

UNIVERSIDAD MILITAR NUEVA GRANADA



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 1 de 10

NOMBRE DEL PROGRAMA: INGENIERÍA MECATRÓNICA	
NOMBRE DE LA ASIGNATURA	Programación II (Objetos)
CÓDIGO	1800101
SEMESTRE	2
PRERREQUISITOS	Programación I
CORREQUISITOS	
COORDINADOR Y/O JEFE DE ÁREA	Ing. Liliana Quintero Lozano
DOCENTE (S)	Ing. Luís Eduardo Pérez Peregrino, Ing. Juan Camilo Gutiérrez Pinzón, Ing. Juan Carlos Niño Maldonado
CRÉDITOS ACADÉMICOS	3
FECHA DE ELABORACIÓN/ ACTUALIZACIÓN	26/11/2020

JUSTIFICACIÓN

La programación orientada a objetos se hace pertinente para ampliar la visión de los estudiantes como programadores con conceptos de mayor dimensión y abstracción, los cuales le permitan resolver problemas de un grado de complejidad más alto y cercano a la realidad.

Con esta orientación se fortalece el pensamiento lógico y creativo. Un estudiante adquiere habilidades complejas para programar y para abordar problemas de forma integral; los estudiantes producen y optimizan el software en cualquier área, como la computación gráfica, el procesamiento de imágenes, la simulación, la realidad virtual, la animación entre otras.

OBJETIVO GENERAL

Solucionar problemas de sistematización de información a través de la programación modular, usando estructuras de datos incursionando en aplicaciones de la Programación Orientada a Objetos para observar sus ventajas en el desarrollo de software.

COMPETENCIA GLOBAL

Al terminar el curso el estudiante estará en la capacidad de diseñar soluciones a problemas complejos empleando los siguientes conceptos y herramientas así:

- Programación modular.
- Almacenamiento de información en Archivos.
- Creación de estructuras de datos cuyos componentes pueden ser de tipos diferentes de datos
- Creación de los programas para modelar los objetos del mundo real de un modo más eficiente, a través de la Programación Orientada a Objetos (POO).



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 2 de 10

COMPETENCIAS ESPECÍFICAS

1. Ocupacionales y Profesionales

- Desarrolla un aplicativo para dar solución a un problema relacionado con la vida real utilizando programación estructurada y programación orientada a objetos
- Reconoce y relaciona información mediante el uso de tipos de datos abstractos
- Implementa un programa aprovechando las ventajas del concepto de funciones

2. Cognitivas

- Identifica los diferentes tipos de arreglos: unidimensionales y bidimensionales.
- Desarrolla algoritmos utilizando funciones de paso valor y paso por referencia y los codifica en el lenguaje de programación seleccionado.
- Aplica los conceptos de la aritmética de punteros en el desarrollo de algoritmos.
- Utiliza los conceptos de recursividad, para la solución de problemas específicos.
- Comprende los conceptos de Tipos de Datos Abstractos (TDA).
- Resuelve algoritmos utilizando la creación e inserción de datos, archivos.
- Comprende los conceptos fundamentales de la Programación Orientada a Objetos.
- Aplica los conceptos de Encapsulamiento en diferentes contextos o problemáticas planteadas dentro de la programación orientada a objetos.
- Identifica los diferentes tipos de clase: Clase base y clase derivada.
- Aplica el concepto de herencia en la programación orientada a objetos y lo aplica en el desarrollo de algoritmos.
- Entiende el concepto de Polimorfismo en la programación orientada a objetos y lo utiliza en el desarrollo de algoritmos.

3. Comunicativas

- Propone alternativas de solución a un problema
- Implementa aplicaciones como expresión de una idea
- Indica las instrucciones básicas para la elaboración de un programa
- Expresa inquietudes con respecto a aspectos semánticos y sintácticos del lenguaje

4. Investigativas

- Detecta y corrige los errores de lógica y sintaxis obtenidos al depurar un programa
- Formula, argumenta y da soluciones para la creación de un aplicativo orientado a objetos
- Manifiesta interés en aprender de forma autónoma
- Desarrolla habilidades para buscar, procesar y analizar información procedente de diferentes fuentes

CONTENIDO

Unidad Modular 1. Recursividad

Temas

- Funciones de paso por valor
- Funciones de paso por referencia
- Arreglos unidimensionales
- Arreglos bidimensionales



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 3 de 10

- Arreglos como parámetros de funciones.
- Punteros.
- Recursividad.

