



Ministerio de Educación
Universidad Tecnológica Nacional
Facultad Regional Rosario

"2022 - Las Malvinas son argentinas"

Rosario, 21 de diciembre de 2022.-

VISTO el Expediente ID N°: 8141987, relacionado con la presentación del Programa Analítico de la asignatura "Sintaxis y Semántica de los Lenguajes", correspondiente a la carrera Ingeniería en Sistemas de Información, y

CONSIDERANDO

Que la presentación realizada obedece a la implementación del nuevo Diseño Curricular aprobado por el Consejo Superior de la Universidad Tecnológica Nacional – Ordenanza CSU N° 1877.

Que dicho Programa Analítico cuenta con el aval del respectivo Consejo Departamental.

Por ello y atento a las atribuciones otorgadas por el artículo 85° del Estatuto Universitario.

**EL CONSEJO DIRECTIVO DE LA FACULTAD REGIONAL ROSARIO
DE LA UNIVERSIDAD TECNOLÓGICA NACIONAL**

RESUELVE:

ARTÍCULO 1°.- Aprobar el Programa Analítico de la asignatura "Sintaxis y Semántica de los Lenguajes" para el Segundo Nivel de la carrera Ingeniería en Sistemas de Información – Plan 2023, que se agrega como Anexo I de la presente resolución.

ARTÍCULO 2°.- Regístrese. Comuníquese. Elévese. Publíquese. Cumplido, archívese.

RESOLUCIÓN N° 539

UTN
FRRo
C.D.
S.R.

Ing. Rubén Fernando CICCARELLI
Decano

Ing. Antonio Luis MUIÑOS
Secretario Académico


Carrera: Ingeniería en Sistemas de Información
Asignatura: Sintaxis y Semántica de los Lenguajes
PROGRAMA ANALÍTICO
1. Datos administrativos de la asignatura

Nivel en la carrera:	2	Dictado:	Cuatrimestral
Plan de Estudio:	2023	Área:	Desarrollo de Software
Bloque curricular:	Tecnologías Básicas	Electiva:	NO
Carga horaria presencial semanal (hs. cátedra):	8	Carga Horaria total anual (hs. reloj):	96
Carga horaria no presencial semanal (hs. reloj) (si correspondiese)	-	% horas no presenciales (hs. reloj) (si correspondiese)	-

2. Presentación, Fundamentación

La presente asignatura tiene carácter teórico-práctico. Transita desde el marco teórico abstracto de los lenguajes formales y los modelos de computación hasta la solución de problemas de ingeniería mediante el desarrollo de programas software, pasando por la exposición de los conceptos centrales de un lenguaje de programación y de su proceso de traducción, estableciendo una base conceptual clara y sólida para la enseñanza y el aprendizaje de las asignaturas del Área de Desarrollo del actual Diseño Curricular.

Ofrece al futuro Ingeniero en Sistemas de Información una perspectiva del amplio campo de conocimientos que fundamentan el desarrollo de herramientas de software que utilizará en el desempeño de su tarea profesional y que contribuyan a la solución de los problemas de ingeniería.

Contemporáneamente, la asignatura aporta a la formación del alumno al promover el aprendizaje de un lenguaje vastamente utilizado y su aplicación en problemas cuya resolución algorítmica le resulta novedosa.

La fundamentación expuesta está enmarcada principalmente dentro de la actividad reservada AR1: Especificar, proyectar y desarrollar sistemas de información, sistemas de comunicación de datos y

software cuya utilización pueda afectar la seguridad, salud, bienes o derechos, y del alcance AL4: Diseñar, desarrollar e implementar programas y actividades de innovación en procesos y productos relacionados con los sistemas de información, sistemas de comunicación de datos y software, del actual Diseño Curricular.

Finalmente, y en relación con su ubicación dentro del Plan de Estudios, por su énfasis en el empleo del paradigma imperativo estructurado de programación, fundamenta y amplía conceptos presentados en la materia Algoritmos y Estructuras de Datos que la precede (además de utilizar herramientas aprendidas en Lógica y Estructuras Discretas que también la precede), y representa un eslabón natural para abordar los paradigmas funcional, lógico y orientado a objetos que se exponen en la materia Paradigmas de Programación, que la sucede.

