



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Programación I

Carrera: Ingeniería en Computación
Profesor Responsable: Sanz Cecilia
Año: 1º
Duración: Semestral
Carga Horaria Semanal: 8hs
Carga Horaria Total: 112hs

Objetivos Generales

Analizar problemas resolubles con computadora, poniendo énfasis en la modelización, abstracción de funciones y en la modularización de los mismos. Obtener una expresión sintética y precisa de los problemas, con una documentación de una metodología de trabajo por el alumno. Estudio, expresión simbólica, implementación y evaluación de algoritmos, orientando los mismos a la resolución de las partes (módulos) en que se descomponen los problemas. Introducción de las nociones de estructuras de datos, tipos de datos y abstracción de datos. Combinar los elementos mencionados anteriormente a fin de que el alumno complete el ciclo del problema a su solución con computadora, con un método de refinamientos sucesivos en el que se traten simultáneamente las estructuras de control y los datos y en el que se introduzcan los conceptos de corrección y de eficiencia.

Contenidos Mínimos

Modelización de problemas del mundo real.
Algorítmica. Estructuras de control.
Tipos de datos simples y compuestos (estáticos y dinámicos, lineales y no lineales).
Procedimientos y funciones.
Eficiencia y corrección.
Estrategias de diseño de algoritmos.

PROGRAMA

A- Introducción. Conceptos básicos

Definiciones.
Modelización de problemas del mundo real.
Del problema real a su solución por computadora.
Software.

B - Algoritmos. Acciones elementales

Estructuras de control. Modelo de máquina abstracta.
El robot como máquina abstracta. Operaciones elementales del robot.
Estructura esquemática de un programa para una máquina abstracta (robot).
Corrección de algoritmos. Importancia de la verificación.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Eficiencia de un algoritmo. Importancia de la documentación de un algoritmo.
Relación de los conceptos anteriores con el modelo de máquina abstracta.
Ejemplos.

C - Tipos de datos simples

Tipos de datos primitivos.
Constantes y variables.
Funciones predefinidas.
Tipos ordinales.
Tipos de datos definidos por el usuario.

D- Modularización. Procedimientos y funciones. Parámetros

Descomposición de problemas. Utilidad e importancia.
Subprogramas o módulos.
Procedimientos.
Funciones.
Conceptos de argumentos y parámetros.
Conceptos de variables locales y variables globales.
Procedimientos y funciones con parámetros.
La noción de reusabilidad.

E- Estructuras de datos compuestas

Introducción.
Registros.
Pilas.
Colas.
Operaciones sobre cada una de las estructuras mencionadas.
Concepto de tipo definido por el usuario. Extensiones a pilas y colas.
Discusión y alternativa en las definiciones de operaciones válidas sobre pilas y colas.

F- Datos compuestos indexados: arreglos

Clasificación de las estructuras de datos.
Arreglos. Operaciones con arreglos de una dimensión.
Matrices. Tratamiento de información estructurada en vectores y matrices.
Algoritmos de búsqueda.
Algoritmos de ordenación. Ordenación por índice.
Métodos de ordenación eficientes.
Comparación de la estructura de datos arreglo con pilas y colas.

G- Computadoras digitales

Conceptos introductorios. Funcionamiento básico. Organización de un sistema de cómputo, modelo de Von Neumann. Otros modelos de organización, clasificación de



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

las computadoras de acuerdo al modelo. Ejemplos. Evolución histórica de las computadoras y la tecnología empleada en su fabricación. Costo y rendimiento. Análisis de la performance, métodos de medición, MIPS, MFLOPS, benchmarks. Concepto de niveles de abstracción.

H- Aritmética de las computadoras

Definición de bit, nibble, byte, palabra, palabra doble, relación con lenguajes de alto nivel. Representaciones numéricas: números enteros con y sin signo. Aritmética con enteros. Fundamentos de la representación en punto flotante, normalización, error de la representación. Representación estándar del IEEE. Aritmética en punto flotante. Representaciones alfanuméricas, ASCII, EBCDIC.

I- Unidad Central de Procesamiento (CPU)

Organización de la CPU. Descripción de microprocesadores actuales. Modelo de ejecución de instrucciones. Ciclo de instrucción, fases. Comunicación CPU – memoria, dato y dirección. Interconexión de subsistemas, buses, ejemplos reales. Concepto de instrucción. Conjunto de instrucciones: operaciones, formato y modos de direccionamiento. Organización de registros. Lenguaje de máquina y assembly.

BIBLIOGRAFIA

Título: Data structures and algorithm analysis in Java

Autores: Weiss, M.A.

Editorial: Pearson/Addison Wesley

Año de edición: 2011

Título: Introduction to Algorithms

Autores: Cormen T., Leiserson C., Rivest R., Stein C.

