Ministerio de Cultura y Educación
Universidad Nacional de San Luis
Facultad de Ciencias Físico Matemáticas y Naturales
Departamento: Informatica
Area: Area IV: Pr. y Met. de Des. del Soft.

(Programa del año 2011)
(Programa en trámite de aprobación)
(Presentado el 21/03/2011 19:56:22)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
PROGRAMACION II	TEC.UNIV.REDES DE COMP.	024/08-CD	2011	1° cuatrimestre
PROGRAMACION II	TEC.REDES COMP.	12/13	2011	1° cuatrimestre
PROGRAMACION II	TCO.UNIV.EN WEB	08/13	2011	1° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
BERON, MARIO MARCELO	Prof. Responsable	P.Adj Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	4 Hs	3 Hs	3 Hs	10 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1º Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
16/03/2011	24/06/2011	15	150

IV - Fundamentación

Con el propósito de recordar y reforzar esos conceptos previos referentes a programación en el presente programa se incluye una unidad inicial que consiste en la revisión de los conceptos de Programación Imperativa usando como lenguaje de programación al lenguaje C. Actualmente este lenguaje es muy usado para el desarrollo de software de base como así también para la interacción con diversos dispositivos. Además el lenguaje C tiene una sintaxis similar a otros lenguajes de amplio uso en la construcción de aplicaciones de Internet. Esta característica es interesante ya que el alumno se familiariza con la notación de esos lenguajes de programación.

El Paradigma Imperativo no es el único, existen otros como por ejemplo el Orientado a Objetos que permiten concebir a la programación desde otras perspectivas. Actualmente, el Paradigma Orientado a Objetos ha alcanzado una amplia popularidad debido a las facilidades que proporciona para la construcción de software. Particularmente, este paradigma se ha utilizado mucho en el desarrollo de aplicaciones de red a través del uso del lenguaje Java. Por esta razón, la enseñanza del Paradigma Orientado a Objetos y de Java se considera esencial en Programación II.

Finalmente, existen lenguajes de programación que posibilitan la utilización de diferentes paradigmas de programación. Dicha posibilidad cuando es apropiadamente usada proporciona una herramienta muy útil para el desarrollo de aplicaciones. Este es el caso del lenguaje de programación Python. Como fue mencionado previamente con Python es posible escribir programas que usen el paradigma imperativo como así también el orientado a objetos y el funcional. Los dos primeros son abordados en las unidades iniciales de la materia mientras que el último, el funcional, sólo será brevemente introducido en la última unidad. Python es un lenguaje ampliamente usado para el desarrollo de aplicaciones de red y proporciona librerías que posibilitan la construcción de sistemas de forma rápida. Estas características hacen de Python un lenguaje atractivo para el

técnico debido a que el desarrollo de sistemas aplicados a la red es una de sus principales funciones.
En el ámbito del desarrollo web existen otras formas de concebir la programación, otros lenguajes de propósito general con aplicaciones a la web y también específicos para el desarrollo de aplicaciones web. No obstante, debido a su importancia para el egresado la enseñanza dichas temáticas son abordadas en cursos superiores de la tecnicatura.
Es importante notar que Programación II presenta temáticas de propósito general, remarcando como las mismas pueden ser aplicadas en el contexto de la red, y prepara el camino para que el alumno pueda asimilar los tópicos específicos de la programación web. De esta manera, el alumno que haya aprobado todas las materias relacionadas con programación tendrá un espectro general de las formas y técnicas de programación y de las herramientas que utilizará en el mercado laboral.
Para finalizar, es relevante mencionar que el programa propuesto, cubre los requisitos descriptos en la ordenanza CD:008/04 de creación de la carrera Tecnicatura Universitaria en Web

V - Objetivos / Resultados de Aprendizaje

Este curso tiene los siguientes objetivos:
*Completar y reforzar los conceptos de básicos de la Programación Imperativa.
*Introducir en los conceptos necesarios para creación de programas que usen el Paradigma de Programación Orientado a Objetos.
*Dar a conocer las bases del paradigma de Programación Funcional.
*Proveer los conceptos y herramientas necesarias para que el alumno pueda desarrollar aplicaciones rápidas con interfaces atractivas.

VI - Contenidos

Unidad 1: Paradigma de Programación Imperativo - Lenguaje C
Revisión de la Noción de Programación: Objetivos, Dificultades y etapas. Paradigma de Programación. Paradigma Imperativo. Lenguaje de Programación C. Características. Creación de un programa escrito en Lenguaje C. Modelo de Compilación. Estructura de un programa. Variables. Constantes. Funciones de Entrada/Salida. Expresiones: Aritméticas, Lógicas, Relacionates, Mixtas. Sentencias de Selección: if, el operador ?, switch. Sentencias de Iteración Iteraciones: while, do-while, for. Tipos de Datos Elementales. Tipos de Datos Estructurados: Arreglos, Cademas de Caracteres, Estructuras, Uniones, Enumeraciones. Punteros: Declaración y asignación de direcciones, Indirección, Operaciones con Punteros, Punteros y Tablas, Punteros y Estructuras. Funciones: Definición y llamada, Parámetros y Variables, Retorno de Resutlados, Pasaje de Parámetros. Archivos: Abrir y Cerrar Archivos, Leer y Escribir Archivos, Archivos Estándar: stdin, stdout, stderr.

