

**UNIVERSIDAD NACIONAL DE SANTIAGO DEL ESTERO  
FACULTAD DE CIENCIAS EXACTAS Y TECNOLOGÍAS**

***LICENCIATURA EN SISTEMAS DE INFORMACIÓN***

# ***PROGRAMACIÓN LÓGICA Y FUNCIONAL***

**PLANIFICACIÓN  
(Plan 2011)**

**EQUIPO CÁTEDRA**

**Ing. Ruben Fernando Araujo- Profesor**

**Msc. Ing. Raquel Zarco – Jefe de Trabajos Prácticos- Exclusivo**

**AÑO 2012**

## 1. IDENTIFICACIÓN

**1.1. Nombre de la Asignatura:** Programación Lógica y Funcional.

**1.2. Nombre de la Carrera:** Licenciatura en Sistemas de Información (Plan de estudios 2011)

**1.3. Ubicación de la Asignatura**

**1.3.1. Módulo y Año en que está Ubicada la Asignatura**

La asignatura está ubicada en el segundo módulo, correspondiente al tercer año de la carrera.

**1.3.2- Ciclo al que pertenece la Asignatura/Obligación Curricular**

La asignatura pertenece al Primer Ciclo.

**1.3.3- Área a la que pertenece la Asignatura/Obligación Curricular**

ÁREAS	CARGA HORARIA EN HORAS RELOJ
Ciencias Básicas	
Teoría de la Computación	
Algoritmos y Lenguajes	75 hs.
Sistemas Operativos y Redes	
Ingeniería de Software, Bases de Datos y Sistemas de Información	
Aspectos Profesionales y Sociales	
Otra	
<b>CARGA HORARIA TOTAL DE LA ACTIVIDAD CURRICULAR</b>	<b>75 hs.</b>

**1.3.4- Carga horaria semanal:** 5 hs.

**1.3.5- Correlativas Anteriores:** Estructura de Datos y Programación

**1.3.6- Correlativas Posteriores:** Inteligencia Artificial

**1.4. Objetivos del Plan de Estudios para la Asignatura**

En el Plan de Estudios no se han definido objetivos para la asignatura.

**1.5. Contenidos Mínimos**

Paradigma Funcional. Clasificación. Principales características: transparencia referencial, evaluación diferida, recursividad, listas y funciones de orden superior. Estructuras de datos. Resolución de problemas y algoritmos fundamentales. Lenguajes funcionales: características fundamentales, tipos de datos y aplicaciones.

Paradigma lógico. Principales características: lógica proposicional, declaraciones, inversibilidad y backtracking. Estructuras de datos. Resolución de problemas y algoritmos fundamentales. Lenguajes lógicos: características fundamentales, tipos de datos y aplicaciones.

**1.6. Año Académico:** 2012

## 2. PRESENTACIÓN

### 2.1. Ubicación de la Asignatura como tramo de conocimientos de una disciplina / ubicación de la obligación curricular como actividad o herramienta

Esta asignatura corresponde al Área de Algoritmos y Lenguajes. Está orientada fundamentalmente al estudio, análisis y diseño de soluciones algorítmicas para resolver problemas computacionales a través de la programación lógica y funcional.

### 2.2. Conocimientos y habilidades previas que permiten encarar el aprendizaje de la asignatura / obligación curricular

- Resolución de problemas con computadoras, aplicando el Paradigma Orientado a Objeto, adquiridos en Estructura de Datos y Programación.
- Conocimientos básicos de idioma extranjero adquiridos en Inglés II.
- Conocimientos adquiridos en la asignatura Lógica II.

### 2.3. Aspectos del perfil profesional del egresado a los que contribuye la asignatura

➤ Posee:

- Los conocimientos básicos (lógico-matemáticos y computacionales) para una adecuada fundamentación teórica de su quehacer profesional específico.

➤ Está capacitado para:

- Realizar tareas de investigación, tanto a nivel básico como de aplicación en el ámbito que es específico de su competencia profesional.
- Analizar y seleccionar las estructuras de datos, necesarias para los diferentes Sistemas de Información.