3. Contenidos Mínimos

- Gramática y Lenguajes Formales.
- Autómatas Finitos. Expresiones Regulares.
- Gramáticas Independientes del Contexto.
- Autómatas con Pila y Máquinas de Turing.
- Análisis Léxico, Sintáctico y Semántico.
- Traductores. Proceso de Traducción.

4. Objetivos establecidos en el DC

1. Comprender la sintaxis y semántica de los lenguajes de programación.
2. Comprender los fundamentos de los lenguajes formales, gramáticas y autómatas.
3. Emplear conceptos y procedimientos de las gramáticas libres de contexto y gramáticas regulares en la especificación de la sintaxis de los lenguajes de programación.
4. Diferenciar los procesos de traducción de los lenguajes.

5. Asignaturas correlativas previas

Para cursar y rendir debe tener cursada:

- Asignatura/s:
Lógica y Estructuras Discretas
Algoritmos y Estructuras de Datos

Para cursar y rendir debe tener aprobada:

- Asignatura/s:

Transcriba el nombre de la asignatura.

6. Asignaturas correlativas posteriores

Indicar las asignaturas correlativas posteriores:

- Asignatura/s que la requieren cursada:
Bases de Datos
- Asignatura/s que la requieren aprobada:
Ingeniería y Calidad de Software

7. Programa analítico

Este programa analítico contempla los contenidos mínimos, previstos en el DC vigente, y aquellos que se consideran necesarios para desarrollar los resultados de aprendizaje propuestos.

Unidad Nº: 1

Título: Traductores. Proceso de Traducción.

Contenidos: Traductores. Proceso General de Traducción. Tipos de Traductores. Diferencias entre Compilador e Intérprete.

Las fases de un compilador:

- .1. Analizador léxico. Implementación típica. Tokens, patrones y lexemas.
- .2. Analizador sintáctico.
- .3. Analizador semántico.
- .4. Generador de código intermedio.
- .5. Optimador de código intermedio.
- .6. Generador de código objeto.
- .7. Administrador de la tabla de símbolos.
- .8. Gestor de errores.

Tipos de errores léxicos, sintácticos y semánticos. Diferenciación y ejemplos.

Agrupamiento de las fases. Pasadas.

Unidad Nº: 2

Título: Lenguajes Formales.

Contenidos: Definiciones básicas sobre lenguajes formales. Símbolo. Alfabeto. Cadena. Cadena vacía. Partes de una cadena. Operaciones sobre cadenas. Lenguaje. Operaciones sobre lenguajes.

Unidad N°: 3

Título: Expresiones Regulares y Análisis Léxico.

Contenidos: Reglas de definición de una expresión regular. Operadores complementarios. Asociatividad de los operadores y precedencia. Lenguaje representado por una expresión regular. Conjuntos regulares. Equivalencia entre expresiones regulares. Propiedades algebraicas. Definiciones regulares. Uso de la expresión regular en el proceso del Análisis Léxico de un compilador.

Unidad N°: 4

Título: Automatas Finitos.

Contenidos: Automatas de estados finitos. Definición de autómata finito no determinístico (AFN). Grafo y tabla de transiciones. Definición de aceptación de una cadena por parte de un AFN. Definición de autómata finito determinístico (AFD). Propiedades.

Construcción de Reconocedores de un lenguaje regular:

- .1. Paso de una expresión regular a un AFN.
- .2. Pasaje de un AFN a un AFD equivalente.
- .3. Minimización de un AFD.
- .4. Ejemplos.

Unidad N°: 5

Título: Gramáticas Formales.

Contenidos: Definición de gramática formal. Derivaciones. Lenguaje generado por una gramática. Frase. Forma de frase. Jerarquía de Chomsky. Lenguajes formales y tipos de gramática:

- 0: Sin restricciones.
- 1: Sensible al contexto.
- 2: Independiente del contexto (GIC).
- 3: Regular (GR).

Relación entre lenguajes regulares, expresiones regulares y gramáticas regulares (GR).

Expresiones regulares frente a GIC.