Unidad Modular 2. Tipos de Datos Abstractos (TDA)

Temas

- Concepto de tipo abstracto
- Implementación de un TDA.
- Operaciones de inserción y extracción de información.
- Tipos abstractos de datos como parámetros de funciones.
- Uso de archivos de datos.
- Cadenas de caracteres

Unidad Modular 3. Programación Orientada a Objetos

Temas

- Definición y características de la POO
- Modelado e identificación de objetos
- Declaración de una clase
- Declaración de objetos de clases
- Declaración de métodos
- Constructores destructores
- Propiedades fundamentales de POO
 - ✓ Herencia
 - Simple
 - Múltiple
 - ✓ Polimorfismo
 - ✓ Encapsulamiento

Semana	Tema o actividad presencial	Actividades de trabajo independiente
1 (25 Ene - 30 Ene)	<i>Presentación de Programa y revisión de conocimientos previos. Concepto de arreglo y tipos de arreglos: unidimensionales y bidimensionales.</i>	<i>Revisión de los conceptos necesarios para dar inicio al programa. Presentación por parte del docente. Lectura en el aula virtual: Manual de Arreglos. Ejercicios de aplicación.</i>
2 (01 Feb - 06 Feb)	<i>Concepto de función Concepto de arreglo y función, concepto de variable y paso por referencia.</i>	<i>Taller de Arreglos con funciones (Aula Virtual) Ejercicios de aplicación.</i>
	<i>Sintaxis, punteros y arreglos, aritmética de</i>	<i>Lectura en el aula virtual: Manual sobre</i>

El uso no autorizado así como la reproducción total o parcial de su contenido por cualquier persona o entidad, estará en contra de los derechos de autor.

UNIVERSIDAD MILITAR NUEVA GRANADA



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 4 de 10

3 (08 Feb - 13 Feb)	<i>punteros. Concepto de recursividad.</i>	<i>funciones página 20 Recursividad. Exposición del tema y ejemplos de aplicación.</i>
4 (15 Feb - 20 Feb)	<i>Cadenas. Definición y Uso de los TDA en programación.</i>	<i>Lectura "Cadena de Caracteres" (págs. 121-126) en el aula virtual. Guía Práctica de laboratorio No. 2. Taller Tipos de datos abstractos.</i>
5 (22 Feb - 27 Feb)	Primera Evaluación Parcial	
	<i>TDA como arreglos de una y dos dimensiones.</i>	<i>Lectura "Registros y archivos" (págs. 112-113) en el aula virtual. Guía Práctica de laboratorio No. 2.</i>
6 (01 Mar - 06 Mar)	<i>Creación e inserción de datos, Archivos</i>	<i>Lectura "Registros y archivos" (págs. 115-120) en el aula virtual. Guía Práctica de laboratorio No. 2. Guía Práctica de laboratorio No. 2.</i>
7 (08 Mar - 13 Mar)	<i>Conceptos fundamentales de POO: concepto de clase, visibilidad de miembros, datos miembros y funciones miembros, objeto, encapsulamiento.</i>	<i>Lectura "Manual de Programación 2" (págs. 127-132) del Aula Virtual, Guía de laboratorio No. 3.</i>
8 (15 Mar - 20 Mar)	<i>Modelado e identificación de objetos, declaración de clases, métodos.</i>	<i>Ejercicios de aplicación.</i>
9 (22 Mar - 27 Mar)	<i>Conceptos de objetos, C++. Concepto de función, atributo y clase.</i>	<i>Ejercicios de aplicación.</i>
(29 Mar - 04 Abr)	Semana Santa	
10 (05 Abr - 10 Abr)	<i>Concepto de Constructor, Destrucción de una Clase.</i>	<i>Ejercicios de aplicación Lectura "Manual de Programación 2" (págs. 142-147) del Aula Virtual. Ejercicios de aplicación.</i>
11 (12 Abr - 17 Abr)	Segunda Evaluación Parcial	
	<i>Constructores y destructores, Clases</i>	<i>Ejercicios de aplicación</i>
12 (19 Abr - 24 Abr)	<i>Encapsulamiento</i>	<i>Ejercicios de aplicación</i>
13 (26 Abr - 01 May)	<i>Clase base y derivadas</i>	<i>Lectura "Manual de Programación 2" (págs. 160-165) del Aula Virtual</i>

El uso no autorizado así como la reproducción total o parcial de su contenido por cualquier persona o entidad, estará en contra de los derechos de autor.

