



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Programación II

Carrera: Ingeniería en Computación
Profesor Responsable: González, Alejandro Héctor
Año: 1º
Duración: Semestral
Carga Horaria Semanal: 8hs
Carga Horaria Total: 112hs

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. Estructuras de datos no lineales. Tipos de datos y abstracción de datos. Introducción a la Programación Orientada a Objetos. Conceptos básicos de arquitectura de procesadores.

Contenidos Mínimos

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

A- Recursividad

- ✓ Características.
- ✓ Ejecución de un programa y la pila de activación.
- ✓ Análisis comparativo entre soluciones iterativas y recursivas.
- ✓ Ejemplos.

B- Conceptos de Eficiencia. Análisis de algoritmos.

- ✓ Concepto de eficiencia.
- ✓ Análisis de eficiencia de un algoritmo.
- ✓ Análisis de algoritmos según su tiempo de ejecución y su utilización de memoria.
- ✓ Análisis de eficiencia en algoritmos recursivos.
- ✓ Análisis de eficiencia en algoritmos de búsqueda y ordenación sobre vectores.

C- Estructura de datos compuestas enlazadas: listas.

- ✓ Repaso de listas simples.
- ✓ Listas circulares, Listas dobles, Listas múltiples. Características y operaciones.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

- ✓ Relaciones entre los accesos a listas, vectores, pilas y colas.

D- 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 combinen árboles, listas y arreglos.

E- Memoria

- ✓ Tipos de memorias, clasificación.
- ✓ Parámetros característicos, tamaño, tiempo de acceso, costo, otros.
- ✓ Memoria principal, formas de organización. Memoria Cache.
- ✓ Memoria secundaria, organización y formato de datos. Organización jerárquica de la memoria. Dispositivos de almacenamiento externo, disco, cinta, disco óptico, otros. Múltiples unidades de discos (RAID).

F- Periféricos

- ✓ Dispositivos de interacción típicos: terminales, pantallas, teclado, mouse, impresora, scanner, tabla digitalizadora, audio. Características de interconexión a cada uno de ellos.
- ✓ Clasificación
- ✓ Comunicación , protocolo RS232

G - Organización funcional de la computadora

- ✓ Manejo de Instrucciones.
- ✓ Modos de direccionamiento.
- ✓ Lenguaje ensamblador.
- ✓ Práctica sobre simulador de procesador.

H- Tipos de datos abstractos

- ✓ Abstracción de datos.
- ✓ Conceptos sobre tipos de datos.
- ✓ Módulos, interfaz e implementación. Encapsulamiento de datos.
- ✓ Pilas y colas como tipos de datos abstractos
- ✓ Introducción a la Programación Orientada a Objetos
- ✓ Motivación. 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. Relación con el re-uso.
- ✓ Aplicaciones.
- ✓ Características de los lenguajes enfocados a POO.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

I- Introducción a la Concurrencia.

- ✓ Esquema de procesadores de múltiples núcleos actuales.
- ✓ Relación con la clasificación de Flynn.
- ✓ Impacto del cambio de los procesadores en el software.
- ✓ Concurrencia.
- ✓ Comunicación y Sincronización entre procesos.
- ✓ Relación con la evolución de las arquitecturas de procesadores.
- ✓ Expresión de la concurrencia en los lenguajes de programación.
- ✓ Ejemplos.

J - Conceptos finales

- ✓ Metodologías de resolución de problemas.
- ✓ Diferentes arquitecturas de procesadores.
- ✓ Noción de paradigmas de programación.
- ✓ Evolución de los lenguajes de programación. Alternativas.