## 3. OBJETIVOS

Que el alumno desarrolle las siguientes competencias:

### a) Competencias básicas

- Capacidad de investigación
- Capacidad para actuar en nuevas situaciones
- Capacidad para buscar, seleccionar y procesar la información necesaria para la resolución de problemas
- Capacidad para verificar las soluciones encontradas

### b) Competencias transversales

- Capacidad para la comunicación oral y escrita
- Capacidad para identificar, plantear y resolver problemas
- Capacidad para relacionar la teoría a la práctica

**c) Competencias Específicas**

- Capacidad para realizar la búsqueda creativa de soluciones algorítmicas y la selección criteriosa de la alternativa más adecuada aplicando los paradigmas de la programación Lógica y Funcional.
- Capacidad para verificar la solución de algoritmos desarrollados usando Programación Lógica y programación Funcional.
- Destreza para diseñar, ejecutar e interpretar programas usando lenguaje PROLOG Y HASKELL, usando distintas plataformas tecnológicas, aplicando los paradigmas Lógico y Funcional.

**4. SELECCIÓN Y ORGANIZACIÓN DE CONTENIDOS****4.1. Programa Sintético sobre la base de los contenidos mínimos****Unidad 1: *Paradigmas***

- Paradigmas. Clasificación.

**Unidad 2: *Paradigma Funcional***

- Paradigma funcional. Clasificación. Principales características: transparencia referencial, evaluación diferida, recursividad, listas y funciones de orden superior. Estructuras de datos. Resolución de problemas. Algoritmos fundamentales.

**Unidad 3: *Lenguaje Funcional***

- Lenguajes funcionales: características fundamentales, tipos de datos y aplicaciones.

**Unidad 4: *Paradigma Lógico***

- Paradigma lógico. Principales características: lógica proposicional, declaraciones, inversibilidad y backtracking. Estructuras de datos. Resolución de problemas y algoritmos fundamentales. Lenguajes lógicos: características fundamentales, tipos de datos y aplicaciones.

**Unidad 5: *Lenguajes lógicos***

- Lenguajes lógicos. Características fundamentales, tipos de datos y aplicaciones.

#### 4.2. Articulación Temática de la Asignatura / Obligación Curricular

La figura 1 presenta los principales conceptos a tratar en la asignatura y la relación entre los mismos.