UNIVERSIDAD MILITAR NUEVA GRANADA



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 5 de 10

14 (03 May -08 May)	Herencia	<i>Ejercicios de aplicación. Guía Práctica de laboratorio No.4. Lectura "Manual de Programación 2" (págs. 155-165) del Aula Virtual. Taller de Herencia (Aula Virtual) Ejemplos de aplicación</i>
15 (10 May -15 May)	Polimorfismo	<i>Guía Práctica de laboratorio No.5. Lectura "Manual de Programación 2" (págs. 96-98) del Aula Virtual. Taller polimorfismo (Aula Virtual) Lectura "Manual de Programación 2" (págs. 160-165) del Aula Virtual.</i>
16 (17 May - 22May)	Polimorfismo	<i>Elaboración de los talleres propuestos al tema.</i>
17 (24 May -29May)	Examen Final	
18 (31 May – 5 Jun)	Socialización de Notas corte 3	

SISTEMA DE EVALUACIÓN

Acorde con la metodología y los porcentajes definidos por la Universidad para cada uno de los cortes del semestre, se realiza la evaluación del curso entre 0.0 y 5.0. La calificación del curso será la que resulte de la ponderación de cada corte y será aprobado si cumple los requisitos del reglamento estudiantil.

Se presentarán dos notas parciales cada una con un valor de 30% y un examen final correspondiente a un 40%. En cada corte se realizarán quices, talleres o entrega de tareas correspondientes a un 40% de la nota parcial.

Para la evaluación de la asignatura se tendrán en cuenta los criterios que se muestran en la siguiente tabla:

UNIVERSIDAD MILITAR NUEVA GRANADA



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 6 de 10

Actividad	Primer Corte (30%)		Segundo Corte (30%)		Tercer Corte (40%)	
	Cantidad	Valor	Cantidad	Valor	Cantidad	Valor
Quices (Individual)	2	25%	2	25%	1	10%
Talleres (Por parejas o individuales)	2	25%	2	25%	1	10%
			Trabajo de Aplicación - Opcional Sustentación - Procedimental -Actitudinal (La sustentación de un trabajo se considera como un quiz)			30%
Examen parcial	1	50%	1	50%	1	50%
Total	100%		100%		100%	

* Realizar mínimo tres actividades

Dimensión Escala	EXCELENTE 5	BUENO	ACEPTABLE	DEFICIENTE 0
CORTE I				
Desarrolla un programa para dar solución a un problema utilizando estructuras de datos de una y dos dimensiones empleando funciones de usuario de paso por valor y referencia, además, utiliza los conceptos de aritmética de punteros y recursividad.	Desarrolla correctamente la solución a un problema aplicando estructuras de datos de una y dos dimensiones y emplea adecuadamente funciones de usuario de paso por valor y referencia en un lenguaje de programación	Desarrolla con algunos errores la solución a un problema aplicando estructuras de datos de una y dos dimensiones y emplea adecuadamente funciones de usuario de paso por valor y referencia en un lenguaje de programación	Desarrolla con algunos errores la solución a un problema aplicando estructuras de datos de una y dos dimensiones y emplea adecuadamente funciones de usuario de paso por valor y referencia en un lenguaje de programación	Ningún planteamiento para la solución del problema
CORTE II				
Desarrolla un programa en un lenguaje de programación para dar solución a un problema utilizando	Desarrolla e implementa eficazmente el paradigma de los TDA, en todo tipo de soluciones	Desarrolla e implementa algunos conceptos de los TDA, en soluciones informáticas con funciones, arreglos y	Desarrolla e implementa pocos conceptos de los TDA, en soluciones informáticas con funciones, arreglos	Ningún planteamiento para la solución del problema.

El uso no autorizado así como la reproducción total o parcial de su contenido por cualquier persona o entidad, estará en contra de los derechos de autor.

