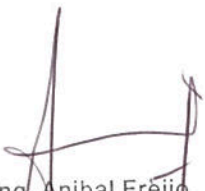

		FUNDAMENTOS DE INFORMÁTICA		P3.4.069
Departamento al que pertenece		Director		
Tecnología Informática		Ing. Anibal Freijo		
Carga horaria		Fecha de aprobación en el Consejo de Facultad y N° de Acta		
68 horas		22/05/2018 CF FAIN 366		
Carrera(s) en la que se dicta			Código(s) Carrera(s)	
Tecnicatura Universitaria En Desarrollo De Software			P1111	
Código(s) Correlativa(s) Precedente(s)	Código(s) Correlativa(s) Subsiguiente(s)		Código(s) Carrera(s)	
	P3.4.071		P1111	
Firmas				
Aprobación del Director de Departamento emisor.		Aprobación Decano(s)		
 Ing. Anibal Freijo Director (a/c) Dpto. de Tecnología Informática Fundación UADE		 Dr. RICARDO FABIÁN OROSCO Rector Fundación UADE		

I. Fundamentos de la materia

Las materias vinculadas a programación constituyen uno de los ejes temáticos dentro de la carrera. Informática en Ingeniería es la primera materia en esta línea, y en ella se presentan y desarrollan los temas básicos que servirán como fundamento y soporte conceptual para las siguientes materias del área.

II. Objetivos

Al finalizar el curso el alumno será capaz de:

- conocer con los conceptos y términos básicos que se utilizarán en el contexto de esta disciplina.
- reconocer la estructura de una computadora moderna, sus partes componentes y periféricos, y comprender sus principales funciones y los procesos que manejan
- entender y saber construir un programa
- expresar un programa en términos estructurados
- diferenciar conceptualmente los distintos tipos de estructuras de control
- plantear estrategias de resolución de problemas
- identificar módulos que puedan resolverse mediante procedimientos
- utilizar estructuras de datos básicas

Los objetivos de aprendizaje se encuentran desglosados en cada de las Unidades temáticas.

III. Contenidos

Contenidos mínimos

Descripción funcional de un computador. Concepto de programa y de dato. Sistemas operativos. Compiladores e intérpretes. Operaciones con datos. Concepto de algoritmo y sub-algoritmos. Estructuras de decisión. Estructuras iterativas. Arreglos. Operaciones con vectores y matrices.

Contenidos conceptuales

Unidad 1: Organización general de una computadora y un sistema.

Objetivos: familiarizarse con los conceptos y términos básicos que se utilizarán en el contexto de esta disciplina y de la programación en particular. Conocer los principios básicos de organización y funcionamiento interno de una computadora.

Contenidos conceptuales: Componentes de una computadora. Arquitectura básica de las computadoras. Organización de una máquina de Von Neumann. Principios de funcionamiento de una computadora.

Unidad 2: Programación Estructurada.

Objetivos: conocer la estructura de un programa simple, y familiarizarse con las estructuras de datos y algoritmos involucrados. Programación imperativa: datos y acciones.

Contenidos conceptuales: Estructura de un programa. Estructuras de datos y las acciones que las modifican: variables y los operadores relacionados: aritméticos y asignación, inicialización, actualización de datos. Noción de estados. Entrada / Salida.

Unidad 3: Variables simples, expresiones y tipos de dato.

Conocer y aprender a utilizar variables simples. Comprender la operación de asignación destructiva. Aprender a utilizar los tipos simples (predefinidos para el lenguaje en uso). Comprender el concepto de expresión y su tipo de dato asociado.

Contenidos conceptuales: Concepto de variable simple. Tipos de datos simples y sus operadores asociados. La asignación como modificador de estado de una variable. Concepto de expresión y su tipo de dato asociado. Entrada / Salida: Lectura y escritura.

Unidad 4: Estructuras de control (secuencia, selección, iteración).

Objetivos: conocer las estructuras de control de un programa. Desarrollar programas simples utilizando las tres estructuras de control básicas de la programación estructurada: secuencia, selección (if) e iteración (while-for). Mostrar la necesidad de incorporar contadores y acumuladores. Consolidar los conceptos de estados, algoritmo, programa y proceso. Rango.

Contenidos conceptuales: Estructuras de control clásicas: secuencia, selección, iteración. Condiciones como expresiones booleanas. Diferentes tipos de ciclos. Nociones de terminación de ciclos. Contadores y acumuladores.