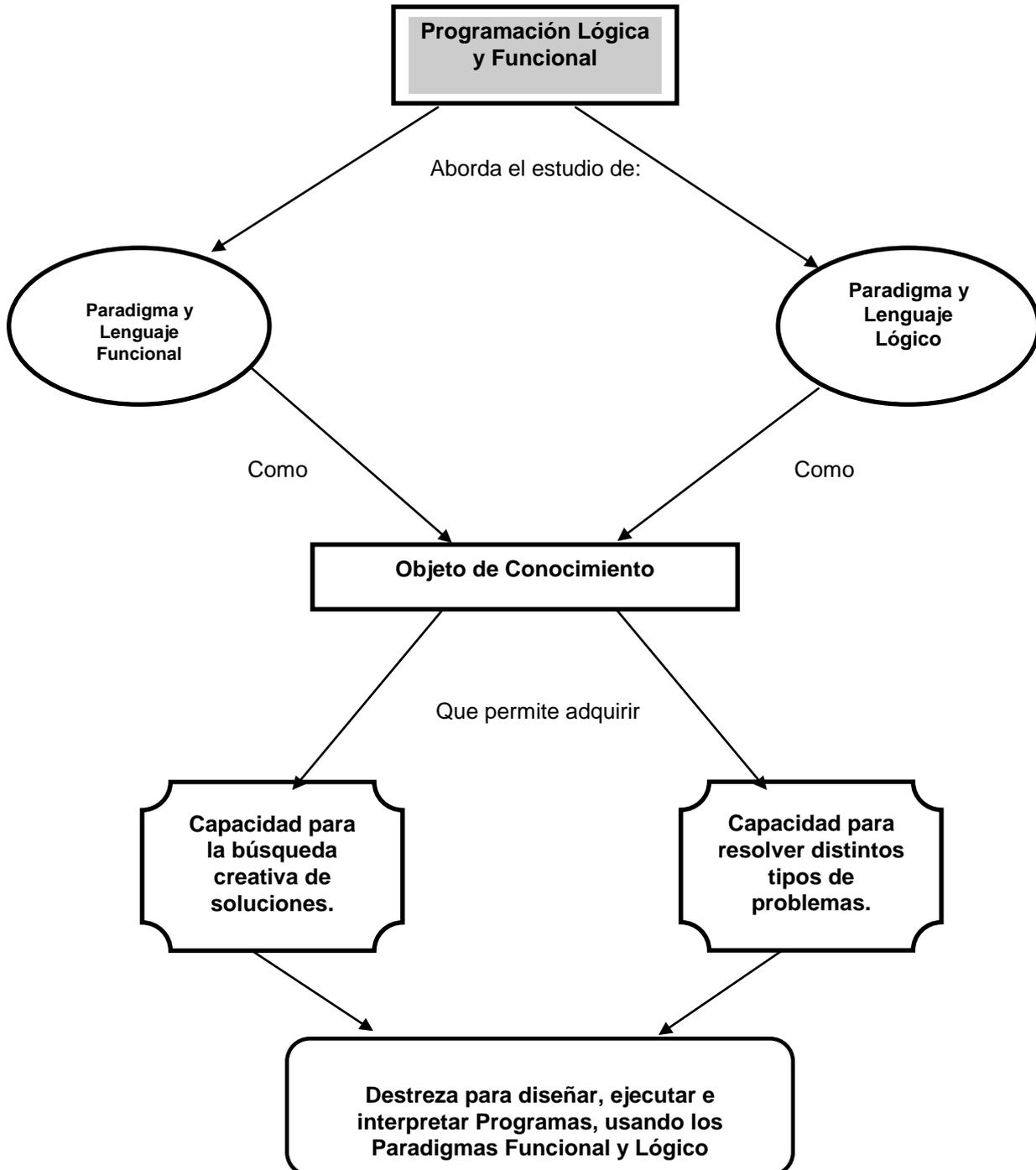


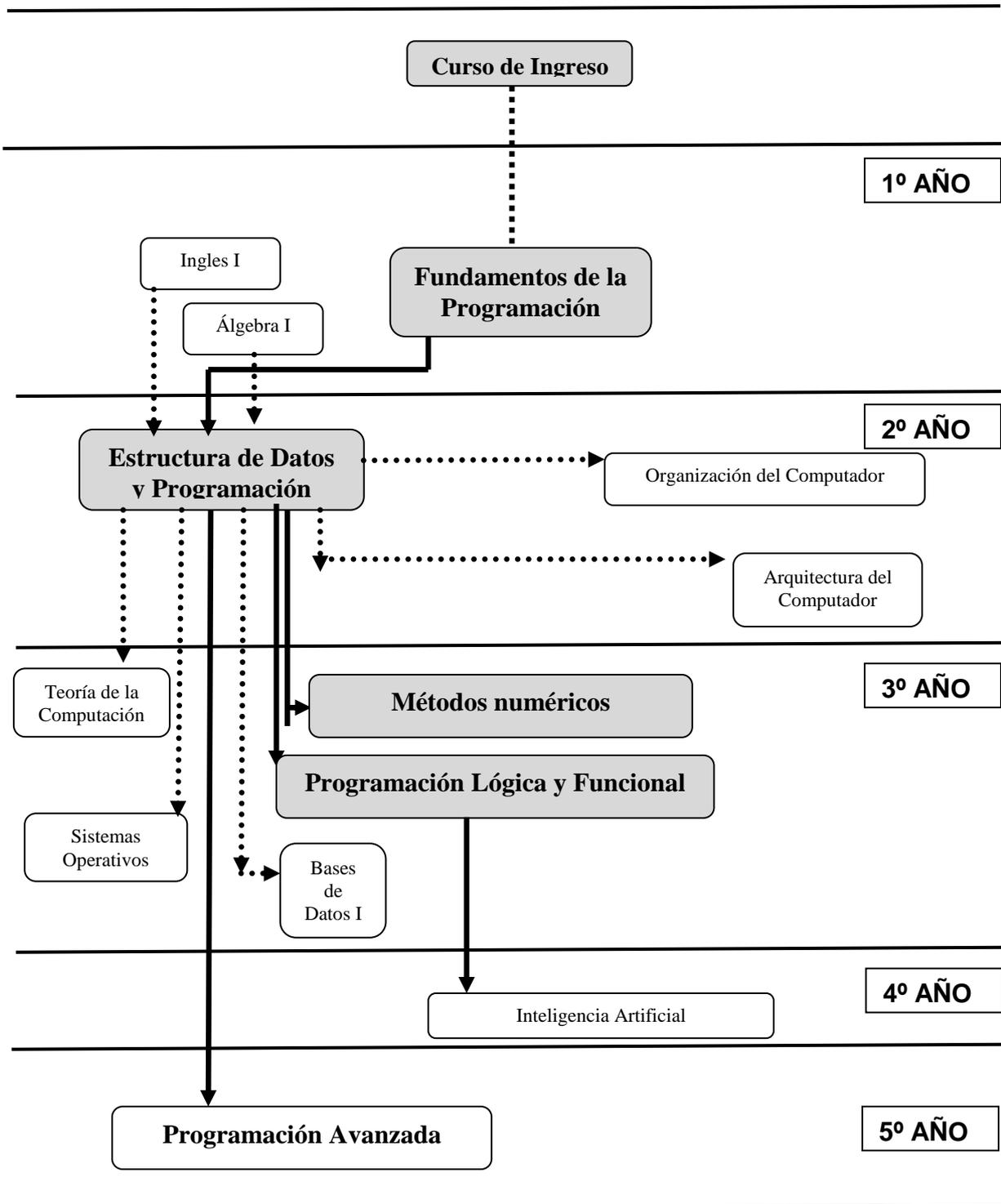
Figura 1

#### 4.3. Integración horizontal y vertical con otras asignaturas

##### Integración vertical

Esta asignatura corresponde al Área Algoritmos y Lenguajes, integrado además por las asignaturas Fundamentos de la Programación, Estructura de Datos y Programación; Programación Avanzada, y

Métodos Numéricos. En la figura se muestra la integración mencionada y la vinculación de las asignaturas del área.



Esta Área tiene como objetivos desarrollar en los estudiantes competencias específicas para:

- La búsqueda creativa de solución/es algorítmica/s a un problema, y la selección criteriosa de la alternativa más adecuada, aplicando distintos paradigmas de programación.
- El empleo de estructuras de control y estructuras de datos en la resolución de problemas, aplicando distintos paradigmas de programación

- La verificación de la solución de algoritmos desarrollados en los distintos paradigmas de programación.
- El uso distintos lenguajes de programación como herramienta computacional

En Fundamentos de la Programación, el alumno adquiere los conocimientos básicos de la programación, resuelve problemas y usa un lenguaje de programación como herramienta, aplicando el Paradigma de Programación Imperativo.

En Estructura de Datos y Programación, el alumno adquiere conocimientos de estructuras de datos avanzadas y del paradigma de programación orientada a objetos.

En Programación Lógica y Funcional y en Programación Avanzada se avanza en el aprendizaje de otros paradigmas de programación de manera de lograr que el alumno formalice y extienda sus conocimientos con respecto a la programación y en Métodos Numéricos adquiere los conocimientos sobre algoritmos numéricos y propagación de error.

