UNIVERSIDAD DEL AZUAY

FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN
ESCUELA INGENIERIA DE SISTEMAS Y TELEMATICA

1. Datos generales

Materia: INTRODUCCIÓN A LA PROGRAMACIÓN
Código: FAD0176
Paralelo:
Periodo : Septiembre-2017 a Febrero-2018
Profesor: ESTRELLA MALDONADO RENE GUSTAVO
Correo electrónico rgestrella@uazuay.edu.ec

Docencia	Práctico	Autónomo:		Total horas
		Sistemas de tutorías	Autónomo	
8				8

Prerrequisitos:

Ninguno

2. Descripción y objetivos de la materia

Introducción a la Programación es una materia fundamental en la carrera de Ingeniería de Sistemas y Telemática porque constituye uno de los ejes de formación del Ingeniero de Sistemas y Telemática. El entendimiento y análisis de problemas, de diversa índole, le permitirá representar mediante algoritmos la solución del problema. Podrá posteriormente elegir y aplicar un lenguaje de programación para construir un programa que encuentre la solución de un problema automáticamente.

Esta materia trata sobre los fundamentos de programación, se capacita a los estudiantes que inician la carrera, en el análisis de un problema bajo estudio y se enseñan las técnicas y metodologías para la elaboración de algoritmos. Se proporcionan los fundamentos sobre programación modular y programación estructurada, y se enseñan las técnicas de programación y el uso de herramientas como diagramas de flujo, diagramas N-S y pseudocódigo. Como apoyo del aprendizaje de las técnicas de programación, se utiliza el lenguaje de programación Borland C++ para la construcción de programas.

Introducción a la Programación sienta las bases para el eje de formación de lenguajes de programación y contribuye en forma transversal con otras materias como Estructuras de Datos, Bases de Datos y Aplicaciones de Bases de Datos, Inteligencia Artificial.

3. Contenidos

1	**Introducción a los lenguajes de programación**
1.1	Indicaciones generales, presentación del Sílabo. Servicios de la página Web de la universidad: WebMail y Bibliotecas Virtuales (2 horas)
1.2	Software: Conceptos, Software del sistema y software de aplicaciones. Los lenguajes de programación: concepto, lenguaje de máquina, ensamblador y lenguaje de alto nivel (1 horas)
1.3	Editor de programas, compiladores, traductores y depurador de programas (1 horas)
2	**Herramientas de programación**
2.1	Fases para resolver un problema (4 horas)
2.2	Algoritmos: Concepto y características. Diagramas de flujo y pseudo-código como herramientas para construir algoritmos. (2 horas)
2.3	Conceptos de la programación estructurada y modular (2 horas)
2.4	Ejercicios de aplicación (8 horas)
3	**Estructura general de un programa**
3.1	Estructura general de un programa (1 horas)
3.2	Datos y tipos de datos (1 horas)
3.3	Constantes y variables (1 horas)
3.4	Operadores básicos, jerarquía de operadores (2 horas)
3.5	Expresiones (1 horas)
3.6	Evaluación de expresiones y tipos de errores (4 horas)
4	**Estructuras de control**
4.1	Secuenciales (1 horas)
4.2	Selectivas: binarias, compuestas, anidadas y de selección múltiple utilizando valores ordinales. (3 horas)

4.3	Repetitivas: condicionales y automática (4 horas)
4.4	Anidamiento de estructuras de control (2 horas)
4.5	Ejercicios de aplicación (8 horas)
4.6	Introducción al lenguaje de programación C++: Estructura gneral de un programa, tipos de datos, Contantes y variables, Sentencias de entrada y salida. (2 horas)
4.7	Ejercicios de aplicación en C++ (6 horas)
5	**Subalgoritmos**
5.1	Introducción a los subalgoritmos o subprogramas (1 horas)
5.2	Procedimientos y funciones (1 horas)
5.3	Paso de parámetros por valor y por nombre o dirección (2 horas)
5.4	Ámbito de las variables: globales y locales (1 horas)
5.5	Ámbito de los procedimientos y funciones (1 horas)
5.6	Ejercicios de aplicación en C++ (6 horas)
6	**Arreglos**
6.1	Introducción a las estructuras de datos (1 horas)
6.2	Arreglos de una dimensión (vectores) (1 horas)
6.3	Operaciones con vectores: ordenamientos y búsqueda (2 horas)
6.4	Arreglos de dos dimensiones (matrices) (2 horas)
6.5	Arreglos multidimensionales (2 horas)
6.6	Ejercicios de aplicación en C++ (6 horas)
7	**Registros**
7.1	Introducción (2 horas)
7.2	Arreglos Vs. Registros (2 horas)
7.3	Arreglos de registros (2 horas)
7.4	Arreglos como campos de registros (4 horas)
7.5	Ejercicios de aplicación en C++ (6 horas)
8	**Cadenas de caracteres**
8.1	Fundamentos de las cadenas de caracteres (2 horas)
8.2	Operaciones con cadenas (3 horas)
8.3	Funciones y procedimientos para el manejo de cadenas (3 horas)
8.4	Ejercicios de aplicación en C++ (4 horas)
9	**Archivos**
9.1	Nociones sobre archivos: archivos físicos y archivos lógicos. (1 horas)
9.2	Organización de archivos y modos de acceso (0 horas)
9.2.1	Organización secuencial (1 horas)
9.2.2	Organización directa (2 horas)
9.2.3	Organización secuencial indexada (2 horas)
9.3	Operaciones sobre los archivos (2 horas)
9.4	Mantenimiento de Archivos (4 horas)
9.5	Ejercicios sobre lectura y escritura de ficheros. Funciones para el manejo de ficheros en C++ (6 horas)

