PROGRAMA DE ASIGNATURA

|  |  |  |
| --- | --- | --- |
| **ASIGNATURA:**  Lenguajes y Compiladores | | **AÑO**: 2019 |
| **CARÁCTER:** Obligatoria | | |
| **CARRERA:** Licenciatura en Ciencias de la Computación | | |
| **RÉGIMEN:** cuatrimestral | **CARGA HORARIA:** 120 hs. | |
| **UBICACIÓN en la CARRERA:** Quinto año – Primer cuatrimestre | | |

|  |
| --- |
| **FUNDAMENTACIÓN Y OBJETIVOS** |

El establecer el significado de las frases de un lenguaje de programación es un problema de múltiples aristas en tanto puede tener variados objetivos, que van desde la necesidad de comprensión humana, hasta el imperativo de que una máquina los pueda interpretar o traducir a una secuencia de instrucciones ejecutables. Un manual de usuario, una sofisticada estrategia de compilación, o alguna herramienta teórica destinada a desentrañar los principios básicos de su diseño, constituyen todas vertientes de significado que responden a distintos intereses y usos de los lenguajes de programación. En las últimas décadas variados desarrollos matemáticos y lógicos dieron forma a una teoría que se posicionó en un lugar privilegiado para el acceso a la comprensión profunda del significado de un lenguaje. La misma permite conectar la descripción intuitiva de un sentido finito y dinámico (un manual), con una modalidad estática del significado, vigente en la lógica formal y la matemática (denotación). A partir del desarrollo de la Teoría de Dominios la semántica denotacional adquiere un relevancia especial, no sólo por tratarse de objetos matemáticos perfectamente definidos en el contexto de una teoría particular, sino además porque comienza a ser utilizada como "la definición" del lenguaje y luego, si se proponen otras semánticas (operacional, axiomática), se las demuestra correctas con respecto a dicha definición.

El objetivo general de la asignatura es lograr que los estudiantes se apropien de las herramientas más importantes que actualmente se utilizan para dar significado a las frases de un lenguaje de programación, poniendo énfasis en la utilidad de estas herramientas para comprender los principios básicos que subyacen en su diseño.

Dentro de los objetivos específicos, mencionamos como relevantes:

* tomar contacto con un lenguaje teórico basado en el Standard ML, en tanto lenguaje que ha sido definido formalmente de manera completa, y cuyos principios básicos coinciden con los lenguajes más populares
* el acceder al uso de herramientas matemáticas apropiadas
* disponer de recursos para evaluar las características principales de los distintos lenguajes actualmente en uso
* reconocer propiedades deseables en lenguajes de programación y las herramientas para garantizarlas
* proveer de recursos para que el estudiante pueda diseñar e implementar lenguajes de programación

|  |
| --- |
| **CONTENIDOS** |

**Unidad I** *Herramientas básicas para dar significado a los lenguajes de programación*

* Distintas formas de dar significado a los lenguajes de programación. Semántica denotacional: las nociones de frase, dominio semántico y función semántica. Semántica operacional: las nociones de configuración, regla de transición y ejecución.
* Nociones en relación a la sintaxis: gramática, gramática abstracta, sintaxis abstracta, lenguaje y metalenguaje.
* Nociones en relación a la definición del significado: dirección por sintaxis, semántica composicional.
* Variables y ligadura (en la lógica de predicados). Sustitución y el problema de la captura. Propiedades de coincidencia y renombre.
* El problema de dar significado a las funciones recursivas. Dominios, función continua y teorema del menor punto fijo. Análisis de las soluciones de una ecuación recursiva a la luz del teorema del menor punto fijo.

**Unidad II** *Lenguajes imperativos.*

* Conjunto de estados. Semántica denotacional de las construcciones básicas de un lenguaje imperativo.
* El problema de dar significado a la iteración. Significado de la iteración utilizando el teorema del menor punto fijo.
* Propiedades de coincidencia y renombre.
* Fallas y manejo de excepciones. Output. Input.
* Semántica operacional para el lenguaje imperativo. Corrección respecto de la semántica denotacional.

**Unidad III** *Lenguajes aplicativos*.

* Las nociones de reducción y evaluación en el Cálculo Lambda. El problema de la terminación. La noción de forma canónica. Modalidad de evaluación: Eager y Normal.
* El problema de la semántica denotacional: el modelo D infinito y sus variantes.
* Lenguaje aplicativo. Sintaxis. Semántica operacional eager y normal: la noción de evaluación, formas canónicas y reglas de evaluación. Tratamiento de errores.
* Semántica denotacional directa del lenguaje aplicativo. Sintaxis y semántica de la recursión en las modalidades eager y normal. Propiedades.

**Unidad IV** *Combinación de paradigmas.*

* Los problemas de la combinación de paradigmas.
* Las nociones de estado, ambiente, identificador y variable.
* Un lenguaje que combina los paradigmas. Semántica denotacional y operacional.
* Construcciones imperativas como abreviaturas. Propiedades.
* Funciones y procedimientos. Pasaje de parámetros.

**Unidad V** *Otros tópicos de interés*

* Semántica de continuaciones.
* Tuplas, patrones y unión disjunta en los lenguajes aplicativos puros.

|  |
| --- |
| BIBLIOGRAFÍA |

**BIBLIOGRAFÍA BÁSICA**

* Fridlender, D., Gramaglia H. Apuntes de Cátedra (basados en el libro de Reynolds).
* Reynolds, Theories of Programming Languages, Cambridge University Press, 1998.

**BIBLIOGRAFÍA COMPLEMENTARIA**

* Tennet, R., Semantic of Programming Languages, Prentice Hall.

|  |
| --- |
| **METODOLOGÍA DE TRABAJO** |

El desarrollo de los contenidos se basa en la modalidad de exposición, empleando en este tipo de actividad un 50% del tiempo de aula. En las mismas se estimula la participación de los alumnos a través del análisis de casos de estudio y preguntas, vinculando la actividad con la práctica.

Para el desarrollo de actividades prácticas se trabaja con una guía de ejercicios adoptando una modalidad de resolución grupal o individual según la preferencia del alumno. Se trabaja bajo la orientación de los docentes respetando los tiempos de cada grupo o individuo.

En las clases teóricas y prácticas se presentan ejercicios ejemplificadores para su discusión en clase. La guía de práctica consta por un lado de una serie de ejercicios que tiene por objetivo afianzar los contenidos trabajados en el teórico, y por otro de un conjunto de problemas más complejos que fomentan la integración de todos los saberes, incentivando el pensamiento analítico y crítico.

|  |
| --- |
| **EVALUACIÓN** |

**FORMAS DE EVALUACIÓN**

* Se tomarán 2 (dos) exámenes parciales teórico-prácticos. Las evaluaciones parciales serán sobre contenidos teórico-prácticos.
* Si la cátedra lo considera necesario se podrán incorporar otras instancias de evaluación formativa.
* La materia contempla un régimen de promoción.
* La aprobación de la materia se dará por promoción, o mediante la aprobación de un examen final en las fechas destinadas a exámenes en el calendario académico. El examen final contará de una evaluación escrita sobre contenidos teórico-prácticos, complementada con una instancia oral si el tribunal lo considera necesario.

**CONDICIONES PARA OBTENER LA REGULARIDAD**

Aprobación de 2 exámenes parciales, con calificación mayor o igual a 4. Se podrá recuperar uno de los dos parciales en la última semana de dictado de la materia.

**CONDICIONES PARA OBTENER LA PROMOCIÓN**

Aprobación de los 3 exámenes parciales, con una calificación mayor o igual a 7. Aprobación del taller.