



Universidad Nacional de Salta

FACULTAD DE CIENCIAS EXACTAS  
Av. Bolivia 5150 - 4400 - Salta  
Tel. (0387)425-5408 - Fax (0387)425-5449  
Republica Argentina



SALTA, 25 de Octubre de 2013

EXP-EXA N°: 8.156/2011

RESD-EXA N°: 572/2013

VISTO: las presentes actuaciones por las cuales se tramita la aprobación del Programa Analítico de la asignatura Teoría de la Computación II, para la carrera de Licenciatura en Análisis de Sistemas (Plan 2010), y

CONSIDERANDO:

Que el Departamento de Informática como así también la Comisión de Carrera de Licenciatura en Análisis de Sistemas, luego de analizar el Programa Analítico de la asignatura Teoría de la Computación II, aconsejan la aprobación del mismo.

Que la Comisión de Docencia e Investigación en su Despacho de fs. 16, aconseja aprobar el programa presentado.

POR ELLO y en uso de las atribuciones que le son propias;

EL DECANO DE LA FACULTAD DE CIENCIAS EXACTAS  
(Ad referéndum del Consejo Directivo)

RESUELVE


ARTÍCULO 1.- Aprobar, a partir del presente período lectivo, el Programa Analítico de la asignatura Teoría de la Computación II, para la carrera de Licenciatura en Análisis de Sistemas (Plan 2010), que como Anexo I forma parte de la presente Resolución.

ARTICULO 2°.- Hágase saber al Lic. Javier Trenti, Departamento de Matemática, Comisión de Carrera de Licenciatura en Análisis de Sistemas, Departamento Archivo y Digesto y siga a la Dirección de Alumnos para su toma de razón, registro y demás efectos. Cumplido ARCHÍVESE.-

RGG

  
Mag. MARIA TERESA MONTERO LAROCCA  
SECRETARIA ACADEMICA Y DE INVESTIGACION  
FACULTAD DE CS. EXACTAS - UNSa



  
Ing. CARLOS EUGENIO PUGA  
DECANO  
FACULTAD DE CS. EXACTAS - UNS.



Universidad Nacional de Salta

FACULTAD DE CIENCIAS EXACTAS  
Av. Bolivia 5150 - 4400 - Salta  
Tel. (0387)425-5408 - Fax (0387)425-5449  
Republica Argentina



ANEXO I RESD-EXA N°: 572/2013 - EXP-EXA N°: 8.156/2011

Asignatura: Teoría de la Computación II  
Carrera: Licenciatura en Análisis de Sistemas (Plan 2010).  
Fecha de presentación: 02/08/2013  
Departamento o Dependencia: Departamento de Informática.  
Profesor responsable: Lic. Javier Trenti  
Modalidad de Dictado: Cuatrimestral (segundo cuatrimestre)

Objetivos de la asignatura:

Los lenguajes de programación en general tienen características comunes que permiten obtener una mayor comprensión, a través de la formalización algorítmica, de problemas de aplicación. Esas características permiten hacer un estudio general de los lenguajes a través del diseño e implementación de los mismos. A su vez, dichas características permiten agrupar los diversos lenguajes en paradigmas a partir de los cuales se puede extraer un modo de enfocar la solución de distintos problemas.

Existen varias formas de encontrar una solución a un mismo problema, el desafío es concurrir al paradigma más apropiado dadas las características de aquel. Se requiere tener un conocimiento previo de los distintos paradigmas y la forma de encasillar los problemas en la solución más adecuada. Es necesario conocer las capacidades y las limitaciones de los distintos lenguajes de programación a fin de determinar cuál utilizar en un determinado dominio de aplicación.

El estudio del diseño de un lenguaje de programación permitirá apreciar cómo funcionan los diversos paradigmas. No se pretende que los alumnos lleguen a construir un lenguaje completo, pero sí que sean capaces de interactuar en la construcción de parte de los mismos, mediante la implementación de reconocedores de enlaces de almacenamiento de datos en memoria y módulos de un lenguaje de programación funcional. De este modo, la práctica ayudará a profundizar los conocimientos teóricos impartidos.