UNIVERSIDAD MILITAR NUEVA GRANADA



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 7 de 10

correctamente Tipos de Datos Abstractos (TDA) y archivos.	informáticas con funciones, arreglos y archivos.	archivos.	y archivos.	
CORTE III				
Aplica los conceptos de Programación Orientada a Objetos, empleando los conceptos de instanciamiento, encapsulamiento, constructores, clases, clase base y clase derivada, herencia y polimorfismo en diferentes contextos o problemáticas planteadas.	Domina y aplica los conceptos de Programación Orientada a Objetos, empleando instanciamiento, encapsulamiento, constructores, clases, clase base y clase derivada, herencia y polimorfismo en diferentes contextos o problemáticas planteadas.	Aplica algunos de los conceptos de Programación Orientada a Objetos, empleando instanciamiento, encapsulamiento, constructores, clases, clase base y clase derivada, herencia y polimorfismo en diferentes contextos o problemáticas planteadas.	Utiliza pocos de los conceptos de Programación Orientada a Objetos, empleando instanciamiento, encapsulamiento, constructores, clases, clase base y clase derivada, herencia y polimorfismo en diferentes contextos o problemáticas planteadas.	Ningún planteamiento para la solución del problema.

BIBLIOGRAFÍA

Textos Guía:

1. Deitel, P. 2009, C++ Como Programar 6ª ed. Pearson Education.
2. Deitel, H. 2008, Cómo programar en C++. 6a ed. Pearson Education. México.
3. Reyes, S Joaquín 2018, Programación en C++ 3ª ed. ITCampus Academy
4. Joyanes, L. 2008, Fundamentos de Programación. 4 ed. McGraw-Hill.
5. Josep Anton Pérez López, Lluís Ribas i Xirgo. Introduction to Software Development.. Free Technology Academy. Topics development, computer-science, computer science, software (2010).

Textos Complementarios:

1. Paul J. Deitel, Harvey M. Deitel 2007, C how to Program 5ª Edition. Pearson Prentice Hall.
2. Dale, N. y Weems, C. 2008, Programación y resolución de problemas con C++. 4a. Ed. McGraw-Hill.
3. Joyanes, L. 2008, Fundamentos de Programación. Algoritmos, estructura y Objetos. 4a. Ed. McGraw-Hill.
4. Bjarne Stroustrup 2014, Programming: Principles and Practice Using C++ Second Edition. Addison-Wesley

Textos Electrónicos:

El uso no autorizado así como la reproducción total o parcial de su contenido por cualquier persona o entidad, estará en contra de los derechos de autor.



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 8 de 10

1. IEEE. <http://ieeexplore.ieee.org/Xplore/dynhome.jsp>.
2. Cómo programar en C++, Harvey M. Deitel, No. Edición: 6, Año: 2008, ISBN e-Book: 9786074424621, ISBN: 9789702612735.
3. Programación y resolución de problemas con C Autor: Dale, Editorial: Mcgraw-Hill, Interamericana EDITORES, ISBN: 9786071500410

MATERIAL COMPLEMENTARIO DE APRENDIZAJE PARA ESTUDIANTES

Glosario:

- Tipo de datos, condicionales, ciclos, vectores, funciones, arreglos, datos abstractos, cadenas, apuntadores, recursividad, clase, objetos, constructores, destructores, atributos, instanciación, herencia, encapsulamiento, polimorfismo.

Preguntas de repaso:

1. Cómo se clasifican las estructuras selectivas y cuál es su diferencia.
2. Cuáles son las estructuras cíclicas que se trabajan en programación.
3. Cuáles son los dos tipos de arreglos se trabajamos en programación y cuál es su diferencia.
- 4.Cuál es el objetivo de una función.

Material Multimedia:

1. Presentaciones de los temas a trabajar.

Enlaces en la Red:

1. www.abcdatos.com/tutoriales/tutorial/l257.html
2. <http://www.lawebdelprogramador.com>
3. https://www.goodreads.com/book/show/1647345.C_for_C_Programmers
4. <http://www.abcdatos.com/tutoriales/tutorial/l257.html>
5. [http://es.wikipedia.org/wiki/C_\(lenguaje_de_programación\)](http://es.wikipedia.org/wiki/C_(lenguaje_de_programación))
6. <http://www.monografias.com/trabajos34/programas-visual-c/programas-visual-c.shtml>
7. http://www.youtube.com/watch?v=ofemmXa_M5o&feature=relmfu
8. <http://www.zator.com/Cpp/>

Curso Virtual:

<http://virtual2.umng.edu.co/>

UNIVERSIDAD MILITAR NUEVA GRANADA



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 9 de 10

COMPETENCIA DEL DOCENTE

Docentes de Cátedra u Ocasional:

Ingeniero de sistemas o ingeniero en computación o ingeniero informático o ingeniero de software, con capacidad de análisis, modelamiento, desarrollo y prueba de software estructurado y orientado a objetos.

