



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

SALTA, 30 de Diciembre de 2011.

Expte. N°: 8615/08

RESCD-EXA N°: 946/2011

VISTO: las presentes actuaciones por las cuales se tramita la aprobación del Programa Analítico de la asignatura Metodología de la Programación, para la carrera de Licenciatura en Análisis de Sistemas (Plan 1997); y

CONSIDERANDO:

Que la Comisión de Carrera correspondiente, aconseja la aprobación del Programa de la asignatura antes mencionada, el cual cumple con los contenidos mínimos contemplados en el Plan de Estudio.

Que el Departamento de Informática, analizó el Programa Analítico de la asignatura Metodología de la Programación, aconsejando la aprobación del mismo.

Que el Consejo Directivo en su sesión del día 21/12/2011, constituido en Comisión elabora el siguiente despacho que aprueba por unanimidad: "Contando con el visto bueno de la Comisión de Carrera de Licenciatura en Análisis de Sistemas y del Departamento de Informática, aprobar el programa".

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

EL CONSEJO DIRECTIVO DE LA FACULTAD DE CIENCIAS EXACTAS

R E S U E L V E

ARTÍCULO 1.- Tener por aprobado, a partir del presente período lectivo, el Programa Analítico de la asignatura Metodología de la Programación, para la carrera de Licenciatura en Análisis de Sistemas (Plan 1997), que como Anexo I forma parte de la presente Resolución.

ARTICULO 2°.- Hágase saber al Mag. Ángel Rubén Barberis, Departamento de Informática, 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

  
M<sup>ra</sup> MARIA TERESA MONTERO LAROCCA  
SECRETARIA ACADEMICA  
FACULTAD DE CS. EXACTAS - UNSA



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



*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 – RESCD-EXA N°: 946/2011 -: Expte. N°: 8615/08

Asignatura: Metodología de la Programación  
Carrera: Licenciatura en Análisis de Sistemas (Plan: 1997)  
Fecha de Presentación: 27/10/2011  
Departamento Docente: Informática  
Profesor Responsable: Mag. Lic. Angel Rubén Barberis  
Modalidad de Dictado: Cuatrimestral (II)

### **Fundamentación**

La principal razón para aprender un lenguaje de programación es utilizar la computadora para resolver problemas. Pero, el desarrollo de una sociedad cada vez más informatizada que demanda con avidez mayor cantidad de productos software de cualquier tipo, está provocando la proliferación incontrolada de componentes software anárquicamente desarrollados; es decir, sin ajustarse a unos fundamentos metodológicos y a una disciplina de diseño que sirva de base para futuros desarrollos. Los programas de computadoras así desarrollados, son muy volátil, y por lo tanto, poco duradero, creando en el usuario informático una sensación de insatisfacción permanente al constatar que, ante un mínimo cambio o una adecuación de los requerimientos iniciales, demandaría un desarrollo completamente nuevo del software. Esta situación se aliviaría en gran medida, si el diseñador del producto software aplicara una línea metodológica de desarrollo, que le permita abordar el problema a resolver desde un enfoque integral, es decir, una metodología que abarque todas las fases del esquema productivo, así como los aspectos de administración (planificación y control, medida y seguimiento, documentación, pruebas y garantía de calidad) que en su mayoría no se tenían en cuenta en los desarrollos tradicionales. Todos ellos suponen mejoras importantes y necesarias en todos los dominios, que abarcan desde el análisis del problema hasta la puesta en marcha del producto final y posterior mantenimiento. Para lograr buenos profesionales del desarrollo de software es necesario que conozcan detalladamente al menos una metodología y la filosofía subyacente; marco de aplicación de la misma; ventajas e inconvenientes de su empleo, así como los principios sobre los que se debe asentar una buena metodología.

### **Objetivos Generales**

Que el alumno logre:

Comprender los fundamentos básicos de los lenguajes de programación, sus propiedades, mecanismos de representación y traducción.

Sentar las bases formales para abordar el diseño y la implementación de lenguajes de programación.

Adquirir la capacidad de construir programas de forma adecuada, apoyándose en buenos hábitos de diseño, codificación, pruebas, documentación, depuración y mantenimiento.

Comprender y aplicar las pautas básicas del diseño de programas (modularidad, encapsulamiento, no redundancia, independencia funcional, etc).

Aplicar una metodología de desarrollo en la Programación Orientada a Objetos.

Aplicar una metodología de desarrollo en la Programación Funcional.