BIBLIOGRAFIA

Título: Diseño de algoritmos con implementaciones en Pascal y C

Autores: Nivo Ziviani, Joaquín Adiego

Editorial: Paraninfo

Año de edición: 2007

Título: Algoritmos, datos y programas con aplicaciones en Pascal, Delphi y Visual Da Vinci

Autores: De Giusti, Armando

Editorial: Prentice Hall

Año de edición: 2002

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: AlfaOmega Grupo editor

Año de edición: 2011

Título: Java. Como programar

Autores: Paul Deitel, Harvey Deitel

Editorial: Perason



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Año de edición: 2012

Título: Organización y Arquitectura de Computadoras. 7ma edición

Autores: William Stallings

Editorial: Pretince Hall

Año de edición: 2016

Título: Organización de Computadoras

Autores: Andrew Tanenbaum

Editorial: Pretince Hall

Año de edición: 2006

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

Descripción de las Actividades Teóricas y Prácticas

Programación 2 es una asignatura de primer año y se dicta en el segundo semestre de cada año. Corresponde a la Carrera de Ingeniería en Computación que se dicta en forma conjunta entre la facultad de Informática de la UNLP y la Facultad de Ingeniería de la UNLP.

La asignatura tiene como correlativa a Programación I y atiende en el semestre unos 70 alumnos que cursan en un turno que requiere una coordinación entre el Profesor Asociado, el Jefe de Trabajos Prácticos y los Auxiliares docentes.

El turno integra los estudiantes que realizan la asignatura por primera vez y los recusantes.

Por otra parte, es responsabilidad de los Profesores revisar los contenidos de las evaluaciones parciales de los alumnos y realizar permanentes visitas a las aulas de práctica.

En particular esta asignatura se vuelve a dictar en el primer semestre de cada año y cuanta con aproximadamente 40 alumnos. El objetivo es permitir a los estudiantes que recursan o aquellos que han realizado el redictado de la materia correlativa Programación I, puedan cursar en un semestre previo, y no esperar para realizar la materia.

Se dispone de un reglamento interno de la cátedra que está a disposición de los estudiantes y se utiliza para acordar la forma de trabajo durante el semestre.

Esquemáticamente la primera parte de la asignatura se concentra en la expresión de algoritmos y conceptos de recursividad realizando análisis de eficiencia, en la definición



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

de tipos abstractos de datos y en la evolución hacia estructuras de datos dinámicas no lineales.

Se continúa con la presentación de los conceptos básicos de Tipos Abstractos de Datos y Programación Orientada a Objetos. Se utiliza un lenguaje que permita trabajar con la definición inicial de clases, objetos, herencia y polimorfismo.

Se finaliza la materia con una introducción al funcionamiento interno de los programas en la computadora y su representación a bajo nivel y un análisis de la evolución de los procesadores que permite abordar el concepto de Programación Concurrente.

Los alumnos deben cumplir con el 66% de asistencia a las clases prácticas para poder acceder a las evaluaciones parciales de la asignatura. Todos los alumnos inicialmente pueden acceder a la promoción de la asignatura.

En forma particular algunos recursantes tienen inconvenientes para poder cursar la asignatura dada la superposición con otras asignaturas que se genera por el plan curricular de la Carrera y los diversos recorridos de los alumnos. En estos casos se justifica el inconveniente y se ofrece al alumno la posibilidad de justificar los presentes entregando resueltos todos los ejercicios de las prácticas y la entrega de un programa ejecutable de un ejercicio en particular de cada práctica. Estos alumnos tienen asignado un tutor a quien puede realizar las consultas necesarias para realizar las actividades propuestas.

Además, los estudiantes disponen de material adicional de lectura y ejercicios resueltos que pueden ser utilizados para una mejor comprensión de los temas tratados en el curso.

Hay un gran esfuerzo por mantener una comunicación fluida con los alumnos. Para ello los alumnos disponen en el EVEA WebUNLP del servicio de Mensajería a través del cual pueden comunicarse en forma directa con sus docentes: Ayudantes, JTP y Profesor. Asimismo, pueden realizar consultas previas a las instancias de evaluación parcial y/o final y disponen también de días y horarios fijos de consulta para los temas teóricos.

Por otra parte, la cátedra trata de realizar anualmente una encuesta a través de la cual los alumnos pueden expresar su opinión acerca de diversos temas vinculados con el desarrollo del curso, que van desde la articulación con el Ingreso hasta las modalidades de evaluación de teoría y práctica.

Para el desarrollo de las actividades que desarrollan la codificación e implementación de programas, los estudiantes deben responder dos cuestionarios. El primero es inicial y está vinculado al nivel de apropiación de tecnología digital. El segundo cuestionario es referido a la actividad entregada indagando en los detalles de la misma y la participación e integración del equipo.

