

Programación II**Carrera/ Plan:**

Ingeniería en Computación Plan 2008/2011

Año 2025

Año: 1**Régimen de Cursada:** Semestral**Carácter (Obligatoria/Optativa):** Obligatorio**Correlativas:** Programación I**Profesor/es:****Hs. semanales:** 8**FUNDAMENTACIÓN**

Los conceptos y actividades abordadas en la asignatura permitirán al alumno familiarizarse con los aspectos vinculados a la tarea de la programación ya que se constituyen en uno de los conceptos básicos de la disciplina.

Se promueve trabajar a partir de diferentes situaciones problemáticas de la vida real y se abordan las estrategias de solución bajo criterios de calidad, eficiencia y corrección, para finalmente llegar a una implementación acorde a las especificaciones planteadas. Todos los aspectos mencionados son abordados a partir del trabajo con los diversos paradigmas de programación para entender sus diferencias y similitudes.

OBJETIVOS GENERALES

Profundizar el análisis de problemas resolubles con computadora, poniendo énfasis en la modelización, abstracción de funciones y en la modularización de los mismos.

Análisis e implementación de estructuras de datos no lineales. Tipos de datos y abstracción de datos.

Introducción a la Programación Orientada a Objetos. Comprender la arquitectura de una computadora

EJES TRANSVERSALES

03. Especificación, proyecto y Desarrollo de Software y Sistemas Conjuntos de Hardware y Software haciendo uso de conceptos, métodos y herramientas de gestión de proyectos, ingeniería de software, base de datos, experiencia del usuario, elicitación, análisis, especificación y validación de requerimientos. (1-Bajo)

04. Desarrollo de Redes de Computadoras y de Redes de Computadoras de área amplia, locales, inalámbricas y móviles. (1-Bajo)

06. Proyecto, desarrollo, dirección, construcción, operación y mantenimiento de Sistemas de Procesamiento de Señales, Sistemas Embebidos y sus periféricos incluido en software de soporte, Sistemas Computarizados de automatización y control y Sistemas Conjuntos de Hardware y Software. (1-Bajo)

09. Identificación, formulación y resolución de problemas de ingeniería en computación. (1-Bajo)

12. Utilización de técnicas y herramientas de aplicación en la ingeniería en computación. (1-Bajo)

14. Fundamentos para el desempeño en equipos de trabajo. (1-Bajo)

15. Fundamentos para una comunicación efectiva. (1-Bajo)

18. Fundamentos para el aprendizaje continuo. (1-Bajo)

CONTENIDOS MINIMOS (de acuerdo al Plan de Estudios)

- Tipos de datos compuestos (estáticos y dinámicos, lineales y no lineales).
- Recursividad. Introducción a los tipos abstractos de datos. Computadoras digitales.
- Organización funcional. CPU. Nociones de circuitos combinatorios y secuenciales.
- Memoria interna y externa. Periféricos. Representación de datos a nivel de máquina.
- Nociones básicas de manejo de interrupciones.

PROGRAMA ANALÍTICO

A- Recursividad

Características.

Ejecución de un programa y la pila de activación.

Análisis comparativo entre soluciones iterativas y recursivas.

Ejemplos.

B- Estructura de datos compuestas no lineales: árboles.

Introducción al concepto de datos no lineales.

Terminología y definiciones básicas del tipo de dato árbol.

Árboles binarios. Representación y operaciones.

Árboles binarios ordenados. Representación y operaciones.

Problemas que combinan árboles, listas y arreglos.

C- Memoria y periféricos

Tipos de memorias, clasificación.

Memoria principal, formas de organización. Memoria Cache.

Memoria secundaria, organización y formato de datos. Organización jerárquica de la memoria.

Tipos de periféricos, comparaciones.

D- Organización funcional de la computadora

Manejo de Instrucciones.

Modos de direccionamiento.

Lenguaje ensamblador.

Práctica sobre simulador de un procesador.

Subrutinas y pasaje de parámetros.

Conceptos de interrupciones.

E- Introducción a la programación orientada a objetos

Abstracción de datos y conceptos sobre tipos de datos.

Introducción a la Programación Orientada a Objetos

Reusabilidad de soluciones.

Abstracción de datos y procesos.

La noción de Objeto. Operaciones (métodos) aplicables a un objeto.

Concepto de clases e instancias.

Noción de herencia y polimorfismo. Relación con el re-uso.

Introducción al lenguaje JAVA

F- Introducción a la Concurrencia.