Editorial: The MIT Press, 3era Edición

Año de edición: 2009

Título: Organización y Arquitectura de Computadoras – Diseño para optimizar prestaciones

Autores: Stallings W.

Editorial: Prentice Hall, 7ma Edición

Año de edición: 2006

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

Autores: De Giusti A., Madoz C., Bertone R., Naiouf M., Lanzarini L., Gorga G., Russo C. y Champredonde R.

Editorial: Prentice Hall



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Año de edición: 2001

Título: Estructuras de Datos y Algoritmos

Autores: Hernández R., Dormido R., Lazaro J. Ros S.

Editorial: Pearson Education

Año de edición: 2000

Título: Organización de Computadoras

Autores: Tanenbaum A.

Editorial: Prentice, 4ta Edición

Año de edición: 2000

Título: Introduction to Computer Science with applications in Pascal

Autores: Garland S.J.

Editorial: Addison Wesley Publishing Company

Año de edición: 1986

Título: Fundamentos de Programación. Libro de Problemas.

Autores: Joyanes Aguilar L., Fernandez M., Rodríguez L.

Editorial: Mc Graw Hill

Año de edición: 2003

Título: Fundamentos de Programación

Autores: Joyanes Aguilar L., Fernandez M., Rodriguez L.

Editorial: Mc Graw Hill

Año de edición: 1999

Título: Estructuras de Datos y Algoritmos

Autores: Weiss M.A.

Editorial: Addison Wesley Publishing Company.

Año de edición: 1995

Título: Data structures, algorithms and software principles

Autores: Standish T. A.

Editorial: Addison Wesley Publishing Company.

Año de edición: 1994

Título: Estructuras de Datos

Autores: Franch Gutierrez X.

Editorial: Alfaomega Grupo Editor Argentino

Año de edición: 2002



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Título: Programación estructurada en Turbo Pascal 7

Autores: López Roman L.

Editorial: Alfaomega Grupo Editor Argentino

Año de edición: 1998

Descripción de las Actividades Teóricas y Prácticas

La secuenciación de los contenidos se planificó para realizar un camino que inicia por los temas más simples y avanza hacia los más complejos.

Algunos temas, se van presentando de forma general, para luego volver a ellos y profundizarlos a partir de la maduración del alumno, lograda a través de las actividades que se le proponen y de la integración con otros conceptos trabajados.

A lo largo de la materia se van integrando los distintos temas ya tratados para abordar los nuevos, de manera tal que el alumno no tenga una visión fragmentada de la materia si no que pueda apreciar cómo existe una vinculación entre los distintos temas presentados.

Se inicia el recorrido con la presentación de qué es la Informática, y cómo esta se vincula con la resolución de problemas con una computadora, se presentan luego las etapas (modelo lineal de la Ingeniería de Software) de cómo llegar a una solución informática para un problema real.

Se explica que pueden existir diferentes enfoques para el diseño de las soluciones, y que al momento de la implementación se pueden abordar diferentes paradigmas.

En la materia se pone especial foco en la etapa de implementación, y se analizan entonces los conceptos de algoritmo y programa.

Se trabaja posteriormente, sobre la idea que realizar un programa involucra definir un conjunto de instrucciones en un lenguaje de programación, y también representar los datos involucrados en el problema a resolver (PROGRAMA=Instrucciones + Datos).

De allí se parte para abordar el concepto de dato, de tipo de dato, y presentar por otra parte, las estructuras de control como parte del conjunto de instrucciones que ofrecen los lenguajes para escribir los programas.

Por el lado de los datos, se profundiza en los tipos de datos: estándares y definidos por el usuario; simples (enteros, reales, caracteres, boolean) y compuestos (conjuntos y string). Al mismo tiempo, se los va involucrando en los componentes internos de una computadora (que será su herramienta de trabajo), y en la forma en que representan internamente los datos vistos, en particular, los numéricos.

Para el diseño de las soluciones se propone abordar el paradigma procedimental, y es por ello que se trabaja con el concepto de modularización, y se estudian los módulos en Pascal y sus mecanismos de comunicación. Se analiza en profundidad el mecanismo de pasaje de parámetros, y lo que propone Pascal en particular.

A partir de la definición de tipos de datos compuestos, se llega al concepto de estructuras de datos, y se estudian los registros, arreglos, y listas simples. Previo al



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

tema de listas, se profundiza en los conceptos de asignación estática y dinámica que son mencionados al inicio del curso. Se ve a la lista como un ejemplo de estructura de datos de asignación dinámica.

