

Carrera INGENIERIA EN INFORMATICA		
Asignatura 3623 - Programación Inicial		
Trayecto Programación		
Año académico 2023		
Responsable / jefe de cátedra Dr. Pablo Vera		
Carga horaria semanal 4 Hs	Carga horaria total 64 Hs	Créditos -----
Modalidad: Presencial		
Correlativas anteriores -----	Correlativas posteriores PROGRAMACION ESTRUCTURADA BASICA- TALLER DE INTEGRACION	
Conocimientos necesarios -----		

Descripción de la asignatura

La asignatura Elementos de Programación forma parte del primer del grupo de Tecnologías Básicas, está planteada dentro del plan general de estudios como una materia básica. Tiene un objetivo general que es el de introducir al alumno en el difícil arte de la programación, lo cual significa aprender a resolver problemas, planteando situaciones originales, utilizando las habilidades y ventajas de un computador. El mismo se desarrolla a través de tres subobjetivos básicos perfectamente diferenciados y complementarios entre sí para el logro del objetivo.

- a) Aprender a interpretar, entender y encontrar los algoritmos necesarios para resolver los problemas propuestos, en distintos enunciados debidamente preparados para que el alumno, razone y adquiera nuevos hábitos de investigación para obtener los resultados correctos.
- b) Aprender a desarrollar lógicamente los procesos interpretados como correctos para resolución de problemas, con medios gráficos, como diagramas lógicos que permitan mostrar visualmente la solución a los problemas planteados. Verificar la lógica diseñada con lotes de prueba, asegurando el correcto funcionamiento de los problemas que pueden ser ejecutados a posteriori en un computador digital.
- c) Aprender a codificar, utilizando un adecuado lenguaje de programación como es el C, los problemas lógicamente desarrollados. Es este un lenguaje de aplicación en diversas áreas de la ingeniería.

Cumplidos estos subobjetivos, podemos afirmar que aquellos alumnos que lo aprueben estarán en condiciones de poder confeccionar un programa del tipo o similar al que pueda controlar la gestión administrativa por ejemplo la de un videoclub, un almacén, un depósito, etc., no será el óptimo, por la limitación de los recursos con que disponemos en esta etapa de la carrera, pero puede funcionar muy bien. Poseerá además una adecuada base para la comprensión de las nuevas tecnologías existentes en el mundo de la programación. Merece destacarse como otro objetivo adicional, durante el desarrollo de la materia, la búsqueda de la integración de la teoría con la práctica.

El alumno será inducido a la búsqueda bibliográfica, aprender a trabajar en equipo y saber documentar lo realizado.

Metodología de enseñanza

El alumno será inducido a: generar métodos de análisis para interpretar y resolver problemas propuestos, aprender a trabajar en equipo, saber evaluar las distintas alternativas algorítmicas para generar procesos que satisfagan problemáticas distintas, realizar una correcta presentación gráfica, a través de diagramas de lógicas que representen desarrollos y procesos determinados, además de su codificación en lenguaje C., con el uso de CodeBlock. Las clases tendrán básicamente contenido teórico / práctico. En la parte práctica se guiará a los alumnos en la resolución de situaciones que tienen como finalidad la fijación de conocimientos teóricos.

Los alumnos de la cátedra cuentan con la posibilidad de acceder, a través de la plataforma Miel a diferentes contenidos que permiten referenciar la totalidad de los temas propuestos. Contenidos Teóricos, Practicas, Videos, Mensajería, Foro, donde podrán interactuar con pares y formular, responder preguntas relacionadas con los contenidos de la asignatura. Los docentes intervendrán exclusivamente cuando alguna de las informaciones vertidas no se ajuste a la realidad o generen posibles confusiones. Además, cuenta con el apoyo de clases de Consultas distribuidas en diferentes horarios en forma presencial y virtual a través de Teams.

Objetivos de aprendizaje

- Reconocimiento de las herramientas disponibles en la actualidad para el proceso de sistematización.
- Razonamiento en forma lógica, para la resolución de problemas presentados al procesador.
- Uso de gráficos para representar la lógica del computador.
- Procesos secuenciales.
- Decisión Simple y Múltiple.
- Proceso repetitivos Fijo y Condicional.
- Programación en Módulos
- Vectores y Matrices

Contenidos mínimos

El papel de la computadora en la actualidad. Que es un Programa. Lenguajes de Programación. Etapas para solución de problemas. Diseño de estrategias. Algoritmos. Confección de Programas. Estructuras Básicas. Simbología utilizada para representar el proceso lógico de un programa. Selección Simple, Selección Múltiple, Proceso repetitivo Fijo, Proceso repetitivo Condicional. Resolución de Problemas. Lenguaje C. Generalidades. Elementos del Lenguaje. Codificación, programa fuente, Compilación, Objeto. Declaración de Variables, Distintas características, Tamaño según su contenido. Funciones de Entrada / Salida Sentencias de Asignación. Funciones de Librería. Estructura de Selección

Simple/Múltiple. Operadores Lógicos. Variables. Contador/Acumulador/Bandera. Máximos. Mínimos. Funciones. Prototipos. Desarrollo, Argumentos. Arreglos Unidimensionales. Búsquedas. Arreglos Bidimensionales. Uso de Matrices.