Impacto del cambio de los procesadores en el software.

Diferentes arquitecturas de procesadores.

Concurrencia.

Comunicación y Sincronización entre procesos.

Relación con la evolución de las arquitecturas de los procesadores.

Expresión de la concurrencia en los lenguajes de programación.

G - Conceptos finales

Metodologías de resolución de problemas.

Noción de paradigmas de programación.

Evolución de los lenguajes de programación. Alternativas.

BIBLIOGRAFIA

Título: Code Complete, 2nd Edition

Autores: Steve McConnell

Editorial: Cisco Press - ISBN: 9780735619678

Año de edición: Junio de 2024

Título: Introduction to Algorithms - 4ta edición

Autores: Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein

Editorial: MIT Press

Año de edición: 2022

Disponible desde: <https://acortar.link/4zN50p>

Título: Lógica de programación: Solucionario en pseudocódigo–Ejercicios resueltos.

Autores: Buriticá, O. I. T. (2021).

Editorial: Ediciones de la U.

Año: 2021

Título: Programación

Autores: Alfonso Jiménez Marín; Francisco Manuel Pérez Montes

Editorial: Ediciones Paraninfo, S.A. ISBN: 9788428342865

Año de edición: 2021

Título: PROGRAMACIÓN ORIENTADA A OBJETOS CON JAVA USANDO B (Spanish Edition) (Español)
1st Edición

Autores: David Barnes (Author), Michael Kolling (Author), S.L. Gestión editorial Avanzada (Translator)

Editorial: Pearson

Año de edición : 2017

Título: Organización y Arquitectura de Computadoras. 7ma edición

Autores : William Stallings

Editorial: Prentice Hall

Año de edición : 2016

Título: Problemas resueltos de estructura de computadores

Autores : Jesus Carretero, Perez Felix García Caballeira, Jose Daniel Garcia Sanchez, David Exposito Singh

Editorial: Paraninfo

Año de edición : 2015

Título: Aprende a programar con Java :un enfoque práctico partiendo de cero

Autores: Jiménez Marín, A., y Pérez Montes, F..

Editorial: Paraninfo

Año de edición: 2a ed., rev., 2016.

Título: Programación orientada a objetos con JAVA usando BlueJ

Autores : Barnes, David; Kölling, Michael

Editorial: Prentice Hall

Año de edición : 2013

Título: Programación estructurada y orientada a objetos

Autores : Lopez Roman Leobardo

Editorial: Alfa Omega Grupo editor

Año de edición : 2011

Título: Organización de Computadoras

Autores : Andrew Tanenbaum

Editorial: Prentice Hall

Año de edición : 2006

BIBLIOGRAFÍA COMPLEMENTARIA

Recursividad, Complejidad y Diseño de Algoritmos.
Jesús Bisbal Riera. Editorial UOC. España. 2009

Arquitectura de computadores- Un enfoque cuantitativo.
Hennessy & Patterson. Ed. Mc Graw Hill (1ra edición).2002

Estructuras de datos: referencia práctica con orientación a objetos.
Román Martínez, Elda Quiroga. Editorial Thomson, 2002.

Estructuras de Datos y Algoritmos.
Hernández R., Dormido R., Lazaro J. Ros S. Pearson Education. 2000.

Título: Algoritmos, datos y programas con aplicaciones en Pascal, Delphi y Visual Da Vinci
Autores : De Giusti, Armando et al.
Editorial: Prentice Hall
Año de edición : 2002

Introduction to algorithms.
Comen, Leiserson. MIT Press 2001.

Análisis y Diseño Orientado a Objetos con Aplicaciones.
Booch Grady. Addison Wesley. 1996.

METODOLOGÍA DE ENSEÑANZA

La asignatura Programación II se organiza en 4 módulos: M1-Programación Imperativa, M2-Programación Orientada a Objetos, M3-Programación en Assembler y M4-Programación Concurrente.

Se propone la modalidad de trabajo de aulas-talleres virtuales y desarrollo de actividades en línea. El trabajo en el aula-taller se fundamenta en un aprendizaje activo donde el alumno se apropia de los conocimientos, y el docente cumple el rol de coordinador y realiza actividades de tutoría.