#### **4.4. Programa Analítico**

##### **Unidad I: Paradigmas**

Paradigmas. Concepto. Necesidad de la existencia de distintos paradigmas de programación. Clasificación. Paradigmas declarativos: paradigma funcional y paradigma lógico.

##### **Unidad II: Paradigma Funcional**

Paradigma funcional: definición. Clasificación. Principales características. Concepto de función. La función como bloque de construcción de programas. Concepto de programa en el paradigma funcional. Variables. Tipos de datos. Definición de funciones. Recursividad. Funciones de orden superior. Transparencia referencial. Estructuras de datos. Listas. Manejo de listas. Evaluación diferida. Sistemas de Tipos. Polimorfismo y tipos genéricos. Tuplas. Expresiones Lambda. Resolución de problemas. Algoritmos fundamentales.

##### *Bibliografía recomendada*

Richard Bird. Introducción a la Programación Funcional con Haskell.  
Prentice hall. 2000

##### **Unidad III: Lenguaje Funcional**

Lenguajes funcionales. Lenguajes funcionales como lenguajes de programación. Historia y evolución de los lenguajes funcionales. Ventajas e inconvenientes de los lenguajes funcionales. Lenguajes de programación Haskell. Características fundamentales. Entorno de trabajo. Definición de programas. Uso del intérprete. Tipos de datos. Notación bidimensional. Módulos. Notación de listas. Notación de listas por comprensión. Operadores infijos y prefijos. Reglas de precedencia. Funciones para el manejo de listas, de tuplas y de funciones de orden superior. Formas de evaluación. Aplicaciones.

