

UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

**SEMINARIO DE LENGUAJES
(OPCION C)**

Año 2014

Carrera/ Plan:

Licenciatura en Informática

Plan 2003-07 / Plan 2012

Licenciatura en Sistemas

Plan 2003-07 / Plan 2012

Analista Programador Universitario

Plan 2007

Año: 2°

Régimen de Cursada: Semestral

Carácter: Obligatoria

Correlativas: Algoritmos, Datos y
Programas

Profesor: Ana Paola Amadeo, Christian
Rodríguez

Hs. semanales: 6 hs.

FUNDAMENTACIÓN

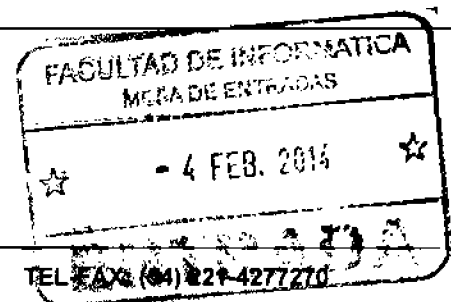
Dentro del marco de la carrera, esta opción introduce el lenguaje C, un lenguaje sencillo y a la vez poderoso, cuya historia se entrelaza con los sistemas operativos, redes y bases de datos. El aprendizaje del lenguaje C en esta instancia de la carrera contribuye favorablemente con la formación general del alumno, dado que lenguajes modernos heredan sus características, así como también es utilizado por otras materias avanzadas de la carrera como lenguaje de base.

OBJETIVOS GENERALES:

Profundizar los conocimientos obtenidos por el alumno en los primeros cursos vinculados con Algoritmos y Programación, permitiéndole desarrollar un estudio teórico-práctico de algún lenguaje de programación (el lenguaje puede variar con los cambios tecnológicos), poniendo énfasis en el análisis formal de las características del lenguaje y su comparación con los que el alumno conociera a ese momento (típicamente Pascal).

CONTENIDOS MINIMOS:

Estudio de un lenguaje de programación en el que se desarrollen aplicaciones concretas. En lo posible la oferta de lenguajes será variable y actualizada con el cambio tecnológico





PROGRAMA ANALÍTICO

- 1 Características Generales
 - 1.1 Un poco de historia
 - 1.2 El proceso de la compilación y la ejecución de un programa en C.
- 2 Sintaxis Básica
 - 2.1 Tipos predefinidos
 - 2.2 Declaración de variables
 - 2.3 Estructuras de Control
 - 2.4 El formato de un programa de C
- 3 Definición de funciones
 - 3.1 Pasaje de parámetros en C
 - 3.2 Definición de prototipos
- 4 Manejo de expresiones
 - 4.1 Operadores
 - 4.1.1 Precedencia entre operadores
 - 4.2 Conversiones de tipo
- 5 Arreglos y Punteros
 - 5.1 Arreglos en C
 - 5.1.1 Arreglos de caracteres
 - 5.2 Manejo de punteros
 - 5.2.1 Introducción a las variables de tipo puntero
 - 5.2.2 Declaración de punteros
 - 5.2.3 Aritmética de punteros
 - 5.2.4 Punteros y parámetros
 - 5.2.5 Arreglos de punteros
 - 5.3 Funciones para manejo de memoria
- 6 Estructuras
 - 6.1 Definición y manipulación de estructuras
 - 6.2 Estructuras recursivas
- 7 Entrada y salida estándar
 - 7.1 E/S (Entrada/Salida) de una secuencia (stream) de caracteres
 - 7.2 Funciones de E/S (Entrada/Salida) de caracteres
 - 7.3 Las funciones `getc()`, `putc()`, `getchar()`, `putchar()`, `getch()` y `putch()`
 - 7.4 La función `printf ()` y `scanf()`
- 8 Variables y constantes
 - 8.1 Tipos de variable
 - 8.2 Declaración de variables. Alcance y tiempo de vida
- 9 Directivas para el preprocesador
 - 9.1 Qué son las directivas para el preprocesador
 - 9.2 La directiva `#include`
 - 9.3 La directiva `#define`
 - 9.4 Otras directivas.
- 10 Manejo de archivos
 - 10.1 Tipos de archivo
 - 10.2 Archivos secuenciales
 - 10.3 Archivos de acceso directo
 - 10.4 Funciones para la manipulación de archivos
- 11 Librerías estándar
 - 11.1 La librería `string.h`
 - 11.2 La librería `math.h`
 - 11.3 La librería `stdlib.h`
- 12 Estándares de C
- 13 GlibC