Cada uno de los módulos trabaja con la modalidad de Aula-Taller donde cada estudiante utiliza su computadora desde la casa, se conecta a una sesión Webex con el ayudante para desarrollar ejercicios en forma grupal y poder consultar. Cada taller virtual tiene una duración estimada de 7 clases con una carga horaria de 8 horas semanales con 2 clases de 4 hs. cada una. Cada clase consta de contenidos teórico-prácticos con actividades en máquina para resolver en el aula taller virtual y también fuera del horario de clase utilizando el entorno virtual de enseñanza y aprendizaje (EVEA) y herramientas digitales específicas para cada caso.

Las clases se desarrollan de manera integral donde participan los diferentes integrantes de la cátedra. El docente responsable organiza la asignatura en general y define la dinámica del taller en conjunto con los docentes de la materia. Trabajan en forma coordinada con los auxiliares docentes para definir estrategias de tutorías virtuales de forma sincrónica usando Webex y/o otras herramientas de videoconferencia de uso en común..

La dinámica de las clases comprenderá una actividad inicial que estimule al estudiante y le aporte la problemática a tratar y el marco teórico involucrado. Un segundo momento de síntesis informativa que orienta y guía al alumno en la elaboración del conocimiento a través de ejemplos y casos. Y un tercer momento de actividades de integración y extensión con una guía de trabajo para ordenar las actividades individuales y grupales. Estas acciones serán tanto sincrónicas como asincrónicas.

En las clases se conformarán equipos de 10 alumnos integrados en grupos 2 o 3 alumnos para el desarrollo conjunto de actividades. Cada equipo estará supervisado por un ayudante.

El material teórico y la ejercitación práctica utilizados en el curso están disponibles en el EVEA (Entorno virtual de enseñanza y aprendizaje), se dispone de videos de explicación teórico-práctica de cada tema en cada módulo y de actividades y materiales de lectura.

Asistencia a los talleres virtuales:

- La asistencia a las clases teórico-prácticas es obligatoria.
- En cada clase los alumnos tendrán presente, ausente, o ausente justificado. Los ausentes justificados no pasan a ser presentes.
- La asistencia será el equivalente a estar conectado en Webex durante el desarrollo de las clases, se buscarán soluciones alternativas en casos de problema de conectividad.
- Pueden justificarse ausentes que serán evaluados por la cátedra.

EVALUACIÓN

Durante la clase, se propone la realización de ejercicios prácticos que los alumnos deben resolver y enviar, por el EVEA, al auxiliar a su cargo. La evaluación de estas actividades forma parte de la evaluación del proceso. El rendimiento satisfactorio de los alumnos en estas pruebas será considerado, a favor del alumno, durante la instancia de evaluación final de la asignatura.

Aprobación de un módulo

El alumno debe cumplir con el 80% de asistencia a las clases del módulo.

Haber revisado con los ayudantes en el aula, un ejercicio completo resuelto por computadora para el módulo a evaluar. Se tendrá en cuenta el trabajo en equipo y el diseño y análisis de los problemas

Asistir los días correspondientes a la resolución del trabajo final (TP final).

Resolver el TP final que se desarrolla durante el horario del Taller.

Aprobar el TP final.

La condición de aprobación del módulo puede ser Aprobado, Desaprobado o coloquio. La instancia de coloquio es para defender el TP final que no alcanza para ser aprobado. El resultado del coloquio puede ser aprobado o desaprobado.

Aprobación de la cursada de la asignatura

Se otorgará la cursada de Programación II a aquellos alumnos que hayan cumplido con el 80% de asistencia a cada uno de los módulos y se hayan presentado a rendir y aprobado los tres primeros módulos (Imperativo, Objetos y Assembler).

Si el alumno aprobó solo uno o dos módulos podrá presentarse a 2 (dos) instancias recuperatorias globales.

Si el alumno aprobó 0 debe recurrar la asignatura.

Aprobación del final con promoción

Aquellos alumnos que aprobaron los 3 (tres) módulos tendrán que entregar la resolución de un problema para el módulo de Programación Concurrente. Aprobada esta entrega, se promociona la asignatura.

Final de la asignatura

El final deberá ser rendido por aquellos alumnos que sólo aprobaron los 3 (tres) módulos mencionados en la aprobación de la cursada (Imperativo, Objetos y Assembler).