##### *Bibliografía recomendada*

Ruiz Jiménez, B.C.; Gallardo Ruiz, J.E.;Gutiérrez López, F. y Guerrero García, P.  
"Razonando con Haskell. Una Introducción a la Programación Funcional". 2000

## **Unidades IV: Paradigma Lógico**

Paradigma lógico: definición. Principales características. Lógica proposicional. Fundamentación. Predicados. Razonamiento y silogismos. Relaciones, hechos y reglas. Definición de programa en paradigma lógico. Diferencia entre una función y una relación. Declaraciones. Concepto de variable o incógnita. Unificación. Múltiples resultados. Backtracking. Inversibilidad. Estructuras de datos: listas. Pattern matching. Predicadores de orden superior. Functores. Polimorfismo. Resolución de problemas. Algoritmos fundamentales.

### *Bibliografía recomendada*

Clocksin, W.F. y Mellish, C.S.

“Programación en Prolog”. Ed. Gustavo Gili S.A.Barcelona 1987

## **Unidades V: Lenguaje Lógico**

Lenguajes lógicos. Lenguajes lógicos como lenguajes de programación. Historia y evolución de los lenguajes lógicos. Ventajas e inconvenientes de los lenguajes lógicos. Lenguajes de programación Prolog. Características fundamentales. Entorno de trabajo. Estructura de un programa en Prolog: dominio, tipos de datos predicados, meta u objetivo a buscar y clausulas. Tipos de datos. Aplicaciones.

### *Bibliografía recomendada*

Sterling, L. y Shapiro E.

“The Art of Prolog”. Ed. MIT Press. Cambridge, MA. 2ª ED 1994.

### **4.5. Programa y Cronograma de Trabajos Prácticos**

**No se prevee**

### **4.6. Programa y Cronograma de Actividades de Formación Experimental**

#### **Taller Lenguaje de Programación Haskell.**

##### Contenidos

Se desarrollaran los contenidos especificados en las Unidades II y III del programa analítico de la asignatura.

##### Objetivos

- ✓ Aplicar el paradigma de programación funcional.
- ✓ Capacidad para ejecutar y verificar correctamente programas en Lenguaje Haskell
- ✓ Adquirir destrezas fundamentales de programación que permitan la implementación de algoritmos usando el lenguaje de programación Haskell

##### Metodología

Las clases serán teóricas y prácticas, para lo cual se aplicaran métodos de resolución de ejercicios y problemas, con el fin de afianzar los conceptos teóricos, aclarar dudas y que el alumno alcance un buen desarrollo en su aprendizaje y se desarrollarán en el Laboratorio de Informática.

### Evaluación

Para aprobar el Taller de lenguaje de programación Haskell, el alumno debe aprobar un parcial o su correspondiente recuperatorio. La escala de valoración a emplear será cuantitativa politómica ( escala de 1 a 100). El puntaje mínimo para aprobar los mencionados parciales será de cincuenta (50) puntos y sesenta (60) puntos respectivamente.

Para tener derecho a rendir la evaluación parcial, el alumno deberá realizar dos practicas evaluativas entre las cuales deberá obtener una nota final (NF)  $\geq 50$  puntos, donde NF es:

$$NF=(0,35 \times \text{Puntaje de Evaluativo 1})+(0,65 \times \text{Puntaje de Evaluativo 2})$$

A su vez debe lograr un mínimo de 50 puntos en el Evaluativo 2.

### Instrumentos para la actividad

Para desarrollar esta actividad de formación experimental es necesario contar con los siguientes actores:

- Profesores
- Estudiantes
- Material didáctico
- Lenguaje de programación Haskell platform -2012.2.0.0 con GHC 7.4.1 Se ejecuta en Windows y Linux.
- Actividad de aprendizaje.

### Cronograma

Esta actividad experimental se desarrollará en un módulo con una dedicación de sesiones semanales de tres horas, totalizando en el año académico 28 hs.

## **2- Taller Lenguaje de Programación Prolog**

### Contenidos

Se desarrollaran los contenidos especificados en las Unidades IV y V del programa analítico de la asignatura.

### Objetivos

- ✓ Aplicar el paradigma de programación lógica.
- ✓ Capacidad para ejecutar y verificar correctamente programas en Lenguaje Prolog
- ✓ Adquirir destrezas fundamentales de programación que permitan la implementación de algoritmos usando el lenguaje de programación Prolog

### Metodología

Las clases serán teóricas y prácticas, para lo cual se aplicaran métodos de resolución de ejercicios y problemas, con el fin de afianzar los conceptos teóricos, aclarar dudas y que el alumno alcance un buen desarrollo en su aprendizaje y se desarrollarán en el Laboratorio de Informática.