Competencias a desarrollar

Genéricas.

Identificación, formulación y resolución de problemas de ingeniería en sistemas de información/informática.

Aprendizaje continuo.

Específicas

Especificación, proyecto y desarrollo de sistemas de información.

Especificación, proyecto y desarrollo de software.

Programa analítico

Unidad 1	<p>a. Descripción general del principio de funcionamiento de un computador digital.</p> <p>b. Unidades componentes y sus características. Hardware y software.</p> <p>c. Ejemplos de aplicación y ejecución.</p> <p>e. <i>PRÁCTICA</i>: Desarrollo de un sencillo programa. Sistemas de numeración.</p>
Unidad 2	<p>a. Compresión de problemas. Enunciados e interpretación. Análisis de ideas.</p> <p>Relación entre datos, incógnitas y resultados</p> <p>b. Metodología a desarrollar para su resolución. Diseño de estrategias.</p> <p>c. Análisis de los diversos tipos de problemas. Algoritmos, concepto y definición. Características, limitaciones. Análisis del problema y diseño del algoritmo. Diseño descendente top-down.</p> <p>d. Definición y aplicación de los diagramas de lógica. Simbología.</p> <p>e. Teorema fundamental de la Programación Estructurada. Estructuras básicas.</p> <p>Combinación de las distintas estructuras. Precauciones.</p> <p>f. Programación modular. Técnicas de construcción y seguimiento de diagramas. Prueba y depuración de algoritmos. Juegos de prueba</p> <p>g. <i>PRÁCTICA</i>: Confección de algoritmos. Juegos de prueba. Análisis de resultados. Confeccionar los diagramas de lógica de sencillos problemas de aplicación utilizando las estructuras básicas. Juego de prueba y análisis de resultados.</p>
Unidad 3	<p>a. El lenguaje "C", breve historia y alcances. El entorno de la programación.</p> <p>b. Estructura general de un programa. El preprocesador, la función del main().</p>

	<p>c. Elementos básicos del lenguaje. Alfabeto. Palabras reservadas. Tipos de datos. Constantes, variables y operadores. Código ASCII.. Expresiones, diferentes tipos.</p> <p>d. Sentencias de asignación. Sintaxis, reglas y prioridades.</p> <p>e. Sentencias de entrada y salida con formato. Funciones printf(...) y scanf(..).</p> <p>f. a. Conceptos de contador y acumulador. Autoincremento. b. Funciones standard del "C" , diversos tipos. c. Combinación de operadores. d. Casos típicos de error en ciertas operaciones. Errores de almacenamiento. e. Sentencias de comentario. Estilo en la programación. f. <i>PRÁCTICA</i>: Confección de programas utilizando las funciones de biblioteca. Programación interactiva.</p>
<p>Unidad 4</p>	<p>a. Sentencia de selección simple, if(...). Sentencias anidadas. Operador condicional.</p> <p>b. Casos particulares en la combinación de las estructuras.</p> <p>c. Operadores lógicos. Prioridades y combinación. Reducción de los programas.</p> <p>d. Estructura de selección múltiple switch(...). Utilización de caracteres.</p> <p>e. Sentencia break., importancia de su utilización.</p> <p>f. Confección de programas. Juegos de prueba. Comprobación de resultados.</p> <p>g. <i>PRÁCTICA</i>: Confección de programas utilizando las sentencias mencionadas.</p> <p>Codificación. Confección de completos juegos de prueba. Pruebas de escritorio.</p> <p>Comprobación practica de los casos mencionados precedentemente.</p>
<p>Unidad 5</p>	<p>a. El concepto de ciclo. La sentencia de iteración definida for(...)</p> <p>b. Variantes en la utilización de los parámetros del for(...). Casos particulares.</p> <p>c. Ciclos anidados. Poder de iteración y sus variantes.</p> <p>d. Sentencia de iteración condicionada. Sentencias while(...) y do while(...). Comparación entre ambas. Ciclos 0-n y 1-n.</p> <p>e. Estructuras de iteración combinadas, sus ventajas.</p> <p>f. Sentencias "continue" . Sentencia "exit" .</p> <p>g. Graficación en la pantalla utilizando caracteres. Sentencias gotoxy(...) y windows(...). Otras funciones que facilitan la operatividad de las ventanas.</p> <p>h. <i>PRÁCTICA</i>: Confección de programas utilizando las sentencias de iteración. Prueba de programas con utilización práctica de ventanas. Facilitar el diálogo con los usuarios de los programas. Pruebas de aplicación directa.</p>
<p>Unidad 6</p>	<p>a. Los subprogramas. Las funciones en C. Forma general</p> <p>b. Declaración, prototipos y definición. Estructura y codificación.</p> <p>c. Variables globales y locales. Llamadas a una función.</p> <p>d. Declaración de parámetros. Parámetros formales y reales. Sentencia return.</p>