4. Sistema de Evaluación

Resultado de aprendizaje de la carrera relacionados con la materia

Resultado de aprendizaje de la materia	Evidencias
ap. Desarrolla la lógica algoritmica en el análisis y resolución de problemas aplicando los fundamentos de la programación.	
-Analizar el problema bajo estudio y plantear una solución precisa.	-Resolución de ejercicios, casos y otros
-Determinar la factibilidad de resolver un problema mediante un programa de cómputo.	-Resolución de ejercicios, casos y otros
-Diseñar la solución de problema aplicando técnicas de programación apropiadas.	-Proyectos
-Operar los programas utilitarios para la edición, compilación y depuración de	-Resolución de

Resultado de aprendizaje de la carrera relacionados con la materia

Resultado de aprendizaje de la materia	Evidencias
programas.	ejercicios, casos y otros
-Utilizar un lenguaje de programación apropiado para escribir programas.	-Resolución de ejercicios, casos y otros

Desglose de evaluación

Evidencia	Descripción	Contenidos sílabo a evaluar	Aporte	Calificación	Semana
Resolución de ejercicios, casos y otros	Planteamiento de algoritmos	Estructura general de un programa, Herramientas de programación, Introducción a los lenguajes de programación	APORTE 1	10	Semana: 4 (16/10/17 al 21/10/17)
Resolución de ejercicios, casos y otros	Ejercicios usando estructuras de control	Estructuras de control	APORTE 2	5	Semana: 7 (06/11/17 al 11/11/17)
Resolución de ejercicios, casos y otros	Uso de funciones en C++	Subalgoritmos	APORTE 2	5	Semana: 10 (27/11/17 al 02/12/17)
Resolución de ejercicios, casos y otros	Uso de funciones, arreglos y clases	Arreglos, Registros	APORTE 3	5	Semana: 15 (02/01/18 al 06/01/18)
Proyectos	Proyecto integrador	Archivos, Arreglos, Cadenas de caracteres, Estructuras de control, Registros, Subalgoritmos	APORTE 3	5	Semana: 16 (08/01/18 al 13/01/18)
Resolución de ejercicios, casos y otros	Examen final	Archivos, Arreglos, Cadenas de caracteres, Estructuras de control, Registros, Subalgoritmos	EXAMEN	20	Semana: 17-18 (14-01-2018 al 27-01-2018)
Resolución de ejercicios, casos y otros	Examen supletorio	Archivos, Arreglos, Cadenas de caracteres, Estructuras de control, Registros, Subalgoritmos	SUPLETORIO	20	Semana: 19-20 (28-01-2018 al 03-02-2018)

Metodología

Para el desarrollo de los aspectos teóricos de la materia se desarrollarán clases magistrales con el apoyo de diapositivas y las computadoras de los laboratorios de la universidad. Se realizarán trabajos prácticos en clase sobre los temas tratados y trabajos prácticos autónomos en grupos fuera del aula.

Criterios de Evaluación

- En los trabajos prácticos dentro del aula se desarrollarán los temas explicados en las clases, estos trabajos no serán calificados.
- Los trabajos autónomos se desarrollarán de manera individual o en grupos de dos estudiantes y versarán sobre los temas tratados en clases. Se elaborará un informe del trabajo realizado y se evaluará la creatividad en la elaboración de la solución de los problemas planteados y la aplicación de los conocimientos adquiridos hasta la fecha.
- Los trabajos autónomos se calificarán sobre 2 puntos y se promediarán.
- Las pruebas se realizarán en las fechas indicadas y tratarán sobre los temas estudiados hasta la última clase anterior a la fecha de cada prueba.
- Se controlará el cumplimiento de las fechas para la presentación de trabajos y se sancionará si se evidencian trabajos similares o copiados.
- En los trabajos en los que aplica, se calificará la redacción y ortografía, con un límite del 20% de la nota.
- Los exámenes y trabajos se calificarán con décimas.

5. Referencias

Bibliografía base

Libros

Autor	Editorial	Título	Año	ISBN
GRANIZO MONTALVO, EVELIO	Editorial ESPE	PROGRAMACIÓN ESTRUCTURADA EN PSEUDO-CÓDIGOS	2000	9978-41-280-8
JOYANES AGUILAR, LUIS	McGraw-Hill	FUNDAMENTOS DE PROGRAMACIÓN: Algoritmos, estructura de datos y objetos	2008	978-84-481-6111-8

Web

Software

Autor	Título	URL	Versión
Borland International Inc.	Borland C++	Laboratorios UDA	5.02
Borland International Inc.	Borland C++	Laboratorios UDA	Versión 5.02

Bibliografía de apoyo

Libros

Autor	Editorial	Título	Año	ISBN
Bjarne Stroustrup	Addison Wesley	El Lenguaje de Programación C++	2013	978-0321563842
Paul Deitel y Harvey Deitel	Deitel	Cómo Programar en C++	2016	978-0134448237

Web

Software

Docente

Director/Junta

Fecha aprobación: 22/09/2017

Estado: Aprobado