CRONOGRAMA DE CLASES Y EVALUACIONES

Cada semana del curso incluye dos días de clases teórico-prácticas que se organizan de acuerdo al siguiente cronograma:

Módulo 1 - Programación Imperativa (Pascal)

Clase	Fecha	Contenidos	Actividades
1	11/08	Presentación del Taller.	Anotarse en el EVEA Armado de grupos de trabajo
2	13/08	Recursión. Concepto. Características Ejercitación.	Resolución de ejercicios básicos utilizando recursión.
	18/08	FERIADO: Paso a la inmortalidad del Gral. José de San Martín	
3	20/08	Árboles Binarios Ordenados. Concepto. Operaciones. Ejercitación	Implementación de un árbol binario ordenado.
4	25/08	Árboles Binarios Ordenados. Borrado. Ejercitación	Implementación de las operaciones básicas de árboles binarios ordenados en Pascal.
5	27/08	Ejercicios combinados de árboles binarios ordenados y listas.	Árboles y listas ejercicios combinados
6	01/09	Ejercicios combinados de árboles binarios ordenados y listas. Segunda parte	Ejercicios combinados
7	03/09	Repaso	
8	08/09	Resolución del Trabajo Final 1	Evaluación
	10/09	Clase 1 + Clase 2 de Assembler	

Módulo 2 - Programación Orientada a Objetos (JAVA)

Clase	Fecha	Contenidos	Actividades
1	15/09	Muestra de resultados TP Final. Conceptos básicos del lenguaje Java	Implementación de programas simples imperativos en Java para ejercitar la sintaxis.
2	17/09	Introducción a la POO utilizando Java. Manejo de arreglos	Ejercitación que comprende la instanciación de objetos y envío de mensajes.
3	22/09	POO utilizando Java	Ejercitación que comprende programación de nuevas clases, instanciación de objetos de dichas clases, envío de mensajes a dichos objetos.
4	24/09	Constructores	Ejercitación que comprende la incorporación de constructores a las clases implementadas con anterioridad.
5	29/09	Concepto de herencia y polimorfismo	Ejercitación con herencia.
6	01/10	Ejercicios combinados de herencia polimorfismo y arreglos	Ejercitación
7	06/10	Repaso	

8	08/10	Resolución del Trabajo Final 2	Evaluación
	13/10	Clase 3 + Clase 4 + Repaso de Assembler	

Módulo 3 - Programación Concurrente

Clase	Fecha	Contenidos	Actividades
1	15/10	Muestra de resultados TP Final. Repaso R-Info: Pasaje de Parámetros. Introducción a la concurrencia.	Pasaje de Parámetros en R-Info. Relaciones entre conceptos de programación concurrente.
2	20/10	Conceptos básicos de Concurrencia. Entorno CMRE + Áreas	Ejercicios que muestran los problemas de concurrencia + Trabajo con el entorno CMRE aplicando los conceptos vistos a ejercicios con áreas.
3	22/10	Memoria distribuida	Trabajo con el entorno CMRE aplicando los conceptos vistos a ejercicios con memoria distribuida.
4	27/10	Memoria compartida	Trabajo con el entorno CMRE aplicando los conceptos vistos a ejercicios con memoria compartida.
5	29/10	Memoria distribuida y compartida	Trabajo con el entorno CMRE aplicando los conceptos vistos a ejercicios con memoria distribuida y compartida.
6	03/11	Repaso	
7	05/11	Resolución del Trabajo Final 3	
	10/11	NO HAY ACTIVIDADES. Corrección TP Final.	
Consulta	12/11	Exámen de Assembler.	
Consulta	17/11	Consultas para recuperatorio	
	19/11	FERIADO: Aniversario de la Fundación de La Plata	
	24/11	FERIADO: Día de la Soberanía Nacional	
	26/11	FERIADO: Día del Trabajador Nodocente	
Recuperatorio Global	01/12	Recuperatorio	
	03/12	NO HAY ACTIVIDADES. Corrección Recuperatorio.	
	08/12	FERIADO: Inmaculada concepción de María	
Muestra final	10/12		

Fechas de Evaluaciones

Tema	Fecha
Imperativo	08/09
Objetos	08/10
Concurrencia	05/11
Assembler	12/11

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

Blog de catedra: <http://blogs.unlp.edu.ar/programacion2/>

Contacto de la cátedra:

Profesor: Alejandro Héctor Gonzalez

Mail: agonzalez@lidi.info.unlp.edu.ar

Sitio Web: <https://ic.info.unlp.edu.ar/alcances-del-titulo-y-competencias-profesionales-de-ingenieria-en-computacion/>

Plataforma virtual: EVEA (Moodle) de la Facultad de Informática Curso 2024-Redictado de Programación 2

Otros: Blog de catedra: <http://blogs.unlp.edu.ar/programacion2/>

Firma del/los profesor/es



Prof Asociado Alejandro Héctor Gonzalez