Unidad 2: Paradigma de Programación Orientada a Objetos-Lenguaje Java
Introducción. Génesis de la Programación Orientada a Objetos: La Vía de la Simulación. La Vía de la Ingeniería de Software. Abstracción de Datos e Información Oculta. El Concepto de Objeto. El Encapsulamiento: Propiedad Fundamental. Mensajes. Objetos en Programación Orientada a Objetos: Instancias y Clases. Clases e Instancias. Definición de Clases. Creación de Clases. Requisitos Iniciales. Definición de la Estructura. Definición del Comportamiento. Constructores. Métodos de Instancia. La Referencia this. Paquetes e Importación de Clases. Reglas de Accesibilidad a una Clase. Reglas de Acceso a Variables y Métodos de Instancia. Composición en la Definición de una Clase. Métodos y Variables de Clase. Jerarquías de Clases. El Mecanismo de la Herencia. Algoritmo de Búsqueda de Métodos. Superposición de Métodos y Variables. Modificadores y Redefinición de Métodos. Clases sin Subclases. Clases y Constructores. Creación de Clases Via Herencia. Compatibilidad entre Clases y Subclases. Polimorfismo. Aplicación de todos los conceptos previamente mencionados usando el lenguaje de programación Java.

Unidad 3: Programación Multiparadigma – Python
Características Imperativas de Python. Carcterísticas Orientadas a Objeto de Python. Breve introducción a las caracterísiticas funcionales de Python. Desarrollo de Interfaces gráficas rápidas usando Python. Uso de la librería Tkinter. Otras librerías útiles para hacer interfaces gráficas en Python.

VII - Plan de Trabajos Prácticos

Unidad 1: Desarrollo de ejercicios de programación usando el Lenguaje de Programación C. También, en los prácticos de esta unidad se enfatizarán ejercicios que trabajen con tipos de datos abstractos sean primitivos o definidos por el programador.

Unidad 2: Los prácticos de esta unidad estarán divididos en dos partes. La primera de ellas se centrará en la familiarización del alumno con el Lenguaje de Programación Java. Para este fin se presentará una guía de prácticos simples que impliquen el uso de tipo básicos y sentencias primitivas. La segunda se basará en el uso de los conceptos de programación orientada a objetos.

Unidad 3: La parte práctica de esta unidad se dividirá en cuatro partes. En la primera se utilizarán las características imperativas del lenguaje Python. En la segunda se utilizarán las características orientadas a objetos del lenguaje Python. En la tercera, se utilizarán librerías gráficas para el desarrollo de interfaces con el usuario. En esta etapa se intentará resolver ejercicios que impliquen el uso de algoritmos que el alumno utilizará en el mercado laboral. Finalmente, en la última parte se mostrarán brevemente las características funcionales de Python.

Laboratorios:
Cada unidad tendrá una laboratorio que consistirá en el desarrollo de un programa que resuelva un problema de aplicación utilizando el paradigma de programación y lenguaje que se estudia en la unidad correspondiente.

VIII - Regimen de Aprobación

Aprobación de la Asignatura:
*Regularización de la Asignatura:
-Se requiere un porcentaje de asistencia del 80%.
-El alumno debe aprobar tres exámenes parciales o sus correspondientes recuperaciones, y presentar en tiempo y forma los prácticos de máquina solicitados por la cátedra.

*Promoción de la Asignatura: El alumno debe cumplir con las condiciones de regularización y aprobar los exámenes parciales con un nivel superior o igual al 70% del total. Se tomará una recuperación general adicional a los alumnos que hayan presentado certificado de trabajo en tiempo y forma.

*Examen Final: Los alumnos rendirán un examen teórico-práctico el cual puede ser escrito u oral.

*Examen Libre: Los alumnos que habiendo aprobado todos los prácticos de máquina hayan quedado libres por parciales, y en este caso, por la no aprobación de uno y no más de los temas que componen la materia, pueden presentarse como alumnos libres. Una lista con los datos de los alumnos habilitados para rendir el examen libre será elevada a sección de alumnos junto con las actas correspondientes a la finalización de la materia. La posibilidad de rendir el examen en calidad de libre expira con el inicio del próximo dictado de la materia.

IX - Bibliografía Básica

[1] Rossum, G; Drake, F. “Python Tutorial”. Python Software Foundation, version en español.
[2] Material de la Cátedra “Programación Orientada a Objetos” del Prof. Mario Martins. Universidade do Minho. http://sim.di.uminho.pt/ensino2.php3?seccao=geral&id=90.
[3] Eckel, B. “Thinking in Java”. Versión en español. President, Mind View Inc.
[4] Kernighn, Brian; Ritchie, Dennis. "El Lenguaje de Programación C". Editorial: Prentice-Hall. Edición 1991.
[5] Material proporcionado por la cátedra

X - Bibliografia Complementaria

[1] Meyer, B. “Object Oriented Software Construction”. Second Edition. Pretince Hall. 1997.
[2] Niemeyer, P; Knudsen, J. “Learning Java”. Editorial: O’Really. ISBN: 1-56592-718-4. 722 páginas.
[3] Norton, P; Samuel, A. Aitel, D; et.al. “Beginning Python”. Wiley Publishing, Inc. 2005. ISBN-10: 0-7645-9654-3
[4] Ascher, D; Lutz, M. “Learning Python”. O'Reilly Second Edition. 2004. ISBN: 0-596-00281-5.

XI - Resumen de Objetivos

*Reforzar conceptos de programación imperativa.
*Introducir los conceptos fundamentales de la programación orientada a objetos.
*Proporcionar conceptos y herramientas para el desarrollo de aplicaciones rápidas.

*Dar a conocer los conceptos básicos de la programación funcional.

XII - Resumen del Programa

Unidad I: Programación Imperativa.
Unidad II: Programación Orientada a Objetos.
Unidad III: Programación Multiparadigma y Desarrollo de Aplicacione Rápidas.

XIII - Imprevistos

Serán resueltos por los integrantes del área en el momento apropiado en la medida de lo posible.

XIV - Otros

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
	Profesor Responsable
Firma:	
Aclaración:	
Fecha:	