

CURSO:	FUNDAMENTOS DE PROGRAMACIÓN
---------------	------------------------------------

1	SEMESTRE:	I	
2	CÓDIGO:	602101	
3	COMPONENTE:	Tecnología	
4	CICLO:	Fundamentación	
5	ÁREA:	Profesional	
6	FECHA APROBACIÓN:		
7	NATURALEZA:	Teórico-Práctico	
8	CARÁCTER:	Obligatorio	
9	CRÉDITOS (RELACIÓN):	4 (1-1)	
10	INTENSIDAD HORARIA:	192	Total Horas/Semestre
		96	Horas/Semestre de trabajo presencial
		96	Horas/Semestre de trabajo independiente

11. JUSTIFICACIÓN:

El desarrollo de Software es un aspecto fundamental en la solución práctica de problemas, para lo cual una herramienta tecnológica actualizada es materia importante para lograr la implementación automatizada de dicha solución, tal herramienta es un lenguaje de programación, el cual brinda la sintaxis necesaria para que el Ingeniero realice el aplicativo útil para los usuarios, complementado con el análisis de problemas para que estas soluciones cumplan a cabalidad con el objetivo trazado.

12. PROPÓSITOS:

- Comprender el concepto y el desarrollo de algoritmos y su aplicación en la solución de problemas.
- Desarrollar capacidades de abstracción en el análisis de problemas a resolver.
- Desarrollar hábitos relacionados con el autoaprendizaje y la cultura, necesarios para la solución de problemas.
- Solucionar problemas básicos, los cuales irán subiendo su grado de complejidad en un lenguaje de programación determinado.

13. COMPETENCIAS QUE DESARROLLARÁ EL CURSO:

- Identificar el orden lógico de situaciones cotidianas y su aplicación en la solución de problemas.
- Conocer los fundamentos de la programación.

- Identificar los aspectos importantes de la evolución y tipos de lenguajes de programación.
- Considerar los aspectos generales de un lenguaje de programación.
- Desarrollar la capacidad de resolver problemas utilizando algoritmos eficientes.
- Desarrollar la capacidad de representar soluciones a problemas cotidianos/reales a través de las diferentes herramientas de diagramación.
- Relacionar los tipos de datos y la forma de almacenamiento de los mismos.
- Identificar las secuencias de instrucciones necesarias para especificar las operaciones que debe ejecutar la computadora.
- Manejar la sintaxis de un lenguaje de programación determinado.
- Conocer y aplicar de manera eficiente y eficaz las diferentes estructuras de control (Secuenciales - selectivas - repetitivas) dentro de un lenguaje de programación.
- Realizar programas modulares o funcionales.
- Ser capaz de desarrollar programas que requieran operaciones de clasificación de datos.

14. DIMENSIONES DE LAS COMPETENCIAS QUE DESARROLLA:

COGNITIVA	PRAXIOLÓGICA	ACTITUDINAL
Comprender el desarrollo de un proceso lógico.	Determinar necesidades y requerimientos que se van a utilizar y aplicar en la solución de problemas.	Objetividad para analizar situaciones cotidianas.
Comprender la historia, evolución de la programación, los lenguajes y el impacto social que desencadena. Comprender los procesos que se llevan a cabo internamente cuando el sistema de computación esta encendido y el flujo de datos.	Participar en el equipo de trabajo, aportando ideas y soluciones utilizando una terminología adecuada.	Trabajar en equipo, y participación en grupo.
Comprender el concepto de traductor, compilador e intérprete.	Establecer un criterio para determinar el uso del proceso de traducción o compilación.	Motivación para profundizar en los conocimientos.
Conocer la evolución y los tipos de lenguajes de programación.	Establecer la diferencia entre lenguajes de alto, medio y bajo nivel.	Motivar para profundizar en los conocimientos.
Conocer las características, ventajas y desventajas de algunos lenguajes de programación.	Establecer criterios para analizar las ventajas o desventajas de la utilización de un lenguaje de programación para resolver un problema determinado.	Adquirir la habilidad crítica.