Adquirir capacidad de abstracción y espíritu crítico.

### **Contenidos Conceptuales (Programa Analítico)**

///...



ANEXO I – RESCD-EXA N°: 946/2011 -: Expte. N°: 8615/08

1. Herramientas de Soporte para el Desarrollo de Software
  - 1.1 Introducción. Motivación. 1.2 Lenguajes de Programación.
    - 1.2.1 Definición. Características. Clasificación.
    - 1.2.2 Traductores de Lenguajes: Intérpretes y Compiladores. Características.
    - 1.2.3 Estructura de un Compilador: Fase de Análisis Léxico, Sintáctico, Semántico y Generación de Código.
  - 1.3 Diseño de Lenguajes de Programación.
    - 1.3.1 Aspectos Lingüísticos.
    - 1.3.2 Principios de Diseño e Implementación.
    - 1.3.3 Técnica de Especificación Semántica.
    - 1.3.4 Familia de Lenguajes de programación.
    - 1.3.5 Lenguajes de Dominio Específico.
    - 1.3.6 Máquina Abstracta.
2. Desarrollo de Programas
  - 2.1 Introducción.
    - 2.1.1 Problemas de Programación. Definición. Características.
    - 2.1.2 Metodología y Paradigma de la Programación.
  - 2.2 Etapas del desarrollo de programas.
  - 2.3 Proceso de diseño de programas. Esquemas de Modelización.
  - 2.4 Fundamentos del diseño de Programas: abstracción, refinamiento, modularidad, estructura de programa, estructura de datos, ocultamiento de la información.
  - 2.5 Diseños de Casos de Pruebas: Unitarias, Integración y Aceptación.
  - 2.6 Documentación: Importancia. Clasificación.
3. Diseño Modular de Programas
  - 3.1 Introducción. Importancia del Diseño Modular
  - 3.2 Diseño Modular: tipos de módulos, independencia funcional, cohesión, acoplamiento. 3.3 Técnicas de Diseño:
    - 3.3.1 Diseño de Datos. Principios de Bases de Datos. Manejo de Archivos.
    - 3.3.2 Diseño Estructural.
    - 3.3.3 Diseño de Programas orientado a la estructura de datos
    - 3.3.4 Diseño Procedimental.
  - 3.4 Estrategias del Diseño Estructurado.
4. Programación Orientados a Objetos (P00)
  - 4.1 Introducción: Programación Estructurada. Desventajas.
  - 4.2 Fundamentos de la Ingeniería de Software Orientados a Objetos.
  - 4.3 Metodología Orientada a Objetos.
  - 4.4 Análisis del Problema Orientado a Objetos. 4.5 Diseño de Soluciones Orientado a Objetos. 4.6 Conceptos básicos de la POO: Clases, objetos, herencias, mensajes, polimorfismo, encapsulación y ocultación de Datos; propiedades de las clases de objetos.
5. Programación Funcional no Convencional
  - 5.1 Introducción a la programación funcional.
    - 5.1.1 Programación funcional y programación imperativa.
    - 5.1.2 Introducción y semántica operacional.
  - 5.2 Tipos predefinidos.



ANEXO I – RESCD-EXA N°: 946/2011 -: Expte. N°: 8615/08

5.3 Patrones y Definiciones de Funciones.

5.4 Funciones de orden superior.

5.5 Polimorfismo.

5.6 Listas.

5.7 Aplicaciones de la programación funcional.

4. Programa de Trabajos Prácticos

Las clases prácticas se estipulan en clases de 3 horas reloj cada una.

TP Nro.	Temas	Cant. Clases asignadas
1	Herramientas para el Desarrollo de Software	3
2	Introducción al Modelado de Programas	4
3	Diseño Detallado de Programas	4
4	Programación Orientados a Objetos	3
5	Programación Funcional	2