A esta altura el alumno se involucra con problemas de complejidad incremental (respecto a los vistos inicialmente en las actividades y clases de la asignatura), que le permiten desarrollar habilidades propias del pensamiento computacional.

Básicamente, comienza a modelizar soluciones, utilizando los conceptos de abstracción y descomposición funcional, representando los datos con los tipos de datos abordados en la materia, según amerite el problema y eligiendo el conjunto de instrucciones que en una secuencia lógica lo llevarán a la solución del problema.

Finalmente, se le explica al alumno que para un mismo problema podría haber diferentes soluciones, se comienza a trabajar sobre el concepto de corrección, y técnicas de corrección para analizar si un programa es correcto o no.

Luego, se analizan diferentes criterios de calidad de los programas (legibilidad, documentación, características de portabilidad, eficiencia, etc.), y se profundiza en el concepto de eficiencia. Se explican técnicas para analizar la eficiencia de un programa, en relación al tiempo de ejecución.

Todos estos temas se trabajan tanto en las clases teórico/prácticas como en las clases de práctica y consulta. Siempre existe una fuerte vinculación entre los contenidos abordados en las clases teóricas/prácticas y las actividades que se le proponen a los alumnos en las clases prácticas.

Vinculación entre teoría y práctica. Actividades propuestas

El cronograma cumple una función fundamental para que el alumno pueda comprender cómo se articulan los temas y actividades propuestas a lo largo del curso y le facilita planificar sus tiempos en función de la propuesta de la cátedra.

Las explicaciones de prácticas que realiza el JTP ayudan a la vinculación de los temas teórico/prácticos, y a que el alumno logre aplicar los conceptos en las actividades propuestas.

De este modo se articulan los contenidos dictados en las clases teórico/prácticas con ejemplos "generales" que permiten arribar a las soluciones de numerosos casos particulares durante las prácticas.

Las actividades que se proponen son:

- De motivación: visualización de videos, y realización de actividades de Realidad Aumentada para reforzar el concepto de estructuras de control
- De comprensión: lectura de los materiales de estudio propuestos
- De comprensión y aplicación: resolución de problemas tipos presentados en las guías de práctica de la materia (un total de 11 guías)
- De integración: una guía de ejercicios prácticos específicamente planificados para la integración de temas. Los ejercicios son de complejidad similar a los que se proponen para las evaluaciones parciales.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Metodología de Enseñanza y de Evaluación

Metodología De Enseñanza

Programación I combina clases teórico/prácticas con clases exclusivamente de práctica y consulta (de ahora en más se referirá a ellas como clases prácticas).

Las clases teórico/prácticas se dictan 2 veces por semana, y se trata de encuentros de 2 horas reloj cada una, en los que mediante una exposición dialogada se presentan los conceptos fundamentales vinculados a los contenidos de la materia (presentes en el programa que se anexa para esta presentación) y se los ejemplifica a través de ejercicios que permiten al alumno abordar la aplicación de dichos conceptos en casos específicos. Estas son de carácter optativo pero se recomienda a los alumnos su asistencia.

Las clases prácticas se dictan dos veces por semana, y también cada una tiene una duración de 2 horas. Estas clases tienen cierta obligación de asistencia para poder rendir luego el parcial (se explica más adelante). Durante las clases prácticas, los alumnos deben resolver las actividades de la guía de práctica correspondiente a la semana en la que están, y realizar consultas sobre los ejercicios planteados, con los ayudantes y adscriptos. Además, pueden presentar sus principales dificultades a los ayudantes y al JTP que coordina estas clases. Esto permite posteriormente analizar cuáles son los temas con mayor dificultad y definir estrategias adicionales para facilitar el aprendizaje. Por ejemplo, durante el primer semestre de 2015 se agregaron clases de consultas optativas, y en el segundo semestre se incorporaron algunos materiales de estudio adicionales para reforzar la comprensión de algunos conceptos.

Por otra parte, se utiliza un entorno virtual de enseñanza y aprendizaje (EVEA) para extender las posibilidades de las clases presenciales. En el EVEA se cuenta con un curso específico para la asignatura, donde se publican los materiales de estudio en formato digital (.pdf, enlaces a sitios de interés, un material hipermedial sobre estructuras de control, videos optativos, el enlace a Lazarus Pascal, etc.). Aquí se publican también las guías de actividades prácticas. Se utilizan las herramientas de comunicación del EVEA para consultas (herramienta de Mensajería) y presentación de novedades (cartelera). Se publican aquí los resultados de exámenes, los listados de las comisiones, y otros aspectos de carácter administrativo.

