



Algoritmos y Estructuras de Datos Planificación Ciclo lectivo 2022

Datos administrativos de la asignatura			
Departamento:	Sistemas	Carrera	Ing. En Sistemas de Información
Asignatura:	Algoritmos y Estructura de Datos		
Nivel de la carrera	I	Duración	
Bloque curricular:	Tecnologías básicas - Programación		
Carga horaria presencial semanal:	5	Carga Horaria total:	160
Carga horaria no presencial semanal (si correspondiese)		% horas no presenciales (si correspondiese)	
Profesor/es Titular/Asociado/Adjunto:	Adj. Ing. Mario Rinaldi	Dedicación:	1
Auxiliar/es de 1º/JTP:	JTP Ing. Pablo Parruccia JTP Ing. Paola Simieli JTP Ing. Paula Toselli	Dedicación:	

Presentación, Fundamentación

Algoritmos y Estructura de Datos contribuye a que los alumnos obtengan las competencias necesarias para la resolución de problemas propios de la ingeniería, que incluyen procesos informáticos.

- **Relación de la asignatura con el perfil de egreso.** Se vincula con el perfil, aportando competencias en la resolución de problemas relacionados con el procesamiento de la información.
- **Relación de la asignatura con los alcances del título.** La relación directa de la cátedra con los alcances del título se establece al brindar los aspectos básicos en lo referido al desarrollo, implementación y prueba de un sistema de información., (Describir la relación y los aportes de la asignatura con los alcances del título).



--

Relación de la asignatura con las competencias de egreso de la carrera

Detallar, en la tabla siguiente, la relación de la asignatura con las competencias de egreso específicas, genéricas tecnológicas, sociales, políticas y actitudinales de la carrera. Indicar a cuáles competencias de egreso tributa (aportes reales y significativos de la asignatura) y en qué nivel (0=no tributa, 1=bajo, 2=medio, 3=alto). Agregar un comentario general de justificación. (Este detalle se integrará en una matriz de tributación de la carrera, dictada en la Facultad Regional, en la cual se explicita el desarrollo de las competencias específicas y genéricas de la carrera y el nivel en que tributa cada asignatura).

Competencias específicas de la carrera (CE)	Competencias genéricas tecnológicas (CT)	Competencias genéricas sociales, políticas y actitudinales (CS)
CE1: 1.3 Especificar, proyectar y desarrollar software. Nivel 1	CT1: Identificar, formular y resolver problemas de ingeniería. Nivel 1	CS1: CG9 Aprender en forma continua y autónoma. Nivel 1
CE2:	CT2:	CS2: CG6 Desempeñarse de manera efectiva en equipos de trabajo. Nivel 1
CE3:	CT3:	CS3:
CE...:	CT...:	CGS:

Propósito

Desarrollar las competencias básicas necesarias para que puedan interpretar y resolver un problema mediante la implementación de algoritmos utilizando un lenguaje informático con la estructura de datos adecuada.

Objetivos establecidos en el DC

- Identificar problemas algorítmicos
- Conocer el proceso de diseño e implementación de software
- Aplicar las herramientas fundamentales representativas de los procesos, integrando la sintaxis elemental de un lenguaje de programación en el laboratorio asociado..



Resultados de aprendizaje
<p>Describir y explicar los Resultados de aprendizaje a promover en el desarrollo de la asignatura. Argumentar su cantidad, sus componentes y la manera en que cada resultado de aprendizaje contribuye al desarrollo de las competencias que aborda la asignatura:</p> <ul style="list-style-type: none">• RA1: Aplica los conceptos del Pensamiento Computacional para resolver problemas básicos de ingeniería utilizando las estructuras de datos correspondientes con el lenguaje informático indicado .• RA2: Participa en la implementación de soluciones en conjunto con pares debatiendo y consensuando diferentes alternativas de resolución.
Asignaturas correlativas previas
<p>Para cursar debe tener cursada:</p> <p>Para cursar debe tener aprobada:</p> <p>Para rendir debe tener aprobada:</p>
Asignaturas correlativas posteriores
<p>Indicar las asignaturas correlativas posteriores:</p> <ul style="list-style-type: none">• Asignatura : Sintaxis y Semántica del Lenguaje• Asignatura: Paradigmas de Programación• Asignatura: Análisis de Sistemas de Información