## 5. Referencias Bibliográficas

### 5.1 Bibliografía Básica

- Aho, A.; Lam, M.; Sethi, R.; Ullman, J. Compiladores. Principios, técnicas y Herramientas. 2da Edición. Addison Wesleyj. 2008.
- Gálvez Rojas, S.; Morán Mata, M.; Compiladores. Traductores y Compiladores con Lex/Yacc, Slex/Cup y JavaCC. <http://www.lec.uma.esrgalvez/Compiladores.html>.
- Schildt, Herbert. Programación y Técnicas. Turbo Pascal Avanzado. McGraw Hill .1998.
- Galve, J. et al. Algorítmica: Diseño y Análisis de Algoritmos Funcionales e Imperativos. Editorial Ra-ma. 2000.
- Alfonso Amo. et al. Entornos y Metodologías de Programación. Editorial Paraninfo. 1995
- Amescua Seco, A. et al. Ingeniería del Software de Gestión. Análisis y Diseño de Aplicaciones. Editorial Paraninfo 1995.
- Alcalde, E. et al. Metodología de la Programación. Editorial McGraw-Hill. 2000.
- Carretero Pérez, J. et al. Fundamentos de Programación. Editorial Thomson. 2007.
- López Herran, J. et al. Fundamentos de Programación. Editorial Paraninfo.1998.
- Cairó, Osvaldo. Metodología de la Programación: Algoritmos, diagramación y programas. 3ra. Edición. McGraw-Hill. 2005.
- Peña Marí. Diseño de Programas: Formalismo y Abstracción. Editorial Pearson. 2007.
- Bertrand Meyer. Construcción de software orientado a objetos. Editorial Paraninfo. 2002.
- Jayanes Aguilar, Luis. Fundamentos de la Programación: Algoritmos, Estructuras de Datos y Objetos. Editorial McGraw-Hill. 2001.
- Joyanes Aguilar, Luis. Programación Orientada a Objetos: Conceptos, Modelado, Diseño y Codificación en C++. Editorial McGraw-Hill. 1999.
- Bird, Richard. Introducción a la Programación Funcional con Haskell. Editorial Prentice Hall. 2001.
- Ruiz Blas, C. et al. Razonando con Haskell. Editorial Thomson.2007.

### 5.2 Bibliografía Complementaria

///...



Universidad Nacional de Salta

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

-4- ...//

ANEXO I – RESCD-EXA N°: 946/2011 -: Expte. N°: 8615/08

- Pressman, R. Ingeniería de Software. Editorial McGraw-Hill. 2007.
- Sommerville, I. Ingeniería de Software. Editorial Pearson. 2005.
- Arto Salomaa. Formal Languajes. Ira. Edición 1973. Editorial Academic Press.
- Sanchis Llorca, Galán Pascual. Compiladores: Teoría y Construcción. Ira. Edición. 1986. Editorial Paraninfo.

## 6. Metodología y descripción de las Actividades teóricas y prácticas

### 6.1 Metodología de la Enseñanza

Para desarrollar las clases teóricas de los contenidos temáticos propuestos, se utiliza una metodología que le permite al alumno, elaborar nuevos conocimientos a partir de la base de enseñanzas anteriores. Se procura que el aprendizaje de los estudiantes sea activo y participativo, en lugar de permanecer de manera pasiva observando lo que se les explica. Esto garantiza que el aprendizaje del alumnado se lleve a cabo mediante dos procesos complementarios: la asimilación y el alojamiento. Se complementará esta metodología con un ambiente de interacción continua del proceso dinámico de enseñanza - aprendizaje. En las clases prácticas, se adoptará la enseñanza concebida como el proceso dinámico que le permite al alumno, incorporar nuevos conocimientos y nuevas pautas de conductas; a través de una serie de actos que realiza el docente con el propósito de crear condiciones que le den al estudiante la posibilidad de aprehender el conocimiento. Teniendo en cuenta las metodologías a utilizar, se plantea como estrategias de enseñanza: Clases Magistrales, Dinámicas de Grupos, Estudios Dirigidos y Trabajos en pequeños grupos.

## 7. Sistema de Evaluación y Promoción

### 7.1 Régimen de Regularización

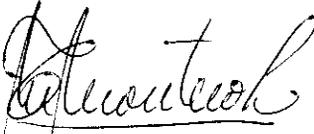
Para regularizar la asignatura, el alumno debe:

- Aprobar cada uno de los dos exámenes parciales o sus respectivas recuperaciones de carácter práctico, con una nota mayor o igual a 60/100.
- Presentar los Trabajos Prácticos con informe de resultados.
- Cumplimentar con el 85 % de asistencia a clases prácticas

### 7.2 Régimen de Aprobación

Para aprobar la asignatura el alumno debe rendir y aprobar un examen final de carácter teórico-práctico con una nota mayor o igual a 60/100.

rgg

  
Mag. MARIA TERESA MONTERO LAROCCA  
SECRETARIA ACADEMICA  
FACULTAD DE CS. EXACTAS - UNSA



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