Unidad 5: Estrategias para resolución de problemas.

Objetivos: comenzar a planificar una estrategia de resolución. Incorporar el hábito de leer y comprender el problema en primer lugar. Reconocer el estado inicial y final del programa. Empezar a incorporar el hábito de desagregar, aprender a descomponer un problema en subproblemas. Hacer la desagregación explícitamente. Introducir los algoritmos clásicos de búsqueda, obtener máximos y mínimos, sucesiones y secuencias.

Contenidos conceptuales: Comprensión global del problema. Estado inicial y estado final. Desagregación del problema. Noción de la descomposición modular. Resolución de los algoritmos clásicos de búsqueda, obtener máximos y mínimos, sucesiones y secuencias.

Unidad 6: Modularización: procedimientos, funciones, parámetros.

Objetivos: introducir el concepto de procedimientos. Utilizar los procedimientos como herramienta para la estructuración de un programa y para facilitar la reutilización de módulos. Presentar el concepto de parámetros y comprender los conceptos de parámetros formales y actuales, y las distintas formas de pasaje de parámetros. Utilizar parámetros como interfaz entre módulos de un programa. Comprender la noción de alcance de parámetros y utilizar este concepto adecuadamente. Comprender la conveniencia del uso de funciones y procedimientos. Utilizar funciones como operando de expresiones.

Contenidos conceptuales: Procedimientos: su utilización como herramienta para la modularización de programas y para la reutilización de módulos. Desagregación de un programa en términos de procedimientos. Enfoque top-down y bottom-up. Calidad de los programas en términos de: estructura, modularidad, legibilidad, mantenibilidad, reusabilidad, confiabilidad. Concepto de parámetros. Parámetros formales y actuales.

Unidad 7: Arreglos.

Objetivos: presentar el concepto de la estructura de datos "Arreglo". Aprender a utilizar los arreglos como tipos de datos estructurados. Comprender la noción de índice y componente de los arreglos. Aprender a programar los algoritmos clásicos en arreglos: búsqueda, inserción, eliminación, ordenamiento.

Contenidos conceptuales: tipos de datos estructurados vs. tipos de datos simples. Concepto de tipo de datos: su definición y utilización. Arreglos de una y más dimensiones. Índices y componentes. Arreglo como tipo de dato estático. Algoritmos de búsqueda, inserción, eliminación, y ordenamiento por selección, inserción y burbujeo.

Contenidos procedimentales

- Búsqueda y planificación de estrategias de resolución de problemas.
- Diseño de un programa estructurado.
- Individualización de datos y acciones.
- Identificación de procedimientos y funciones necesarias para la resolución de un problema.
- Reconocimiento e identificación de parámetros de entrada y salida.
- Identificar/analizar si un programa "está bien escrito".
-

Contenidos actitudinales

- Concientización sobre la importancia de adquirir un estilo de programación simple y práctico.
- Valoración de la sencillez, claridad y reusabilidad de un programa.
- Adquisición de una posición autocrítica sobre las soluciones planteadas.
- Desarrollo de programas en forma sistemática.
-

IV. Estrategias de enseñanza

Clases expositivas

Resolución de ejercicios en el pizarrón por parte del docente, fomentando:

- la participación de la clase
- la evaluación de soluciones alternativas, sopesando ventajas y desventajas de cada una.

Resolución individual y grupal de ejercicios con la supervisión del docente.

Resolución de ejercicios fuera del aula, para su posterior discusión en clase.

V. Recursos

Pizarrón y/o PC con cañón en aula normal.

Aulas de informática con cañón y PCs para los alumnos.

Software: Entorno de desarrollo para el lenguaje usado

VI. Modalidad de Evaluación y condiciones de aprobación

Para aprobar la asignatura el alumno deberá cumplir con la aprobación de las instancias de evaluaciones parciales y con un mínimo del 75% de asistencia a clase.

En el caso de materias con posibilidad de promoción, los alumnos que obtuviesen una calificación mínima de 7 (siete) puntos en cada una de las evaluaciones parciales (sin rendir recuperatorio) obtendrán la aprobación de la asignatura, cuya nota final consistirá en el promedio simple de las calificaciones de las evaluaciones parciales.