Las herramientas aprendidas en esta materia sirven, además, para la resolución de problemas generales, más allá del propósito especial por el que son transmitidas, ya que el esquema de trabajo elaborado requiere una formalización susceptible de ser utilizada en otras áreas.

El alumno deberá adquirir habilidades para:

1. Construir definiciones mediante lenguajes formales.
2. Argumentar sobre la utilización de un paradigma determinado, de acuerdo a las características del problema a resolver.
3. Comparar en forma crítica los resultados de la aplicación de distintos lenguajes de programación para la resolución de problemas.
4. Comprender y/o formular el diseño y la aplicación de un lenguaje de programación.
5. Aprender nuevos lenguajes de programación por su cuenta y ser capaz de evaluar sus capacidades y limitaciones.

Desarrollo del Programa Analítico

Unidad 1: Introducción.

Conceptos generales sobre Lenguajes de Programación. Definiciones. Dominios de programación. Criterios de evaluación de lenguajes. Historia: influencias en el diseño de lenguajes, evolución de los principales lenguajes. Clasificación, categorías de lenguajes.

Unidad 2: Sintaxis y Semántica Formal de Lenguajes.

Problema general de descripción de sintaxis. Métodos formales de descripción de sintaxis: BNF, grafos sintácticos, gramáticas libres de contexto. Análisis sintáctico (parsing). Gramática de atributos. Semántica estática y dinámica. Principios de Diseño de los Lenguajes de Programación.

///...



Universidad Nacional de Salta

FACULTAD DE CIENCIAS EXACTAS  
Av. Bolivia 5150 - 4400 - Salta  
Tel. (0387)425-5408 - Fax (0387)425-5449  
Republica Argentina



-2- ...///

## ANEXO I RESD-EXA N°: 572/2013 - EXP-EXA N°: 8.156/2011

Unidad 3: Administración de datos en memoria.

Entidades, atributos y ligaduras: tiempos de ligaduras, alcance y visibilidad. Variables, constantes y expresiones. Tipos de datos. Lenguajes Tipados. Sistemas de Tipos. Polimorfismo. Niveles. Encapsulamiento y Abstracción.

Unidad 4: Implementación de lenguajes.

Procesamiento de lenguajes de programación: intérpretes y compiladores. Representación de datos: objetos de datos, datos simples y estructurados, estructuras estáticas y dinámicas. Sentencias de asignación y expresiones. Estructuras de control a nivel de sentencias. Subrutinas y su implementación. Implementación top-down y bottom-up. Herramientas de implementación de analizadores léxico-sintácticos.

Unidad 5: Paradigmas de lenguajes.

Paradigma imperativo: principios y conceptos. Programación estructurada. Diseño modular. Diseño top-down. Paradigma funcional: fundamentos de la programación funcional. Introducción al Cálculo Lambda. Paradigma lógico: fundamentos y conceptos. Introducción al Cálculo de Predicados. Paradigma de orientación a objetos: fundamentos y conceptos. Características de diseño de lenguajes orientados a objetos. Evaluación de los Lenguajes: dominios de aplicación. Introducción a la teoría de las Bases de Datos.

Desarrollo del programa de Trabajos Prácticos:

Trabajo Práctico N° 1: Lenguajes, evaluación, clasificación.

Trabajo Práctico N° 2: Sintaxis y Semántica Formal de Lenguajes.

Trabajo Práctico N° 3: Administración de datos en memoria.

Trabajo Práctico N° 4: Implementación de lenguajes.

Trabajo Práctico N° 5: Cálculo Lambda. Cálculo de predicados.

Trabajo Práctico Trasversal de Programación: implementación del análisis léxico-sintáctico de un lenguaje.

