



GOBIERNO DE LA CIUDAD DE BUENOS AIRES

INSTITUTO DE FORMACIÓN TÉCNICA SUPERIOR (IFTS) N° 4

ASIGNATURA: PARADIGMAS DE PROGRAMACION

AÑO: Primer Año

APELLIDO Y NOMBRE PROFESOR: CIRELLO, Silvia

AÑO: 2017

FUNDAMENTACIÓN:

La materia parte de la necesidad educativa y social de actualizarse en los saberes tecnológicos. Ello es así ya que su manejo determina el grado de inclusión en los avances tecnológicos y las destrezas necesarias para asimilar las permanentes mutaciones a las que nos vemos expuestos a raíz de los grandes cambios de este siglo con respecto a las formas de comunicación y las características cognitivas que identifican al hombre actual y del futuro.

Para lograr una formación integral en Sistemas computacionales se deben conocer los términos científicos propios del área, sin dejar de lado los conocimientos básicos de lógica y estadística que son la base de entendimiento y desarrollo de software.

Los avances tecnológicos fueron acompañando el desarrollo social y permitiendo la transformación de paradigmas que fueron influyendo en las formas de vivir, trabajar y relacionarse. La educación no fue ajena a estos fenómenos, y en consecuencia pueden percibirse distintos paradigmas educativos. Pero los paradigmas en los que se encuadra esta materia tienen que ver con los conceptos de algorítmica elemental con el propósito de introducir a los estudiantes, el concepto de la eficiencia de los programas. El análisis de algoritmos se percibe con una serie de ejercicios aplicados, que incluyen la práctica en laboratorio con el propósito de que, los estudiantes asimilen estos conceptos en forma experimental.

Al punto que, la programación es útil para cualquiera: tanto para la secretaría administrativa que usa hojas electrónicas de cálculo como para el programador de alta tecnología. Enseñar programación es dar conocimiento, es instruir las destrezas lógicas, es analizar problemas y su resolución sin la imposición de notaciones y herramientas tradicionales.

Finalmente, usar un generador de programas en muchas formas depende de las mismas habilidades que fueron necesarias para los paquetes de aplicación. Un generador de programas crea un programa en un lenguaje de programación tradicional, como C++ o Java, Pseint, desde descripciones de alto nivel, como las reglas de negocio o leyes científicas. Tales reglas de manera típica relacionan cantidades, ventas, y registros de inventarios y sus cálculos específicos.

OBJETIVOS GENERALES:

- Comprender los fundamentos de los paradigmas básicos que son utilizados por los lenguajes de programación actuales.

OBJETIVOS ESPECÍFICOS:

- Reconocer los diferentes tipos de paradigmas;
- Comprender el vocabulario de pseudocódigo e interpretar diagramas de flujo;
- Aplicar los diferentes paradigmas en la resolución de problemas;
- Aplicar lógica y lograr pensamiento continuo en los procesos a resolver.

CONTENIDOS:

Conforme los objetivos planteados para este espacio curricular se lo ha dividido en tres módulos o líneas de trabajo (teórica/práctica) principales. El Modulo III se implementara junto con el modulo I y II como forma práctica.

Módulo I. Conceptos Básicos de Programación.

Concepto de algoritmo. Modelos computacionales. Programas: Algoritmos para ser ejecutados por un ordenador. Lenguaje de máquina. Lenguaje ensamblador. Programación para seres humanos. Lenguajes de alto nivel. Traductores e intérpretes. Lenguajes interpretados vs. Lenguajes compilados.

Módulo II. Tipos de paradigmas de programación.

Imperativo o por procedimientos. Funcional. Lógico. Declarativo. Orientado a objetos. Programación multiparadigma.

Modulo III. Algoritmos y técnicas de programación.

Formas de comunicación de los algoritmos: diagramas estructurados. Pseudocódigo. Ordinogramas. Técnicas de programación, ciclos, condicionales, ejecución y compilación. Uso de Soft: Pseint. Pseudocódigo y Diagramas de flujo. Técnicas informática de tratamiento de los problemas: modularidad, generalización.

MODALIDAD DE EVALUACIÓN:

Criterios de evaluación: Los siguientes ítems contemplan la forma que el docente evaluará a los alumnos:

- Trabajos prácticos de elaboración individual o grupal, se tendrá en cuenta:
 - Aplicación de conocimientos vistos,
 - aporte personal en el trabajo en equipo,
 - propuesta creativa y compromiso frente a la tarea,
 - entrega en tiempo y forma.
- Coherencia en la exposición o redacción;
- Dominio de contenidos y bibliografía;
- Vocabulario técnico adecuado;
- Capacidad de síntesis.
- Cumplir con el 75% de asistencia a las clases presenciales.

BIBLIOGRAFÍA Y WEBGRAFIA

- David Watt, Programming Languages Concepts and Paradigms, Prentice Hall. 2009
- F. Alonso Arno y F. Segovia Pérez, Entornos y Metodologías de Programación, Paraninfo. 2005
- Carlo Ghezzi y Mehdi Jazayeri, Conceptos de Lenguajes de Programación, Díaz de los Santos. 2004
- <http://pseint.sourceforge.net/>