Entidades y Ligaduras. Calculo lambda.

Lenguaje asociado: Haskell. Entorno de trabajo, definición de programas, uso del intérprete. Notación bidimensional. Módulos. Notación de listas [n..m]. Notación de listas por comprensión. Operadores infijos y prefijos. Reglas de precedencia. Prelude de Haskell. Funciones incorporadas en el prelude para manejo de listas, de tuplas, de funciones de orden superior. Entidades y ligaduras.

UNIDAD 4: PARADIGMA ORIENTADO A OBJETOS

Concepto de Objeto, métodos y mensaje, estado y comportamiento. Encapsulamiento y ocultamiento. Interfaz e implementación. Concepto de Clase como modelo/molde de objetos. Delegación y responsabilida. UML. Variables y métodos de clase. Herencia. Subclases. Clases abstractas. Polimorfismo. Redefinición de métodos. Relaciones entre clases: asociación, composición; relación con delegación. Aplicación del concepto de tipo en el paradigma de objetos. Efecto de lado. Orden superior. Introducción al manejo de errores. Persistencia de objetos. Colecciones. Concurrencia.

Lenguaje asociado: Java. Características generales, sintaxis y semántica del lenguaje. Estructura de programas. Clases. Constructores. Encapsulamiento y modularidad. Instanciación y destrucción de objetos. Compilación y ejecución. Variables y métodos. Modificadores de acceso. Lanzamiento, captura y tratamiento. Excepciones. Herencia e Interfaces. Polimorfismo. Colecciones Entidades y ligaduras.

METODOLOGÍA DE ENSEÑANZA

Por tratarse de una asignatura con importante respaldo conceptual, y para facilitar la apropiación de los conocimientos, se aplicará la ejemplificación y demostración de manera metódica. Los contenidos del programa se presentan y analizan en las clases teóricas y se consolidan con los trabajos prácticos. Existe una estrecha relación entre la teoría y práctica, a su vez la práctica fundamenta sus procedimientos en la teoría.

Los ejercicios están orientados al análisis o entendimiento de las soluciones propuestas y a la concepción o diseño para responder a nuevos requerimientos y problemas. Para ello el material de práctica es seleccionado de manera de cubrir un amplio espectro temático y muestra diferentes niveles de dificultad. Algunos ejercicios son resueltos completamente en clase, y otros son encaminados para ser completados por los alumnos fuera de clase.

Estrategias metodológicas y de enseñanza

Las siete horas semanales que tiene asignada la materia se distribuyen de la siguiente manera:

- Clases teóricas de cuatro horas para presentación y exposición de los temas, con el uso de pizarrón con apoyo de proyección de imágenes y computadora. Dichas clases incluyen gran contenido de forma práctica y serán participativas y de debate. Se

desarrollarán de manera expositiva los conceptos fundamentales, con presentación de casos de estudio prácticos de aplicación inmediata del tema presentado. Se promoverá la investigación y la búsqueda de información por parte del alumno. Se exigirá un trabajo continuo al alumno tanto dentro del aula como fuera de ella.

- Clases prácticas de tres horas de ejercitación para resolución de problemas y consultas trabajando en el aula. Estas clases incluirán presentación del tema, casos de estudio abiertos a debate y de construcción colectiva, y consulta individual de ejercicios. Se incluyen actividades de laboratorio para aplicar diferentes lenguajes de programación (Python, Java, Prolog, Haskell) como herramienta instrumental de resolución de problemas de acuerdo al paradigma que se esté estudiando.

Actividad	**Carga horaria por semestre**
Teoría y resolución de ejercicios simples	64
Formación práctica	
Formación Experimental – Laboratorio	48
Formación Experimental - Trabajo de campo	0
Resolución de problemas de ingeniería	0
Proyecto y diseño	0
Total	**112**