	<p>e. Llamada a funciones, argumentos. Parámetros por valor y por variable.</p> <p>f. Diagramas sintácticos.</p> <p>g. Los punteros, definición y utilización como parámetros de funciones.</p> <p>h. Reglas de ámbito. Variables externas. Algunas funciones standard</p> <p>Concepto de recursividad. Aplicaciones.</p> <p>i. <i>PRÁCTICA</i>: Confección de funciones con y sin parámetros. Inclusión de funciones pre programadas. Aplicación en el diseño y modularización de los programas. Usar funciones recursivas.</p>
Unidad 7	<p>a. Conceptos básicos sobre estructuras de datos tipo array. Declaración.</p> <p>b. Arrays unidimensionales, los vectores. Inicialización. Comprobación de límites.</p> <p>c. Arrays multidimensionales, tablas y matrices.</p> <p>d. Los arrays como parámetros de funciones.</p> <p>e. Operaciones de ordenamiento y búsqueda en arrays. Distintos métodos.</p>

DESARROLLO DE CONTENIDOS 1ER. CUATRIMESTRE - PROGRAMACION INICIAL						
					2023	1º Cuatrimestre
MES	Semana	Nro Clase	SEM-FECHA	TEMAS	Observaciones	
M	0	0	LU-27 SA-01	Introducción General/Programación de computadoras Resolución de Problemas/Estrategias /Algoritmos	Unidad 1 / Unidad 2	
A B R I L	1	1	LU-03 SA-08	Introducción General/Programación de computadoras Resolución de Problemas/Estrategias /Algoritmos	Unidad 1 / Unidad 2	
	2	2	LU-10 SA-15	Lenguajes de Programación estructurados/Compila Lenguaje C - Estructura Gral./Variables-Tipos	Unidad 3 / Unidad 4	
	3	3	LU-17 SA-22	Contador,Acumulador/Concatenación Practica de Ejercicios	Unidad 4	
	4	4	LU-24 SA-29	Estructura de Control de Selección múltiple Prac. Ejercicios	Unidad 4	
M A Y O	5	5	LU-01 SA-06	Estructura de Iteración Definida / Ejercicios FOR-1 - Max/Min Uso de Señales/Banderas-	Unidad 5	
	6	6	LU-08 SA-13	Estructura de Iteración Condicionada -Ejercicios While / Ejercicios -Do While / Estr.Combinadas-Ejercicios	Unidad 5	
	7	7	LU-15 SA-20	Practica Ejercicios Tipo Parcial	Practica Parcial primero	
	8	8	LU-22 SA-27	1er Parcial		
J U N I O	9	9	LU-29 SA-03	Programación Modular/Prototipo/parametros -Ejercicios	Unidad 6	
	10	10	LU-05 SA-10	Datos Arreglos / Búsqueda /Ordenamiento	Unidad 7	
	11	11	LU-12 SA-17	Datos Arreglos / Búsqueda /Matrices	Unidad 7	
	12	12	LU-19 SA-24	Practica Ejercicios Tipo Parcial	Practica segundo Parcial	
	13	13	LU-26 SA-01	2do Parcial		
J U L I O	14	14	LU-03 SA-08	Aclaraciones Recuperatorio		
	15	15	LU-10 SA-15	Recuperatorio de 1er. O 2 do. Parcial		
				Entrega de Notas		

Evaluación

Descripción del proceso evaluativo desarrollado por la cátedra

La evolución del proceso de enseñanza aprendizaje tiene carácter de permanente e integral y contempla la adquisición de conocimientos (aspecto este que se observa de manera específica en cuatro momentos durante el ciclo lectivo), la formación de actitudes, el desarrollo de capacidades de análisis, destrezas y habilidades para encontrar información y resolver situaciones que se le presentan.