METODOLOGÍA DE ENSEÑANZA

La metodología es del tipo taller, con clases teóricas donde se desarrollan los aspectos conceptuales del lenguaje, que se dictan utilizando presentaciones del tipo slideshow. Se incluyen ejemplos de código para resolver distintas situaciones problemáticas y animaciones que permitan ilustrar los conceptos más complejos. Durante el año 2014, se prevé dictar las clases teóricas en la sala de PC, con la asistencia de 2 ayudantes. La realización de esta actividad estará sujeta a la disponibilidad de la sala en los horarios de los estudiantes, de acuerdo a las bandas horarias. Estas clases no son obligatorias pero se tomará asistencia.

Las clases prácticas se realizan en la sala de PC de la facultad, los estudiantes plantean sus dudas y trabajan con los ayudantes, quienes los acompañan en este proceso. Y el JTP es quien realiza explicaciones de práctica al inicio de cada trabajo práctico, haciendo hincapié en los ejercicios más importantes y puntos a evaluar. En total se deben completar 5 prácticas. Los trabajos prácticos incluyen preguntas teóricas. Algunas de estas prácticas incluyen ejercicios para entregar, de carácter grupal. Cada entrega incluye una instancia de coloquio donde el docente a cargo del grupo realiza distintas preguntas sobre la temática abordada. La entrega se realiza a través de la plataforma virtual.

Esta metodología se complementa con la plataforma virtual Moodle. El uso de la herramienta desde hace aproximadamente 5 años generó un incremento significativo en el uso de los foros para intercambio entre los alumnos y docentes, en la resolución de problemas planteados en los trabajos prácticos. En este espacio también han compartido dirección Web de interés y otros recursos con esta finalidad.

A partir del año académico 2014 se planifica la realización de autoevaluaciones para reforzar temas teóricos y los coloquios de los trabajos prácticos utilizando la plataforma. Cada ayudante tendrá a cargo un grupo de alumnos, y será su función seguirlo en el desenvolvimiento de la cursada, intentado identificar los puntos más problemáticos para poder resolverlos en forma rápida y no provoque el abandono de la cursada.

A fin de mejorar la comunicación entre los estudiantes y la cátedra, se utilizarán las redes sociales Twitter y Facebook integradas a la plataforma virtual para informar los distintos eventos y materiales de la cátedra en forma ágil y automática.

Se utilizará también una herramienta para el seguimiento de la cátedra, de la asistencia de los alumnos a las prácticas y del aprovechamiento de cada práctica.



EVALUACIÓN

A partir de la práctica 3 se incluirán ejercicios de entrega obligatoria o evaluación online por medio de la plataforma virtual. Las entregas formarán parte del trabajo final para la aprobación de la cursada.

La **aprobación de la cursada** estará dada por la aprobación del 80% de los trabajos prácticos y el trabajo final integrador.

La **aprobación de la materia** estará dada por la aprobación de la cursada y un trabajo de investigación. La asistencia a las clases teóricas aportará a la calificación final.

La nota promedio de todos los trabajos será la nota final de la materia.