Programa analítico, Unidades temáticas
<p><u>Unidad 1:</u> Introd. al pensamiento computacional (PC). Solución de problemas. Conceptos básicos del PC: Pensamiento algorítmico, Descomposición, Patrones , Abstracción, Evaluación. Soft utilizado: Light Bot / Karel</p> <p><u>Unidad 2:</u> Resolución de problemas con computadoras. Fases en la resolución de problemas. Abstracción. Programación modular. Programación estructurada. Algoritmos. Refinamientos sucesivos. Mapas de resolución. Algoritmo del programador y pensamiento algorítmico. Soft utilizado: Karel</p> <p><u>Unidad 3:</u> Estructura general de un programa: Concepto de programa. Partes de un programa Instrucciones. Datos. Tipos de datos. Operaciones primitivas. Constantes. Funciones internas. Abstracción procedimental. Abstracción de Datos.</p>



Soft utilizado: Pseint / Python

Unidad 4: Estructuras selectivas. Flujo de control del programa. Estructura secuencial. Estructuras selectivas. Alternativa simple (if/then). Alternativa múltiple (if/then/else). Estructuras de decisión anidadas.

Soft utilizado: Pseint / Python

Unidad 5: Estructuras repetitivas: Mientras. Hacer-Mientras. Repetir. Desde / Para. Salida internas de los bucles. Interrupción. Invariantes

Soft utilizado: Pseint / Python

Unidad 6: Subprogramas, procedimientos y funciones: Introducción. Funciones. Procedimientos. Ambito de las variables. Paso de parámetros. Recursividad.

Soft utilizado: Pseint / Python

Unidad 7: Arrays. Vectores. Operaciones con vectores. Arrays multidimensionales.

Soft utilizado: Pseint / Python

Unidad 8: Cadena de caracteres. El juego de caracteres. Datos tipo carácter. Operaciones con cadenas. Funciones de cadenas de caracteres (Longitud. Subcadena. Concatenación)..

Soft utilizado: Pseint / Python

Unidad 9: Archivos: Introducción. Conceptos de campo, registro, archivo. Organización de archivo: secuencial, directa, secuencial indexada. Operación sobre archivos: creación, consulta, actualización. Procesamiento de archivos (ABM).

Soft utilizado: Pseint / Python

Unidad 10: Ordenación. Métodos de la burbuja, inserción, selección, Shell. Búsqueda secuencial y binaria.

Soft utilizado: Pseint / Python

Unidad 11: Estructuras lineales de datos. Listas. Pilas. Colas

Soft utilizado: Pseint / Python

Unidad 12: Complejidad computacional. Orden de complejidad.

Unidad 13: TAD. Abstracciones de datos. Tipos de datos. Formas de abstracción

Soft utilizado: Pseint / Python



Metodología de enseñanza

El proceso de enseñanza – aprendizaje se basa en la aplicación de metodologías de aprendizaje activo: clase invertida, gamificación, aprendizaje de a pares, etc. El alumno recibe previamente a cada clase, el video que debe ver, el apunte o parte de la bibliografía a leer y un cuestionario a contestar. Al inicio de la clase se realiza una evaluación de los conceptos insertos en los videos y apuntes que debe aprender en forma previa a la clase.

Cada evaluación semanal junto a las evaluaciones parciales (formativas y sumativas) junto a los proyectos a presentar, tienen un puntaje cuya sumatoria es de 100 puntos.

En base a las consultas recibidas por parte de las alumnas, se resuelven las dudas presentadas. La resolución de ejercicios y problemas planteados en clase, se resuelven bajo la metodología de aprendizaje de a pares.

Luego se discuten diferentes alternativas de solución. Se estimula al alumno para que participe activamente de la clase, resolviendo diversas situaciones problemáticas presentadas, a efectos que el alumno vaya internalizando los diferentes mecanismos de resolución. Se utiliza intensivamente el laboratorio en las clases prácticas, a efectos de que el alumno adquiera la solvencia básica necesaria, en la resolución de los algoritmos propuestos en una guía de ejercicios.

Recomendaciones para el estudio

Es muy importante tener incorporadas eficientes técnicas de estudio y lectura comprensiva. En caso de reconocer algún déficit en estas áreas, se recomienda el desarrollo de las mismas mediante de MOOCs específicos como el que brinda Coursera : Aprendiendo a aprender

<https://es.coursera.org/learn/aprendiendo-a->

Además es importante revisar/actualizar conceptos sobre :

- Método Polya para resolución de problema.
- Uso de mapas conceptuales.
- Uso UV Gowin.