CONDICIONES DE CURSADA Y APROBACIÓN

En la evaluación integral del alumno detallado anteriormente, se suma como registro de sus saberes, la propuesta de dos evaluaciones escritas donde se presentará una problemática a resolver. En estas se deberá desarrollar una lógica en diagrama y posterior codificación, que indique el proceso para resolver el problema. Estas problemáticas permiten la utilización de recursos divididos en puntos a resolver. La correcta resolución de cada uno de los puntos permitirá obtener la nota correspondiente. Utilización de gráficos correctos. Codificación en lenguaje que represente la lógica presentada.

1. Aquel alumno que obtenga **2 (dos) calificaciones menores a 4 (cuatro) puntos**, consecutivas o alternadas, pierde su condición de regular debiendo recurrir a la asignatura. En este caso la nota que se consignara en la planilla de calificaciones para el ciclo lectivo es **ausente**.
2. Debido a la modularidad establecido para las evaluaciones parciales, los días en que se rinden éstas, (tiempo aproximado 120 minutos) no habrá suspensión de actividades. Con posterioridad a la realización del parcial, los docentes explicaran y / o desarrollaran según corresponda cada uno de los temas evaluados. El resto de la clase será dedicado a actividades prácticas o teóricas según necesidad. Para el caso de recuperatorios, podrán desarrollarse en un día habitual de cursada. La instancia de recuperación tendrá lugar la última semana de curso normal (se pueden rendir hasta **un parcial** durante el horario normal de clase y manteniendo los límites de tiempo preestablecidos para cada parcial).
3. La condición del alumno con respecto a la asignatura deberá definirse como máximo al concluir el periodo de cursada que nos ocupa
4. Para los recuperatorios, la nota obtenida en el parcial que se recupera reemplaza a la original, registrando como calificación del parcial, el nuevo valor obtenido en el recuperatorio.
5. La aprobación de la materia (**aprobó**) se obtendrá si se cumplen los siguientes requisitos:
 - Aprobar la totalidad de exámenes parciales (directamente o a través de recuperatorio) con una nota igual o superior a 7 (siete) puntos.
 - Asistir como mínimo al 75 % de las clases.
6. La condición de alumno regular (**cursada**, habilita para rendir examen final) se obtendrá si se cumplen los siguientes requisitos:
 - Aprobar la totalidad de exámenes parciales (directamente o a través de recuperatorio) con una nota igual o superior a 4 (cuatro) puntos.
 - Asistir como mínimo al 75 % de las clases.

7. Quienes no cumplan con el requisito de asistencia (presencia igual o superior al 75 %) perderán su condición de alumno regular y merecerán la condición de **ausente**.
8. Aquellos alumnos que habiendo satisfecho el requerimiento de asistencia, no logren la aprobación o regularidad merecerán la condición de **reprobados** o **ausentes**. La condición de **reprobado** se consigna cuando se haya presentado en la instancia de recuperación y no satisfizo los objetivos de aprobación o Para el caso que no se haya presentado a la instancia de recuperación teniendo necesidad de hacerlo, la condición a consignar es **ausente**. -
9. **Exámenes libres.** Aquellos alumnos que consideren tener los conocimientos suficientes de los temas que se imparten podrán optar por este tipo de examen. La evaluación con esta característica será del mismo nivel de complejidad que las correspondientes al examen regular, aunque más extensas. El alumno comenzará con una evaluación teórica y al concluir la misma continuará con una evaluación práctica. Para lograr la aprobación de la asignatura deberá satisfacer ambas instancias. Este tipo de examen tendrá lugar exclusivamente en la primera fecha de cada llamado a examen final.

Primera evaluación	Semana 8	Examen Escrito	2 horas
Segunda evaluación	Semana 13	Examen Escrito	2 horas
Recuperatorio	Semana 15	Examen Escrito	2 horas

Bibliografía obligatoria

Titulo	Autor	Editorial	Edición	Año
Freijedo/ Cortagerena	Tecnologías de la Información y las comunicaciones.	Ediciones Macchi	2000	1ra Edición
Byron/ S. Gottfried	Programación en C	McGraw-Hill	2015	2da.Edición

Bibliografía complementaria recomendada

Titulo	Autor	Editorial	Edición	Año
B. Kernighan/ D. Ritchie	EL LENGUAJE DE PROGR. C ++	Prentice Hall.	2010	2da.Edición

Otros recursos obligatorios (videos, enlaces, otros) Incluir una fila por cada recurso

Nombre

Otros recursos complementarios (videos, enlaces, otros) Incluir una fila por cada recurso

Nombre