Metodología de Enseñanza y Evaluación

METODOLOGÍA DE ENSEÑANZA



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

La metodología empleada en las clases teóricas consiste en la presentación de los temas, planteo de situaciones problemáticas específicas para abordar cada tema y posteriormente el análisis y discusión de las soluciones planteadas conjuntamente con los alumnos. Se proponen lecturas complementarias de diferentes textos citados en la bibliografía del curso. También se trabaja con lecturas recomendadas disponibles en el EVEA (Entorno Virtual de Enseñanza y Aprendizaje) WebUNLP (<https://webunlp.ead.unlp.edu.ar/>) y en el blog de la cátedra (<http://blogs.unlp.edu.ar/programacion2/>) que se utilizan en el curso.

A partir del cronograma definido inicialmente, la cátedra avanza coordinadamente contando con diferentes puntos de articulación entre teoría y práctica. Una de las estrategias de articulación son las denominadas “explicaciones de teoría-práctica” dictadas semanalmente por los Jefes de Trabajos Prácticos, que vincula las clases teóricas con las clases prácticas. De este modo se articulan los contenidos dictados en la teoría con ejemplos “generales” que permiten realizar las soluciones de numerosos casos particulares en las clases prácticas.

Otra estrategia de articulación es el desarrollado de actividades en la teoría. Los estudiantes deben desarrollar actividades que relacionan temas teóricos y prácticos en forma colaborativa tanto en las clases presenciales como en línea utilizando el entorno Webunlp y el blog de la cátedra.

METODOLOGÍA DE EVALUACIÓN

La cuidadosa organización de la cátedra y la coordinación entre sus docentes es un aspecto central para lograr el dictado de manera de adaptar los contenidos y la metodología teniendo en cuenta las características particulares de cada cohorte de alumnos.

Se desarrolla una evaluación continua que apunta a revisar el proceso de aprendizaje de los alumnos.

Se propone un sistema de promoción integrado (teórico-práctico) de todas las tareas que los alumnos desarrollan en la asignatura supervisado por el Profesor.

Se realizan varias reuniones de cátedra y se define (antes del inicio de cada ciclo lectivo) un cronograma de los temas teóricos y prácticos, la metodología adecuada, los contenidos, la bibliografía a utilizar y las condiciones administrativas del curso.

La evaluación finaliza con la integración de una nota por cada alumno. Los estudiantes pueden aprobar la materia por promoción o solo la parte práctica debiendo a posterior rendir el correspondiente final.

Se toman dos exámenes teórico-prácticos en las fechas estipuladas en el cronograma (durante el curso). Cada examen tiene su correspondiente recuperatorio. Además, cuentan con un examen flotante que se puede utilizar para aprobar uno de los dos exámenes.

Para lograr la aprobación del examen será necesario aprobar la parte práctica. Y para acceder a la promoción será necesario aprobar la parte práctica y los ejercicios teóricos con nota 6 o superior.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Se desarrollan además dos entregas de actividades grupales obligatorias con nota numérica.

La nota final de la asignatura se compone de las notas de cada examen aprobado y las dos notas de las entregas de las actividades grupales.

Los alumnos que obtienen una nota promedio (de las 4 notas) de 6 o mayor a 6 tendrán aprobada la promoción. En caso de no alcanzar la nota de 6 pueden recuperar las actividades entregadas para alcanzar la nota mínima de promoción.

Los alumnos que obtienen una nota promedio (de las 4 notas) de 4 o mayor a 4 y menor que 6 tendrán aprobada la cursada y deberán rendir examen final.

Durante toda la cursada los estudiantes tienen disponible la consulta en el aula y vía el EVEA WebUNLP. Los docentes responden dentro de las 48 hs de efectuada la consulta.

Los estudiantes que deban rendir a posterior el examen final tienen clases de consulta para el final y se pueden comunicar con los profesores vía los post del blog de la cátedra y los e-mails personales. Los docentes también publican las novedades de la cátedra en el blog para mantener informados a los estudiantes cuando ya terminaron de cursar y deben realizar algún proceso administrativo.