Los alumnos que, habiendo aprobado las evaluaciones parciales con un mínimo de 4 (cuatro) puntos o bien reprueben una de las dos instancias evaluatorias parciales pero aprueben luego el recuperatorio

correspondiente, no alcanzarán la promoción pero podrán rendir examen final de la asignatura en los 11 (once) turnos de exámenes finales consecutivos posteriores a la aprobación de la cursada. En este caso se consignará como nota final al promedio simple entre la nota de aprobación de la cursada (promedio de las calificaciones de las evaluaciones parciales aprobadas) y la calificación obtenida en el examen final regular.

En el caso que el alumno haya aprobado las instancias de evaluación y no requiera recuperar, podrá optar por rendir el examen final regular en la fecha prevista para el examen recuperatorio o bien en la fecha prevista para el examen final regular (una de las dos).

Los alumnos que rindan el examen final en la etapa de previos, la nota final a consignarse será exclusivamente la obtenida en dicha instancia de evaluación.

NORMAS DE SEGURIDAD: El trabajo en laboratorios y talleres debe llevarse a cabo respetando las normas de seguridad obligatorias. La aprobación de la cursada/materia estará sujeta al cumplimiento de las mismas, ya que son el principal factor de riesgo en las actividades de los alumnos, docentes, investigadores o técnicos. Utilizar siempre los Equipos de Protección Individual que se requiera (consultar procedimientos o protocolos de trabajo), por ejemplo protección ocular (anteojos gafas/pantallas faciales), guantes de vinilo y guardapolvo.

VII. Bibliografía

Básica

Joyanes Aguilar, Luis: Fundamentos de Programación. 4ta. Edición. Editorial Mc.Graw-Hill. ISBN 978-84-481-6111-8. EAN 9788448161118. Fecha de publicación: 24-03-2008

Complementaria

PILGRIN, Mark. Inmersión en Python 3 [consultas: 19/4/2018] Traducido por José Miguel González Aguilera. Disponible gratuitamente bajo licencia Creative Commons 3.0 en <http://www.jmgaguilera.com/libro/python/traduccion/C3/B3n/latex/2016/08/19/inmersion-python.html> y en <https://openlibra.com/es/book/inmersion-en-python-3>

MARZAL VARÓ, Andres; GRACIA LUENGO, Isabel; GARCÍA SEVILLA, Pedro. Introduccion a la programacion con Python 3 [En línea]. Disponible gratuitamente bajo licencia Creative Commons en <https://openlibra.com/es/book/introduccion-a-la-programacion-con-python-3>. [consultas: 19/4/2018]

Van Rossum, Guido. El tutorial de Python. Traducido y empaquetado por la comunidad de Python Argentina. [consultas: 19/4/2018] Español: docs.python.org.ar/tutorial/pdfs/TutorialPython3.pdf.

[En línea]. The Python Language Reference. [consulta: 19/4/2018] Español: <https://docs.python.org/3/reference/>

VIII. Cronograma

Clase	Actividad/contenido
1	Generalidades sobre la organización de una computadora. Conceptos básicos. Algoritmo y programa. Lenguajes de programación. Generaciones de lenguajes. Compiladores e intérpretes. Algoritmos elementales. División en partes y trabajo por niveles.
2	Estructuras básicas en el ámbito de la programación. Estructura secuencial. Constantes, variables y expresiones. Tipos de dato. Asignación. Operadores aritméticos. Orden de evaluación.
3	Estructuras de control: Estructura alternativa. Operadores relacionales. Operadores lógicos.
4	Estructuras de control: Estructura iterativa. Contadores y acumuladores.

5	Ejercitación combinando las tres estructuras básicas: Secuencia, ciclo y condicional.
6	Funciones. Valor de retorno. Uso de funciones como operando. Pasaje de parámetros. Ámbito de las variables.
7	Ejercitación. Simulacro.
8	1er Parcial
9	Resolución del examen parcial. Devolución y análisis de los errores más frecuentes.
10	Listas. Definición. Índice. Búsqueda de máximos y mínimos.
11	Ejercitación. Simulacro.
12	2do parcial
13	Aplicaciones de las listas. Inserción y eliminación de elementos.
14	Métodos de ordenamiento de listas: Selección, burbujeo e inserción
15	Ejercitación.
16	Recuperatorio y Final Regular
17	Recuperatorio