### Evaluación

Para aprobar el Taller de lenguaje de programación Prolog, el alumno debe aprobar un parcial o su correspondiente recuperatorio. La escala de valoración a emplear será cuantitativa politómica ( escala de

1 a 100). El puntaje mínimo para aprobar los mencionados parciales será de cincuenta (50) puntos y sesenta (60) puntos respectivamente.

Para tener derecho a rendir la evaluación parcial, el alumno deberá realizar dos practicas evaluativas entre las cuales deberá obtener una nota final (NF)  $\geq 50$  puntos, donde NF es:

$$NF=(0,35 \times \text{Puntaje de Evaluativo 1})+(0,65 \times \text{Puntaje de Evaluativo 2})$$

A su vez debe lograr un mínimo de 50 puntos en el Evaluativo 2.

### Instrumentos para la actividad

Para desarrollar esta actividad de formación experimental es necesario contar con los siguientes actores:

- Profesores
- Estudiantes
- Material didáctico
- Lenguaje de programación SWI-Prolog y entorno Eclipse Ver 5.4.7. Esta práctica se realizara en Windows.
- Actividad de aprendizaje.

### Cronograma

Esta actividad experimental se desarrollará en un módulo con una dedicación de sesiones semanales de tres horas, totalizando en el año académico 25 hs.

## 5. BIBLIOGRAFÍA

### 5.1. Bibliografía Especifica

Título	Autor(es)	Editorial	Año y Lugar de edición	Disponible en	Cantidad de ejemplares disponibles
Lógica para la computación	Luis Ledesma	AlfaOmega	2009,Mexico	Biblioteca Dpto. Informática. F.C.E.y T.	1
Razonando con Haskell	<i>Ruiz Blas, Gallardo Jose, Guerrero Pablo, Gutierrez Francisco</i>	Editorial PARANINFO	2007, España	Biblioteca Dpto. Informática. F.C.E.y T.	1
Programación Lógica: Teoría y practica	<i>Pascual Julian Iranzo, Maria Alpuente</i>	Prentice Hall	2007, España	Biblioteca Dpto. Informática. F.C.E.y T.	1
Programming in Prolog Using the ISO Standard	<i>William F. Clocksin, Christopher S. Mellish</i>	Springer	2003,EEUU	Biblioteca Dpto. Informática. F.C.E.y T.	1
PROLOG, Programación y Aplicaciones	<i>W.D.Burnham y A.R.Hall</i>	Limusa	1990,Mexico	Biblioteca Dpto. Informática. F.C.E.y T.	1
Introducción a la Programación funcional con Haskell	<i>Richard Bird</i>	Prentice Hall	2000,España	Biblioteca Dpto. Informática. F.C.E.y T.	1

Tabla 1

\*sitios de interés en la Web

[www.haskell.org/haskellwiki/haskell](http://www.haskell.org/haskellwiki/haskell)

[www.es.utah.edu/hal/docs/daume=2yaht.pdf](http://www.es.utah.edu/hal/docs/daume=2yaht.pdf)

[www.haskell.org/haskellwiki/learning\\_haskell](http://www.haskell.org/haskellwiki/learning_haskell)

[www.swi-prolog.org](http://www.swi-prolog.org)

## 6. ESTRATEGIAS METODOLÓGICAS

### 6.1. Aspectos Pedagógicos y Didácticos

La metodología de enseñanza que se aplicará en el aula será en un espacio de diálogo y construcción, en el que se trabaje interactuando permanentemente, y de esta manera tanto los alumnos como el docente se consideran fuente de información.

En las clases teóricas se aplicará el método expositivo/lección magistral donde la exposición explicativa se realizará con el diseño de las clases en power point o con el uso del pizarrón, usando como recursos didácticos: esquemas, tablas, gráficos. Cada contenido desarrollado es mediado para su mejor comprensión y con el fin de propiciar el diálogo y discusión; además se realizan ejercicios de aplicación con el fin de aclarar conceptos, técnicas y métodos a utilizar.

