

## PROGRAMA DE ASIGNATURA

<b>ASIGNATURA:</b> Lenguajes Formales y Computabilidad	<b>AÑO:</b> 2012
<b>CARÁCTER:</b> Obligatoria	
<b>CARRERA:</b> Licenciatura en Ciencias de la Computación	
<b>RÉGIMEN:</b> cuatrimestral	<b>CARGA HORARIA:</b> 120 hs.
<b>UBICACIÓN en la CARRERA:</b> Cuarto año – Primer cuatrimestre	

### FUNDAMENTACIÓN Y OBJETIVOS

Lograr que el alumno maneje con madurez los siguientes conceptos:

- lenguajes libres de contexto
- maquinas de estado finito (autómatas a pilla y maquinas de Turing)
- funciones recursivas, funciones computables y funciones Turing computables, y su equivalencia
- computabilidad efectiva y Tesis de Church
- conjuntos recursivamente enumerables y conjuntos recursivos
- el halting problem

Estos conceptos le permitirán acceder a ideas y habilidades fundamentales para el desempeño en las ciencias de la computación teórica.

### CONTENIDO

#### Capítulo 1

Gramáticas libres de contexto. Lenguajes libres de contexto. Derivaciones leftmost. Automatas a pila. Equivalencia de lenguajes aceptados por vaciamiento de pila y por alcance de estado final. Equivalencia entre los lenguajes libres de contexto y los lenguajes aceptados por automatas a pila.

## Capítulo 2

Funciones  $\Sigma$ -mixtas. Identificación entre  $\Sigma^*$  y  $\omega$  para un orden total sobre  $\Sigma$ . Funciones  $\Sigma$ -recursivas y  $\Sigma$ -recursivas primitivas. Conjuntos  $\Sigma$ -recursivos y  $\Sigma$ -recursivos primitivos. Lema de división por casos. Iteración de funciones  $\Sigma$ -recursivas primitivas. Cuantificación acotada de predicados  $\Sigma$ -recursivos primitivos. Minimización acotada de predicados  $\Sigma$ -recursivos primitivos. Lema de independencia del alfabeto (sin demostración).

## Capítulo 3

El lenguaje imperativo  $S$  asociado a un alfabeto finito  $\Sigma$ . Sintaxis y semántica. Macros. Funciones  $\Sigma$ -computables. Equivalencia entre funciones  $\Sigma$ -computables y  $\Sigma$ -recursivas. Forma normal de Kleene. El halting problem. Caracterización de los conjuntos  $\Sigma$ -recursivamente enumerables.

## Capítulo 4

Maquinas de Turing. Lenguaje aceptado por una maquina de Turing.(por detención y por alcance de estado final). Equivalencia entre funciones  $\Sigma$ -Turing computables y  $\Sigma$ -recursivas y entre lenguajes  $\Sigma$ -recursivamente enumerables y lenguajes aceptados por maquinas de Turing.

## **BIBLIOGRAFÍA**

- BELL and MACHOVER, A Course in Mathematical Logic, North-Holland, 1986.
- M. DAVIS and E. WEYUKER, Computability, Complexity and Languages, Academic Pres 1983.
- J. HOPCROFT and J. ULLMAN, Introduction to Automata Theory, Languages and Computacion, Addison-Wesley 1979.

## **METODOLOGÍA DE TRABAJO**

Clases teóricas de aproximadamente dos horas, en las cuales se cubren los contenidos teóricos de la materia y clases prácticas de dos horas en las cuales se asiste al alumno en la resolución de los ejercicios con la finalidad de que se afirmen y esclarezcan los conceptos introducidos en la teoría.

## **EVALUACIÓN**

### **FORMAS DE EVALUACIÓN**

Se toman tres parciales para evaluar regularidad y/o promoción de la parte práctica de la materia. Los exámenes finales consisten de una parte práctica y una teórica,

en general tomadas por separado. La parte práctica se toma por medio de un escrito de cuatro horas aproximadamente y la parte teórica se toma ya sea por medio de un escrito de dos horas o por medio de un examen oral de duración aproximada de una hora.

### **CONDICIONES PARA OBTENER LA REGULARIDAD Y PROMOCIÓN**

Dos o más parciales aprobados con nota mayor o igual a 4 aseguran la regularidad y la parte práctica se promociona con los tres parciales aprobados con nota mayor o igual a 4 y promedio de los tres mayor o igual a 7.