Metodología y descripción de las actividades teóricas y prácticas:

El dictado de la asignatura se organiza en clases teóricas y prácticas. Las clases teóricas son magistrales en las cuales se desarrollan los contenidos teóricos de la asignatura, relacionando los distintos contenidos teóricos entre sí como así también con las actividades correspondientes a la práctica. Las clases prácticas están comprendidas por actividades grupales e individuales en las que se aplican los conceptos desarrollados en las clases teóricas. En las clases prácticas se desarrollan guías de trabajos prácticos con ejercicios de resolución en carpeta y ejercicios de programación. Se trabaja con las herramientas Flex y Bison. Estas herramientas permiten la implementación de analizadores léxicos y sintácticos.

Referencias bibliográficas:

[5E1306] Sebesta, R.W. Concepts of Programming Languages. Novena edición. Ed. Pearson. Boston. 2006.

[MIT04] Mitchel, J.C. Concepts in Programming Languages. Ed. Cambridge University Press. Cambridge. 2004.

[REY98] Reynolds, J.C. Theories of Programming Languages. Ed. Cambridge University Press. Cambridge. 1998.

///...



Universidad Nacional de Salta

FACULTAD DE CIENCIAS EXACTAS  
Av. Bolivia 5150 - 4400 - Salta  
Tel. (0387)425-5408 - Fax (0387)425-5449  
Republica Argentina



-2- ...///

ANEXO I RESD-EXA N°: 572/2013 - EXP-EXA N°: 8.156/2011

- [GHE97] Ghezzi, C., Jazayeri, M. Programming Language Concepts. Tercera Edición. Ed. Wiley. New York. 1997.  
[PRA01] Pratt, T.W., Zelkowitz, M.V. Programming Languages. Design and Implementation. Cuarta Edición. Ed. Prentice Hall. 2001.  
[AH008] Aho, A.V., Lam, M.S., Sethi, R., Ullman, J.D. Compiladores. Principios técnicas y herramientas. Segunda edición. Ed. Pearson. México. 2008.  
[MIT96] Mitchel, J.C. Foundations for Programming Languages. Ed. MIT Press. Boston. 1996.  
[MUC97] Muchnick, S.S. Advanced Compile ( Design and Implementation. Ed. Morgan Kaufmann Publishers - Academic Press. San Diego. 1997.

Régimen de regularización y promoción:

Condiciones de regularización:

Para regularizar la asignatura el alumno debe:

- Registrar un mínimo de 80% de asistencia a clases prácticas.
- Aprobar dos exámenes parciales o sus recuperaciones. Cada examen parcial tiene una única recuperación. Para aprobar cada parcial o su recuperación, el alumno debe obtener, al menos, sesenta puntos sobre cien.
- Presentar y aprobar al finalizar el cuatrimestre el Trabajo Práctico Transversal.


Condiciones de aprobación:

El alumno regular debe aprobar una evaluación escrita de 15 preguntas teórico-prácticas.


El alumno libre debe aprobar dos instancias de evaluación. Una primera instancia escrita en la que se evalúan contenidos prácticos y una segunda oral sobre los contenidos teóricos de la materia.

Para acceder a la primera instancia de evaluación el alumno debe poner en conocimiento a la cátedra de su intención de rendir en condición de libre para que se le asigne un trabajo de contenido similar al trabajo práctico transversal visto en clases. Este trabajo deberá ser presentado el día del examen final y será evaluado por el tribunal junto con el examen escrito del alumno. Para aprobar esta instancia el alumno deberá aprobar tanto el trabajo como el examen escrito.

rgg

  
Mag. MARIA TERESA MONTERO LARocca  
SECRETARIA ACADEMICA Y DE INVESTIGACION  
FACULTAD DE CS. EXACTAS - UNSa



  
Ing. CARLOS EUGENIO PUGA  
DECANO  
FACULTAD DE CS. EXACTAS - UNSa