Experiencia docente mínimo de un (1) año como docente de tiempo completo en instituciones de educación superior o su equivalente en medio tiempo u hora de cátedra.

Docentes Públicos de Carrera:

Ingeniero de sistemas o ingeniero en computación o ingeniero informático o ingeniero de software, con estudios de maestría o doctorado en sistemas TIC o software o informática.

Experiencia docente mínimo de un (1) año como docente de tiempo completo en instituciones de educación superior o su equivalente en medio tiempo u hora de cátedra.

Docentes Públicos de Carrera:

Ingeniero de sistemas o ingeniero en computación o ingeniero informático o ingeniero de software, con estudios de maestría o doctorado en sistemas TIC o software o informática.

Experiencia docente mínimo de un (1) año como docente de tiempo completo en instituciones de educación superior o su equivalente en medio tiempo u hora de cátedra.

CONTROL DE CAMBIOS

CAMBIO REALIZADO	JUSTIFICACIÓN DEL CAMBIO	ACTA DE APROBACIÓN
Actualización del Contenido	Por cambio de formato	Acta N°06 del Comité de Currículo y Autoevaluación de la FCCBA. Julio 31 de 2018.
Actualización	Inclusión de dos líneas en el cronograma (Examen Final y Socialización) y una línea del programa al cual se le va a dictar la asignatura en la presentación del contenido.	
Actualización competencias	Requerimientos Institucionales	Acta # 11 del Claustro Docente de diciembre 13 de 2018
Actualización Contenido programático.	Inicio periodo académico 2019-1	Acta 9 de diciembre 19 de 2018 del Comité de Currículo y Autoevaluación de la Facultad.
Inclusión de rúbricas	Requerimientos Institucionales	Acta 04 de abril 8 de 2019 del Comité de Currículo y Autoevaluación de la Facultad.
Actualización del Contenido	Incluir en el contenido antes del cronograma el contenido programático de la asignatura.	Acta 05 de mayo 27 de 2019 del Comité de Currículo y Autoevaluación de la Facultad.

El uso no autorizado así como la reproducción total o parcial de su contenido por cualquier persona o entidad, estará en contra de los derechos de autor.

UNIVERSIDAD MILITAR NUEVA GRANADA



CONTENIDO PROGRAMÁTICO	Fecha Emisión: 2018/02/09	AC-GA-F-8
	Revisión No. 3	Página 10 de 10

CAMBIO REALIZADO	JUSTIFICACIÓN DEL CAMBIO	ACTA DE APROBACIÓN
Actualización del Contenido	Modificación en el cronograma de las fechas para el semestre 2019-2	Acta # 8 del Comité de Currículo y Autoevaluación del DEPTEC, junio 10 de 2019.
Actualización del Contenido	Porcentaje de las Evaluaciones Actualización rubricas de evaluación.	Acta # 22 del Comité Asesor y Curricular del DEPTEC. Diciembre 02 del 2019
Actualización Contenido programático.	Inicio periodo académico 2020-1	Acta No. 11 de diciembre 18 de 2019 Comité de Currículo y Autoevaluación de la Facultad.
Actualización del Contenido programático.	Modificación en el cronograma de las fechas para el semestre 2020-1, ajuste al porcentaje en las actividades.	Acta de reunión # 8 del Comité asesor y curricular del DEPTEC junio 6 de 2020
Actualización Contenido programático.	Inicio periodo académico 2020-2	Acta No. 06 de junio 18 de 2020 Comité de Currículo y Autoevaluación de la Facultad.
Actualización del Contenido	Actualización de fechas de clase, actualización de docentes, actualización de la bibliografía y se revisaron los enlaces de red	Acta # 19 del Comité Asesor y Curricular del DEPTEC. Noviembre 26 del 2020
Actualización de fechas para el periodo 2021-1	Modificación de fechas para el primer semestre	Acta 11 de Comité de Currículo de Facultad, diciembre 15 de 2020