BIBLIOGRAFÍA OBLIGATORIA

- Kernighan, Ritchie (1991), *El lenguaje de Programación C. 2ª Edición*. Prentice Hall. ISBN 0-13-110362-8.
- Weiss. (1995) *Efficient C Programming. A Practical Approach*. Ed. Prentice Hall. ISBN 0-13-362658-X
- Perry (2000) *C con ejemplos*. Ed. Prentice Hall. ISBN: 9879460073
- Richard Reese. (2013) *Understanding and Using C Pointers*. Ed. O'Reilly. ISBN 978-1-44934-418-4

BIBLIOGRAFÍA COMPLEMENTARIA

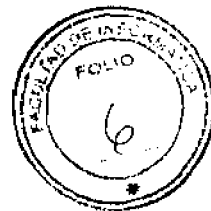
- Langsam, Yedidyah (1996) *Data structures using C and C++*. Ed. Prentice Hall.
- Van der Linden, Peter (1994) *Expert C programming: deep C secrets*. SunSoft Press. Ed. Prentice Hall.
- Antonakos, James L.; Mansfield, Kenneth (1997) *Programación Estructurada En C*. ISBN 8489660239.



CRONOGRAMA DE CLASES Y EVALUACIONES

Fecha de comienzo 11/03/2013

Clase	Contenidos/Actividades	Evaluaciones previstas
1	Introducción. Qué es C, que no es, que se puede hacer, que no. Repaso de operadores, ejemplos de programas en C. Características generales, tipos de datos, declaración de variables y constantes. Operadores. Conversiones de tipos: explícitas e implícitas. Estructura de un programa en C: fases en la construcción de un programa.	
2	Repaso de conceptos generales. Estructuras de control y Manejo de Entrada Salida.	Entrega de un ejercicio de la Práctica 1 de desarrollo y compilación sencilla. La entrega implica la aprobación.
3	El Preprocesador.	Autoevaluación de conceptos teóricos.
4	La memoria. Punteros y arreglos multidimensionales.	Entrega de un ejercicio de la Práctica 3 relacionado con el Trabajo Final. Coloquio en línea sobre temas teóricos y prácticos.
5	Arreglos de punteros y argumentos al main	Autoevaluación de conceptos teóricos.
6	Estructuras, typedef, Estructuras dinámicas (listas) TADs y tipos de variables	Entrega de un ejercicio de la Práctica 4 relacionado con el Trabajo Final. Coloquio en línea sobre temas teóricos y prácticos.
7	Arboles, Uniones, Manejo de Archivos.	Autoevaluación de conceptos teóricos.
8	Archivos – Continuación	Entrega de un ejercicio de la Práctica 5 relacionado con el Trabajo Final. Coloquio en línea sobre temas teóricos y prácticos.
9	Punteros a funciones	Autoevaluación de conceptos teóricos.
10	Estandares de C / Glibc	Completar el desarrollo del Trabajo Final con 2 entregas parciales cada 2 semanas y 2 posibilidades de re entrega.
11	Desarrollo del Trabajo Final	
12	Desarrollo del Trabajo Final	
13	Desarrollo del Trabajo Final	
14	Desarrollo del Trabajo Final	
15	Desarrollo del Trabajo Final	
16	Desarrollo del Trabajo Final	
17	Desarrollo del Trabajo Final	
18	Luego de la entrega final se realizará una puesta en común de todos los estudiantes de su trabajo	



**UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA**

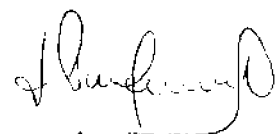
	final, haciendo hincapie en los problemas que tuvieron en el desarrollo del mismo y la forma de solucionarlos, así como también haciendo un balance de su desempeño.	
--	--	--


Contacto de la cátedra (mail, página, plataforma virtual de gestión de cursos):

seminarios@info.unlp.edu.ar

<http://catedras.info.unlp.edu.ar>

Firmas del/los profesores responsables:


Ana María Amador


Rodger Christian