Las exposiciones teóricas se complementan con el desarrollo un Taller de Lenguaje de Programación Haskell y el Taller del lenguaje de programación Prolog a los efectos de lograr una práctica de formación experimental que permita encontrar las soluciones a los problemas planteados en las clases prácticas, utilizando software libre en distintas plataformas, con el fin de lograr el desarrollo de habilidades prácticas, necesarias para el diseño de experimentos, y preparar a los alumnos para enfrentar los distintos cambios del mundo tecnológico.

### 6.2. Actividades de los Alumnos y de los Docentes

#### Actividades de los docentes

La asignatura esta a cargo de un equipo docente conformado por:

Un Profesor : Ing. Ruben Fernando Araujo

Una Jefe de Trabajos Prácticos: Msc. Ing. Raquel Zarco

El rol que desempeñen las docentes en el aula será de:

- ✓ Facilitador del aprendizaje.
- ✓ Propiciador de la comunicación.
- ✓ Facilitador del proceso de aprendizaje.
- ✓ Consultor frente a las dudas e inquietudes de los alumnos.

Las funciones específicas de cada docente serán:

- Preparar la planificación de la asignatura.
- Desarrollar clases teóricas y de formación experimental.
- Preparar y evaluar los practicos evaluativos.
- Preparar y evaluar los parciales y recuperatorios.
- Seleccionar material bibliográfico.
- Preparar material didáctico
- Supervisar el desarrollo de clases prácticas.
- Atender consultas de los alumnos

- Analizar las diferentes evaluaciones efectuadas con el fin de mejorar la calidad de la enseñanza.

#### Actividades de los alumnos

- ✓ Participar de las discusiones sobre los temas que se traten en cada clase.
- ✓ Resolver y presentar los trabajos experimentales que se les soliciten.

### 6.3. Mecanismos para la integración de docentes

Se realizarán actividades de revisión y coordinación en el área Algoritmos y Lenguajes en el marco de la Comisión de Seguimiento del Plan de Estudios de la carrera de Licenciatura en Sistemas de Información. Además, se efectuarán reuniones periódicas con las asignaturas de esta Área con el fin de aunar tareas conjuntas de integración.

### 6.4. Cuadro Sintético

TEORICAS	FORMACION PRACTICA					
	FORMACIÓN EXPERIMENTAL	RESOLUCIÓN DE PROBLEMAS DEL MUNDO REAL	ACTIVIDADES DE PROYECTO Y DISEÑO DE SISTEMAS DE INFORMACIÓN	INSTANCIAS SUPERVISADAS DE FORMACIÓN EN LA PRACTICA EXPERIMENTAL	OTRAS	TOTAL
22 HS. <sup>1</sup>	53 hs.	-----	-----	-----	----	75 hs.

Tabla 2

### 6.5. Recursos Didácticos

Se utilizarán como recursos didácticos:

- ✓ Lenguaje de programación Haskell y Prolog, equipo computacional del Laboratorio de Informática, consultas a Internet, para contribuir al desarrollo de la fase de formación experimental.
- ✓ Software PowerPoint, PC, cañón, tiza y pizarrón se usarán para clases teóricas.
- ✓ Bibliografía actualizada, que los alumnos utilizarán para adquirir habilidad de sintetizar e integrar informaciones e ideas.

## 7. EVALUACIÓN

### 7.1. Evaluación Diagnóstica

Teniendo en cuenta que la evaluación diagnóstica no sólo es una estimación, sino que tiene como propósito contribuir al aprendizaje, se llevará a cabo una única evaluación diagnóstica, al comienzo de las clases, cuya finalidad será determinar el nivel de conocimientos y habilidades previas que permitan encarar el aprendizaje de la asignatura.

Se evalúan contenidos adquiridos en las asignaturas Lógica II y Estructura de Datos y Programación.

La evaluación diagnóstica será especialmente diseñada, individual, escrita y objetiva. Se hará una prueba de opción múltiple para que el alumno marque la opción correcta. El nivel de calificación será cualitativa politómica (nivel medio, bajo alto).

### 7.2. Evaluación Formativa

La evaluación formativa es de carácter continuo y está dirigida a evaluar el proceso de enseñanza-aprendizaje durante todo el desarrollo de la asignatura. Para ello se tendrá en cuenta el desempeño que demuestren los alumnos en la realización de los trabajos prácticos, taller y evaluaciones parciales, con el fin de determinar el grado de aciertos, desaciertos y problemas en general, con el propósito de encarar, si fuera necesario, acciones correctivas.

