



TECNICATURA SUPERIOR EN DESARROLLO DE SOFTWARE

CICLO LECTIVO 2014

Carrera	Tecnicatura Superior en Desarrollo de Software
Resolución N°	462

Unidad Curricular:	Programación II
Campo de Formación:	FE
Formato Curricular:	Asignatura
Horas Cátedras Semanales:	6 hs.
Horas Cátedras Anuales:	192 hs.
Curso:	2° Año
División:	A
Ciclo Lectivo:	2014
Profesor/a:	Martin Onetti

1. FUNDAMENTACIÓN

Esta materia lo que intenta es adentrar a los alumnos mas específicamente en el mundo de los lenguajes de programación.

Para ello se propone el recorrido por variados conceptos referentes a este universo. El objeto de eso es conocer lo ya estudiado y comprender su funcionamiento.

Esta materia se independiza del lenguaje de programación a utilizar ya que lo que se ve es conceptual. Se propone sin embargo el lenguaje python como herramienta por su simple y potencia a la hora de expresar estos conceptos

Se asume que los alumnos ya tienen contacto con la programación (ya que Programación I) es correlativa con esta materia

2. OBJETIVOS

2.1. Objetivos generales



- Conocimiento de algoritmos y Estructuras de Datos
- Resolución de algunos tipos de problemas utilizando algoritmos y estructuras de datos
- Contribuir a optimizar las condiciones de estudio/aprendizaje de los alumnos.

2.2. Objetivos específicos

- Estudio de Tipos Abstractos de Datos conocidos
- Estudio e implementación de Algoritmos conocidos para la ordenación de elementos
- Estudio e implementación de algoritmos conocidos para la búsqueda de objetos
- Implementación de Algunos tipos de datos Abstractos
- Promover el trabajo en equipo y la producción individual de calidad.
- Fortalecer las capacidades de resolución de problemas
- Desarrollo de software

3. APRENDIZAJES Y CONTENIDOS

Unidad I

Repaso. Tipos de Datos. Definición. Clasificación: Abstractos vs Pre-Definidos. Uso de Tipos de Datos Pre-definidos. Enteros, Caracteres y Booleanos: Propiedades y Operaciones sobre los mismos. Listas. Operaciones sobre listas

Unidad II

Estudio de sendos casos de TAD. Especificación e Implementación: PILAS, COLAS y LISTAS ENLAZADAS

Unidad III

Recursión. Casos de Estudio. Torres de Hanoi. Cálculo del Factorial. Algoritmos de Euclides. Los números de Fibonacci

Unidad IV

Algoritmos de Búsqueda. Búsqueda Lineal. Código. Implementación. Búsqueda Binaria. Implementación

Unidad V

Algoritmos de Ordenación. Métodos Iterativos: Método de la Burbuja, de la Inserción y de la Selección. Métodos Recursivos: Ordenamiento por Mezcla y Ordenamiento Rápido. Nociones de complejidad de los algoritmos. Ejercitación.

Unidad VI

Árboles. Definición. Propiedades. Algoritmos conocidos sobre árboles. InOrden, PreOrden y PostOrden. Ejercitación



4. ORIENTACIONES PARA LA ENSEÑANZA

La materia esta estructurada en clases teóricas y prácticas, haciendo hincapié en estas últimas. A lo largo del desarrollo se irán entregando prácticos con ejercicios relacionados con los contenidos de la materia, teniendo un horario para realizarlos en clase.

Si bien estos prácticos no serán evaluables, se los motivará en la resolución de ellos dado que los trabajos estarán basados en estos.

A partir de la mitad del año las clases se desarrollaran en el laboratorio con el fin de utilizar las maquinas para programar los ejercicios de los prácticos así como también el proyecto de integración.

A su vez se pretende que fortalecer en el/la alumno/a las siguientes habilidades:

- Identificar las herramientas que nos brinda un lenguaje y buscar el mejor uso de estas.
- Lograr implementar objetos a partir de conceptos
- Lograr implementar algoritmos para resolver problemas a partir de los prácticos planteados

5. INSTANCIAS Y CRITERIOS DE EVALUACIÓN

5.1. Trabajos prácticos evaluativos

Son la principal herramienta de evaluación aprendizaje. Se pretende que los estudiantes resuelvan ejercicios desde un punto de vista eminentemente práctico. Así cada unidad expuesta en la seccion "Aprendizajes y Contenidos" de este documento consta de 1 ó 2 prácticos donde se evaluara la unidad.

5.2. Exámenes parciales

Se realizarán 2 parciales. En estos se evaluarán además de ejercicios con resolución eminentemente práctico, consignas que tengan que ver con la resolución más conceptual del problema.

5.3. Recuperatorios

Cada instancia evaluativa (parciales y exámenes parciales) tendrá su respectivo recuperatorio. Se prevé que los recuperatorios de parciales se realizan en la fecha posterior a la fecha en que ocurre el examen parcial mientras que los recuperatorios de trabajos practicos evaluativos serán realizados en los momentos finales del cursado de la materia

5.4. Exámenes finales



El examen final consta de ejercicios para resolver a través del desarrollo de pequeños programas y de consignas para resolver mediante el razonamiento más conceptual.

Para el caso de los alumnos libres se agregarán dos ejercicios más indicando esta situación

5.5. Proyectos de integración

El proyecto de integración estará en el marco de la Feria de Ciencias que se realiza oportunamente cada año. De no realizarse se podrá prescindir de este realizando más prácticos sobre temáticas particulares.

El desarrollo del proyecto será en forma grupal. Cada grupo no superará los 3 integrantes

6. REQUISITOS PARA ACREDITAR

CONDICIÓN DE REGULARIDAD

- Asistencia a clases: Se considera la asistencia de por lo menos el 75% a clases
- Trabajos prácticos evaluativos: Se realizaran prácticos evaluativos requiriendo al menos el 80% de los mismos
- Parciales: Se necesitan 2 parciales aprobados con al menos la calificación de 4
- Examen final: Un examen final que sera aprobado con calificación mayor o igual a 4

CONDICIÓN DE PROMOCIÓN

Promoción directa

Para promocionar se necesita desarrollar completamente el proyecto de laboratorio. Este consiste en el desarrollo de un videojuego.

7. ARTICULACIÓN

En el segundo año de cursado los alumnos asisten a materias que describen distintos aspectos de la informática. Es así que muchos conceptos que los alumnos aprenden/conocen en otros espacios curriculares son retomados y enmarcados en el contexto de la programación.

De esto surge dos estrechas vinculaciones a nivel de contenidos: Sistemas Operativos y Modelado de Sistema

8. BIBLIOGRAFÍA



BIBLIOGRAFÍA BÁSICA

Wachenchauzer y varios – Algoritmos y Programación I con Lenguaje Python
FLOREZ RUEDA, Roberto (2005) - Algoritmos, Estructuras de Datos y Programación Orientada a Objetos
MARZAL, Andrés; GRACIA, Isabel (2003) - Introducción a la Programación en Python

SITIOS WEB CONSULTADOS – SUGERIDOS

Sitio Oficial Python: <http://www.python.org>
Sitio Oficial Pilas Engine: <http://pilas-engine.com.ar>

BIBLIOGRAFÍA COMPLEMENTARIA

BLANCO, Javier; SMITH, Silvina; Barsotti, Damián - Calculo de Programas
BACKHOUSE, Roland (2007) - Algorithmic Problem Solving