BIBLIOGRAFÍA

Bibliografía básica

Autor	Título	Editorial	Año	Ejemplares en biblioteca
T.W. Pratt, M.V. Zelkowitzs	Programming Languages - Design and Implementation, 4º edición	Prentice Hall	2000	
Deitel&Deitel	Como Programar en Java, 7ma edición	Prentice Hall	2008	
R. Bird	Introducción a la Programación Funcional con Haskell	Prentice Hall	2000	
W.D. Burnham, A.R. Hall	Prolog, Programación y Aplicaciones	Limusa	1990	
Allen B. Tucker, Rober E. Noonan	Programming Languages_ Principles and Paradigms	McGraw-Hill	2007	
Aho, Alfred V., Ravi Sethi, Jeffrey D. Ullman	*Compiladores: principios, técnicas y herramientas*	Pearson Educación	2008	

Bibliografía complementaria

Autor	Título	Editorial	Año	Ejemplares en biblioteca
John S. dean, Raymond H. Dean	Introducción a la Programación con Java	McGraw-Hill	2009	
H. Schildt	Java 7 - Oracle Press	ANAYA	2012	

EVALUACIONES (S/ Ord. 108-10_CS)

Se realizará una evaluación integral y continua durante el cursado. **Se regirá por la Ord. 108/10 C.S. UNCUYO**, explicitando claramente en la planificación didáctica del Espacio Curricular el régimen de acreditación y promoción, según Artículos. 7-8 y 9 de la mencionada norma. Se tendrá en cuenta en la evaluación las características de validez, pertinencia y consistencia con la propuesta de enseñanza y aprendizaje realizada.

Evaluaciones durante el cursado

Para la evaluación continua en el aula se considerarán la resolución de problemas, trabajos prácticos, laboratorios y evaluaciones parciales, con observación del desempeño individual y grupal y retroalimentación constante. La evaluación incluirá propuestas de autoevaluación que posibiliten la reflexión del alumno como protagonista de su proceso de aprendizaje.

Condición de regularidad tras el cursado

Para adquirir la condición de alumno regular se deberá cumplir con los siguientes requisitos:

- Aprobar dos evaluaciones parciales teóricas y dos evaluaciones practicas, cada uno de ellos consta de puntaje mínimo de 1 (UNO) y un máximo de 10 (DIEZ). Para la obtención de la regularidad es necesario aprobar cada evaluación (teórica y practica) con un mínimo de 6 (SEIS).
    - Primer parcial: Unidad 2 y 3
    - Segundo parcial: Unidad 4
    - Laboratorio 1: Unidad 2 y 3
    - Laboratorio 2: Unidad 4
- Aquellos alumnos que desaprueben todos o cualquiera de las evaluaciones parciales (teóricas o practicas) podrán rendir un examen recuperatorio global con los mismos temas del examen parcial que desaprobó. En el caso de los laboratorios, existirá un laboratorio global para recuperar en caso de desaprobar alguno o los dos laboratorios. Cualquier examen recuperatorio global se aprueba con 6 (SEIS) como mínimo sobre un total de 10 (DIEZ).
- Las evaluaciones son teóricas/prácticas, y cada examen quedará como respaldo documental de lo elaborado por el alumno, posibilitando cualquier revisión posterior que pueda ser necesaria por el mismo.

Evaluación final

Para la aprobación de la asignatura del estudiante en condición regular, se realizará una evaluación calificativa que apunta a identificar cuánto ha aprendido un alumno en este espacio curricular con el fin de tomar la decisión de acreditación de saberes. Constituye una evaluación que refleje el trabajo del estudiante, comprensivo y fundamentado, y brinda al docente, una mirada global del proceso de aprendizaje. El examen comprende una instancia escrita con preguntas prácticas sobre los tópicos de la asignatura para luego pasar a una instancia oral para evaluar la capacidad de la aplicación de saberes teóricos.

Examen final para alumnos libres – libres por pérdida de regularidad.

El mismo tiene dos fases, la primera de ellas es un examen escrito con ejercicios prácticos integrales de las diferentes unidades y se aprueba con un mínimo de 6 (SEIS) sobre un total de 10 (DIEZ). Una vez aprobado el examen escrito, el alumno libre deberá rendir un examen oral según los criterios de evaluación mencionados previamente.

Alumnos recursantes.

No hay régimen especial para alumnos recursantes.

FECHA, FIRMA Y ACLARACIÓN TITULAR DE CÁTEDRA