### 7.3. Evaluación Parcial

#### 7.3.1. Programa y Cronograma de Evaluaciones Parciales

En la tabla 3 se muestra el programa de evaluaciones parciales a llevar a cabo durante el presente año académico.

Evaluación	Contenidos	Tipo	Fecha probable	Instrumento	Condiciones para rendir
Primer Parcial	Temas incluidos en unidades 2 y 3.	Especialmente diseñada, individual, escrita, prueba de desempeño	17/09 al 21/09	Resolución documentada de problemas	Aprobar los prácticos evaluativos
Recuperatorio del Primer Parcial	Temas incluidos en unidades 2 y 3.	Especialmente diseñada, individual, escrita, prueba de desempeño	24/09 al 28/09	Resolución documentada de problemas	Aprobar los prácticos evaluativos Haber desaprobado o estar ausente en el primer parcial
Segundo Parcial	Temas incluidos en unidades 4 y 5.	Especialmente diseñada, individual, escrita, prueba de desempeño	05/11 al 09/11	Resolución documentada de problemas	Aprobar los prácticos evaluativos Haber aprobado el primer parcial o su correspondiente recuperatorio
Recuperatorio del Segundo Parcial	Temas incluidos en unidades 4 y 5.	Especialmente diseñada, individual, escrita, prueba de desempeño	19/11 al 23/11	Resolución documentada de problemas	Aprobar los prácticos evaluativos Haber desaprobado o estar ausente en el segundo parcial

Tabla 3

#### 7.3.2. Criterios de Evaluación

A continuación se expresan en forma genérica los criterios de evaluación, los cuales serán refinados al momento de diseñar la prueba correspondiente.

- ✓ Capacidad para emplear un Lenguaje de Programación funcional Haskell como herramienta computacional.
- ✓ Capacidad para emplear un Lenguaje de Programación lógico, Prologl como herramienta computacional.

#### 7.3.3. Escala de Valoración

La escala de valoración a emplear en los parciales y recuperatorios será cuantitativa politómica (escala de 1 a 100)

#### **7.4. Evaluación Integradora**

No se prevee.

#### **7.5. Autoevaluación**

La autoevaluación se llevará a cabo por parte de los alumnos y del equipo cátedra. Para el primer caso se pretende lograr que el alumno desarrolle el juicio crítico, la capacidad de autoevaluarse, el valor de su trabajo y cultive un activo compromiso por la honestidad; para ello se diseñará una encuesta de opción múltiple, en la que el alumno se evaluará respecto de los conocimientos adquiridos en la asignatura, los procedimientos aprendidos, las actitudes profundizadas, etc. En el caso de los docentes permitirá una autoevaluación del equipo cátedra con el fin de corregir errores y fijar las pautas a seguir para lograr un mejor aprendizaje.

#### **7.6. Evaluación Sumativa**

##### **7.6.1. Condiciones para lograr la Promoción sin Examen Final de la Asignatura/ obligación Curricular (Rige la Resolución HCD N°135/ 00)**

No se prevee.

##### **7.6.2. Condiciones para lograr la Regularidad de la Asignatura**

Para regularizar la asignatura el alumno deberá:

- Aprobar los parciales previstos o sus correspondientes recuperatorios.
- Asistir al menos al 75 % de las clases teóricas y de formación experimental.

#### **7.7. Examen Final**

En el examen final los alumnos serán evaluados sobre los contenidos teóricos previstos en el programa de la asignatura y la realización de un trabajo de formación experimental. El examen podrá ser oral o escrito y en cualquiera de los casos, será individual. En el caso de ser escrita la escala de valoración a emplear será cuantitativa politómica (escala de 1 a 100). El puntaje mínimo para aprobar será de cincuenta (50) puntos.

#### **7.8. Examen Libre**

Para el examen Libre el alumno deberá aprobar las instancias que se detallan a continuación, siendo cada una de ellas eliminatorias:

Etapa 1: Evaluación escrita

- a) Aprobar una evaluación escrita de tipo teórica de los contenidos de las unidades II, III, IV y V del programa analítico de la asignatura.
- b) Resolución en computadora de un trabajo de formación experimental.

Etapa 2: Evaluación oral

Se utiliza la misma modalidad que rige para los alumnos regulares.