La asignatura se dicta en los dos semestres. Durante el primer semestre se trabaja con alumnos ingresantes y recursantes de años previos, mientras que en el segundo semestre se trabaja con alumnos del post-ingreso a la carrera y alumnos recursantes, mayoritariamente se cuenta con alumnos que han realizado su primera cursada de la asignatura durante el primer semestre. Es por esto, que si bien la materia se estructura de la forma ya descrita para ambos semestres, las estrategias didácticas son un tanto diferentes, especialmente durante los encuentros de carácter teórico/prácticos. En el segundo semestre, durante las clases teórico prácticas además de realizar exposición dialogada, se organizan actividades grupales como análisis de soluciones a ciertos problemas para encontrar posibles errores, resolver ejercicios sencillos con evaluación



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

de pares, intercambios de opiniones, entre otros. Además, se los invita a participar de actividades adicionales como la utilización de actividades de Realidad Aumentada para reforzar determinados temas. Esto es debido a dos aspectos fundamentales: a. el grupo del segundo semestre suele ser más heterogéneos habiendo mayor cantidad de recursantes que de ingresantes; b. la matrícula del segundo semestre es algo menor y esto permite llevar adelante dinámicas diferentes durante los encuentros.

Finalmente, es importante destacar que también se cuenta con un blog de la cátedra para la comunicación permanente con aquellos alumnos que han finalizado la cursada y por ejemplo, aún deben rendir el final. Allí se publican fechas y horarios de inicio de cursada, fechas de exámenes finales, y sus resultados.

Se busca así mantener una atención y comunicación frecuente con los alumnos a través de diferentes canales: los encuentros presenciales, la mensajería y la cartelera en el EVEA, el blog y también se hace uso de la cartelera física de la cátedra en la Facultad. Varios de los docentes realizan tareas de investigación en la Facultad por lo que esto facilita al alumno contactarlo.

Metodología De Evaluación

Se realiza un seguimiento del desempeño del alumno en las clases prácticas, y en las clases teórico/prácticas. Además, se analizan las consultas planteadas a través del EVEA.

La cursada se aprueba mediante un parcial práctico. Este tiene 2 instancias recuperatorias. La aprobación de la materia se obtiene por régimen de promoción o por examen final.

A continuación se detallan.

Régimen de Promoción

Aquellos alumnos que optan por el régimen de promoción pueden elegir entre dos alternativas (A o B, según se describe a continuación) con las siguientes condiciones:

A. Rendir 2 Evaluaciones parciales sobre temas teórico-prácticos

1. Se toman dos evaluaciones breves sobre la Teoría en las fechas estipuladas en el cronograma (durante el curso).
2. Para poder rendir la segunda evaluación breve, el alumno debe Aprobar el parcial práctico de la materia en la primera o segunda fecha, y debe haberse presentado a rendir la primera evaluación de promoción sobre la Teoría.
3. Las evaluaciones breves tienen una única nota que se corresponde con el promedio obtenido en ambas actividades.
4. Los alumnos que obtienen una nota promedio de 6, o mayor a 6 en estas evaluaciones breves tienen aprobada la promoción.
5. Los alumnos que, habiendo aprobado el parcial práctico en primera o segunda fecha, no obtuvieran el mínimo de 6 como nota promedio (deben haber rendido las dos evaluaciones de promoción sobre la Teoría y obtener un promedio superior o igual a 4), pueden rendir un examen recuperatorio, que abarcará los temas de la materia que el Profesor le indique.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

B. Rendir una Evaluación parcial sobre temas teórico-prácticos y realizar un trabajo teórico práctico

1. Se toma una evaluación sobre aspectos teórico-prácticos en la fecha estipulada en el cronograma (durante el curso) y se plantea la realización de un trabajo teórico-práctico (a partir de la fecha acordada con el Profesor a cargo de la teoría), que puede realizarse en forma individual o grupal (de hasta tres personas por grupo).
2. Para poder acceder a realizar el trabajo teórico-práctico, el alumno debe Aprobar el parcial práctico de la materia en la primera fecha, y haberse presentado a rendir la primera evaluación de promoción sobre la Teoría, obteniendo una nota mayor o igual a 6.
3. Los alumnos deben aprobar el trabajo teórico-práctico con una nota mayor o igual a 6 para aprobar la promoción
4. Los alumnos que no obtuvieran una nota de 6 o más en el trabajo teórico-práctico pueden acceder a una única re-entrega para mejorar los aspectos del trabajo que el Profesor le indique, en la fecha estipulada por el Profesor de la materia. En caso, de que la reentrega no alcance la nota de 6, el alumno pierde la promoción.

Con examen final

Los alumnos que no hayan aprobado el régimen de promoción, pero hayan aprobado el parcial práctico, obtienen la aprobación de los Trabajos Prácticos y la habilitación para rendir el Examen Final de la asignatura.