



CARRERA de TÉCNICO SUPERIOR en ANÁLISIS de SISTEMAS

Resolución 1148-MEGC-2007

PARADIGMAS DE PROGRAMACIÓN

Curso: 1° Año

Profesor: Leandro E. Colombo Viña

Ciclo Lectivo: 2015

Régimen: Cuatrimestral. Segundo Cuatrimestre.

Carga Horaria: 5Hs. Cat./Sem.

1. Fundamentación:

Esta instancia curricular brinda herramientas para que los estudiantes puedan interpretar, en el contexto de un proyecto, especificaciones o requisitos de las asignaciones a programar. Comprende validar la coherencia e integridad de las mismas y convalidar su propia interpretación con quienes la hayan realizado o provisto. A su vez, plantea la realización de operaciones tales como la verificación del cumplimiento de las especificaciones recibidas; la planificación y documentación de casos de prueba; la preparación de datos y entornos de prueba ("testing") y la generación de código adicional para simular el entorno o activar las pruebas. Todos estos procedimientos posibilitarán que los estudiantes analicen las causas de comportamientos o resultados no previstos para proponer correcciones respecto del código realizando así un mejoramiento de la eficiencia ("tunning") de la solución propuesta.

2. Objetivos:

Que los estudiantes:

- ✓ Conozcan los fundamentos de diseño de programas basados en la descomposición modular y la programación estructurada.
- ✓ Adquieran los conocimientos fundamentales de programación tomando como base un lenguaje de programación.

3. Promoción:

- ✓ Asistencia al 75% de las clases.
- ✓ Aprobación de los Trabajos Prácticos.
- ✓ Aprobación de 2 (dos) exámenes parciales.
- ✓ Aprobación de 1 (un) examen final.

4. Ejes de contenidos:

- Conceptos generales. El proceso de programación. Diseño de programas.
- Codificación de programas. Compilación de un programa. Elementos para codificar programas.
- Paradigmas de Programación: imperativo, orientado a objetos, funcional, lógico.
- La codificación de las estructuras de control. Tipos de selección: simple (if-else) o múltiple (switch). Tipos de repetición: do-while y for.
- Algoritmos fundamentales: recorrido, búsqueda, ordenamiento, actualización.
- Codificación de un programa con arreglos. Manejo de los elementos de un arreglo. Manejo de los elementos de una matriz.
- Diseño descendente. Proceso de modularización.
- Utilización de funciones que no regresan valor (módulos).
- Validación de la entrada de datos.
- Creación de programas que utilizan registros y archivos. Proceso de un archivo secuencial. Proceso de un archivo directo.

5. Programa analítico de la asignatura:

La asignatura se divide en 3 (tres) unidades temáticas que nuclea los contenidos según la relación existente entre los mismos.

UNIDAD 1: PROGRAMAS SENCILLOS

Construcción de programas. Un ejemplo sencillo. Piezas de un programa Python. Nombres. Expresiones. Tipos de datos. Instrucciones. Ciclos definidos. Guía para el diseño. Documentación de funciones. Imprimir vs Devolver. Uso de funciones en un programa. Resultados de las funciones. Un ejemplo completo. Devolución múltiples resultados. Expresiones booleanas. Expresiones de comparación. Operadores lógicos. Comparaciones simples. Múltiples decisiones consecutivas.

UNIDAD 2: USANDO FUNCIONES

El problema de la búsqueda. Búsqueda lineal. Búsqueda binaria. Apertura y cierre de archivos. Procesamiento. Escritura y Lectura. Manipulación binaria. Persistencia de datos (CSV y binarios). Directorios. Corte de control. Apareo. Errores. Excepciones. Manejo. Procesamiento y propagación. Acceso a la información de contexto. Validaciones.

UNIDAD 3: APLICACIONES

Concetos web: HTTP y HTML. Diseño de aplicaciones. Modelos de datos. Patrón MVC: modelo – vista – controlador. Utilización de framework.

6. Bibliografía:

| Autor | Título | Editorial | Observaciones |
|---|---|---|---------------------------------|
| Allen Downey, Jeffrey Elkner & Chris Meyers | Aprenda a Pensar Como un Programador con Python | Green Tea Press. Año: 2002 | ISBN: 0-9716775-0-6 |
| Rosita Wachenchauzer & Colaboradores | Algoritmos y Programación I Aprendiendo a programar usando Python como herramienta | Apuntes de la Facultad de Ingeniería de la UBA. | ISBN: N/D |
| Raúl González Duque | Python Para Todos | http://mundogeek.net/tutorial-python/ | ISBN: N/D |
| Comunidad Python Argentina | Tutorial de Python | http://docs.python.org.ar/tutorial/ | Proyecto de PyAr |
| Paul Barry | Head First Python | O'Reilly. Año: 2011 | ISBN: 978-1-449-38267-4 |
| Eugenia Bahit | Curso: Python para Principiantes | http://library.originalhacker.org/biblioteca/articulo/autor/2 | Safe Creative: 1207302042960 |
| Donald E. Knuth | The Art of Computer Programming. Volume I. | Addison-Wesley. Año: 1997 | ISBN: 0-201-89683-4 |
| Miguel Grinberg | The Flask Mega-Tutorial | http://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world | ISBN: N/D |

7. Planificación:

| Clase | Unidad | Contenido | Actividad |
|-------|--------|--|---|
| 1 | I | Introducción a la materia. Conceptos de programación estructurada. | Diálogo con la clase. Metodología de trabajo. Componentes de un programa en Python. |
| 2 | I | Manejo de archivos. | Modo de apertura de archivos. Ejemplo de procesamiento de archivos. Cierre de archivo. Escribir y leer un archivo. Persistencia de datos. |
| 3 | I | Manejo de errores y excepciones. | Definiciones. Errores. Excepciones. Procedimientos y propagación. Validaciones. |
| 4 | I | Procesamiento de archivos. | Cortes de control. Apareo. Buenas prácticas. |
| 5 | II | Modelo de ejecución de funciones. Recursividad. | Pila de ejecución de funciones. Pasaje de parámetros. Devolución de resultados. Recursividad. Limitaciones. |
| 6 | II | Ordenamientos. | Ordenamientos simples. Por selección y por inversión. |
| 7 | II | Ordenamientos recursivos. | Ordenamientos avanzados. Por mezcla, rápido. Características. |
| 8 | - | Revisión de contenidos. | Evaluación parcial escrita. |
| 9 | III | Diseño de aplicaciones Web. | Concepto básico de la Web. Comunicación web: HTTP y HTML. Requerimientos. Estructura de un proyecto. |
| 10 | III | Plantillas. | Plantillas web. Uso de las plantillas. Estructuras de control y ciclos dentro de las plantillas. Herencia. |
| 11 | III | Formularios. | Formularios web. Ingreso de datos. Validaciones. Plantillas de formularios. Vistas. Recibiendo datos. |
| 12 | III | Introducción breve a las Bases de Datos. | Conceptos básicos de bases de datos. Usos. Modelos. Relaciones. |
| 13 | III | Registro de usuarios. | Integración de conceptos. Desarrollo de aplicaciones web con registros de usuarios. |
| 14 | - | Revisión de contenidos. | Evaluación parcial escrita. |
| 15 | - | Revisión de contenidos. | Recuperatorio de evaluación parcial escrita. |
| 16 | - | Revisión de contenidos. | Recuperatorio de evaluación parcial escrita. |