Isomorfismo existente entre los tipos de gramática, los lenguajes formales de la Jerarquía de Chomsky y los distintos tipos de autómatas.

GIC y GR en la especificación de la sintaxis de los lenguajes de programación.

Unidad N°: 6

Título: Gramáticas Libres de Contexto y Análisis Sintáctico.

Contenidos: El proceso del análisis sintáctico. Gramáticas Libres de Contexto y notación BNF-EBNF. Árboles de análisis gramatical y árboles sintácticos abstractos.

Métodos de Análisis Descendente:

- Analizador sintáctico por descenso recursivo.
- Analizador sintáctico predictivo (por descenso recursivo sin retroceso).
- Analizador sintáctico predictivo y no recursivo.
- Recuperación de errores en análisis sintácticos predictivos.

Métodos de Análisis Ascendente:

- Ventajas e inconvenientes.
- Método por desplazamiento y reducción.
- Analizadores sintácticos LR.
- Recuperación de errores en analizadores sintácticos LR.

Unidad N°: 7

Título: Autómatas con Pila y Máquinas de Turing.

Contenidos:

- Autómatas con Pila: Definición de autómata con Pila 'Push-Down' (APD). Aceptación de una cadena por parte de un APD. APD determinísticos y APD no determinísticos. Propiedades.
- Máquinas de Turing: Definición de Máquina de Turing (MT). Aceptación de una cadena por parte de una MT. Propiedades.

Unidad N°: 8

Título: Lenguaje de Programación C.

Contenidos: Símbolos terminales. Tipos de datos. Operaciones. Expresiones. Notación Sintáctica del lenguaje C. Propositiones. Entrada y salida estándar. Estructura del programa. Bloques. Funciones. Pasaje de parámetros por valor y referencia. Clases de almacenamiento de las variables. Arreglos. Estructuras. Punteros. Semántica del lenguaje C. Implementación en C de reconocedores de lenguajes regulares (analizador léxico diminuto). Reconocimiento de errores léxicos, sintácticos y semánticos en un código C.

Carga horaria por tipo de formación práctica de toda la asignatura

Tipo de formación práctica	Horas reloj
Formación experimental	0
Análisis y resolución de problemas de ingeniería y estudios de casos	48
Formulación, análisis y desarrollo de proyectos.	0

Bibliografía Obligatoria:

Louden, K. (2004). Construcción de Compiladores – Principios y práctica. (1a. ed.). International Thomson Editores.

Jiménez Millán, J. A. (2014). Compiladores y procesadores de lenguajes.. Servicio de Publicaciones de la Universidad de Cádiz. <https://elibro.net/es/lc/utnfrro/titulos/33847>

Giró J. F., Vázquez J. C., Meloni B. E., Constable L. E. (2015). Lenguajes Formales y Teoría de Autómatas. Buenos Aires, Argentina. Alfaomega.

Mendoza García, M. G. (2015). Teoría de autómatas: un enfoque práctico.. Pearson Educación. <https://elibro.net/es/ereader/utnfrro/37993>

Kernighan, B. Ritchie, D. (1991). El Lenguaje de Programación C. (2a. ed.). Prentice-Hall Hispanoamericana.

Bibliografía optativa y otros materiales a utilizar en la asignatura:

Aho A., Sethi R. y Ullman J. (1990). Compiladores: Principios, Técnicas y Herramientas. Wilmington, Delaware, U.S.A. Addison Wesley.

Martínez López, F. (2015). Teoría, diseño e implementación de compiladores de Lenguajes.. RA-MA Editorial. <https://elibro.net/es/ereader/utnfrro/106460>

Ullman, J. D. E. Hopcroft, J. & Motwani, R. (2007). Introducción a la teoría de autómatas, lenguajes y computación (3a. ed.).. Pearson Educación. <https://elibro.net/es/ereader/utnfrro/52537>

Isasi P., Martínez P. y Borrajo D. (1997). Lenguajes, Gramáticas y Autómatas, un enfoque práctico. España. Addison-Wesley Iberoamericana.

Menchaca García, F. R. (2010). Fundamentos de programación en Lenguaje C.. Instituto Politécnico Nacional. <https://elibro.net/es/lc/utnfrro/titulos/74076>