Metodología de evaluación

Se adopta un sistema de evaluación continua, donde el alumno tendrá un feedback de su avance en forma semanal. A esto se le incorporan evaluaciones acreditativas y trabajos prácticos. I). La asistencia deberá ser del 75% o más.(Ord. 1549)



Se valorará la participación y la dedicación del alumno en las clases.

Los parciales son ejercitaciones prácticas, que se realizan en los laboratorios de informática, utilizando el software que indiquen los docentes.

El examen final es teórico – práctico. Se exigen las mismas condiciones de resolución que en los exámenes parciales. Si el alumno supera la instancia práctica pero no aprueba la parte teórica, en el próximo examen deberá rendir nuevamente la parte práctica.

Los alumnos dispondrán de horarios de consultas presencial y virtual (se acuerdan con los alumnos).

- **Evaluación de cada Resultado de Aprendizaje.** Los instrumentos a utilizar se componen de evaluaciones periódicas mediante MC, presentación de actividades propuestas, evaluaciones parciales y un T.P. integrador.

Condiciones de aprobación: Si el alumno logra el 80% o más de los puntos, obtiene la aprobación directa (A.D) de la asignatura. En caso contrario, el alumno deberá presentar un Examen Final (Aprobación No Directa).

-

Cronograma de clases/trabajos prácticos/exámenes (tentativo)

- Ver documento adjunto.

Recursos necesarios

- Espacios Físicos : aulas y laboratorios. Los alumnos traen sus notebooks/netbooks a clase.
- Recursos tecnológicos de apoyo : proyector multimedia. Moodle . Actividades en diversos MOOCs. Software: Karel, Pseint, Lightbot, Python.

Referencias bibliográficas (citadas según Normas APA)

Bibliografía obligatoria

Herrera Morales Jorge Orlando (2017). Introducción a la lógica de programación. Colombia. Editorial Elizcom S.A.S.

Gagliano Gracia María (2014). Elementos esenciales para programación: Algoritmos y Estructura de Datos. Editorial Iniciativa Latinoamericana de Libros de Texto Abiertos.

Cárdenas David (2015). Pensamiento lógico computacional. México. Editorial Digital (Tecnológico de Monterrey).

Bibliografía optativa

Joyanes Aguilar Luis (2008). Fundamentos de Programación. España. Editorial Mcgraw Hill.



Función Docencia
Capacitaciones relacionadas con la formación por competencias y aprendizaje centrado en el alumno.
Reuniones de asignatura y área
Las reuniones son semanales de forma virtual. Además se dispone de un grupo de WA para mantenernos al tanto de todas las situaciones que se van presentando.
Atención y orientación a las y los estudiantes
Detalle y cronograma de actividades de atención y orientación a las y los estudiantes (dentro y/o fuera del horario de clase) <ul style="list-style-type: none">● Momento de recuperación de actividades no cumplidas.● Actividades previas a la clase que deben realizar los y las estudiantes (sugerencias de revisión de conceptos teóricos y actividades prácticas, así como un recordatorio de las actividades pendientes). Previo a cada clase, se especifican actividades a realizar vía Moodle.● Actividades posteriores a la clase que deben realizar los y las estudiantes, en horario no presencial. Consultas utilizando el aula virtual. Realización de tutorías.● Actividades de aprendizaje autónomo. Al utilizar la metodología de clase invertida, el estudiante debe realizar esta actividad en forma periódica.



ANEXO 1: FUNCIÓN INVESTIGACIÓN Y EXTENSIÓN (si corresponde)

En este Anexo 1 (a completar si correspondiese) la cátedra detallará las actividades previstas respecto a la función docencia en el marco de la asignatura.

Lineamientos de Investigación de la cátedra

Para introducir a las y los estudiantes a las actividades de investigación que realiza la cátedra. Se recomienda incorporar al Programa analítico de la asignatura los lineamientos de investigación en los cuales la asignatura este participando.

Lineamientos de Extensión de la cátedra

Para introducir a las y los estudiantes a las actividades de Extensión que realiza la cátedra. Se recomienda incorporar al Programa analítico de la asignatura los programas de Extensión en los cuales la asignatura este participando.

Actividades en las que pueden participar las y los estudiantes

Incluir todas aquellas instancias en las cuales las y los estudiantes puedan incorporarse como participantes activos tanto en proyectos de investigación como de extensión, en la asignatura o mediante el trabajo conjunto con otras asignaturas.

Eje: Investigación

Proyecto	Cronograma de actividades

Eje: Extensión

Proyecto	Cronograma de actividades



**UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL VILLA MARIA
CARRERA ACADÉMICA**