Comprender el concepto de algoritmo.	Analizar y describir una situación real mediante secuencia de pasos lógicos.	Desarrollar la abstracción.
Representar mediante diagramación las soluciones dadas mediante algoritmos a situaciones/problemas cotidianos. Utilizando los diagramas de flujo y los diagramas N-S.	Asociar eventos simples y complejos a representaciones graficas estandarizadas (ANSI).	Desarrollar la habilidad de expresar sus ideas de forma gráfica.
Conocer la forma en que se almacena la información en el computador.	Diferenciar y utilizar los tipos de datos de manera óptima.	Desarrollar la habilidad de optimizar recursos.
Comprender el funcionamiento de las instrucciones en cada paso.	Asociar a cada forma de representación una instrucción.	Adquirir la habilidad de asociación.
Conocer la estructura de un programa en un lenguaje determinado.	Dar soluciones a un problema utilizando un lenguaje de programación específico.	Desarrollar hábitos de autoaprendizaje, necesarios para la solución de problemas.
Comprender, analizar y aplicar de manera adecuada las diferentes estructuras de control (Secuenciales - selectivas - repetitivas) en un programa.	Utilizar las estructuras de control de una manera adecuada y eficaz para codificar algoritmos previamente diseñados y depurados.	Desarrollar la lógica y la habilidad para optimizar situaciones concretas de manera rápida.
Comprender la estructura de una función.	Desarrollar programas de forma modular o funcional.	Adquirir la habilidad de descomponer un problema para solucionarlo de una manera más eficiente.
Comprender la forma como se almacenan los datos en un vector y en una matriz.	Desarrollar programas utilizando vectores y matrices.	Aprovechar las características que ofrecen las matrices y los vectores para optimizar las soluciones.
Conocer los algoritmos clásicos de ordenamiento y búsqueda de datos.	Implementar los algoritmos clásicos de ordenamiento y búsqueda de datos.	La capacidad de selección de herramientas adecuadas a casos específicos.

15. UNIDADES TEMÁTICAS:

UNIDAD TEMÁTICA	ESTRATEGIA PEDAGÓGICA	RECURSOS PEDAGÓGICOS	TIEMP. PRES.	TIEMP. INDEP.
Estrategias para el análisis y solución de problemas.	Mapas conceptuales, solución a problemas cotidianos/reales, talleres individuales, debate/intercambio de criterios sobre guías clase anterior.	Computador portátil. VideoBeam Clase Magistral	4	4
Generalidades de programación	Líneas de tiempo, cuadros sinópticos, mapas conceptuales, Análisis de Impacto actual. Debate/Intercambio de	Clase Magistral. Monitor	4	4

UNIVERSIDAD DE LOS LLANOS
 Facultad de Ciencias Básicas e Ingeniería
 Programa Ingeniería de Sistemas



	critérios.			
Concepto de compilador, interprete	Conferencia Traductores (Interpretes y Compiladores), Funcionamiento y técnicas actuales usadas por los lenguajes de programación.	Clase Magistral. Monitor	4	4
Evolución de los lenguajes de programación	Video - computación. - ensayo. - análisis de etapas de la historia de la programación. - debate. Video Introducción a la computación. Resumen escrito y guías con mapas conceptuales.	Videobeam Computador portátil. Clase Magistral	6	6
Generalidades de los lenguajes de programación	Conferencia JAVA/C/C++. Discusión sobre los diferentes puntos de vista. Ventajas, Aplicaciones, Entorno, Utilidad, Conclusiones. Ensayo individual.	Clase Magistral Monitor	6	6
Algoritmia	Mapas conceptuales. - solución a problemas cotidianos/reales. - talleres individuales	Clase Magistral Monitor	8	8
Diagramación	Fundamentación teórica - ejemplos - ejercicios resueltos por el docente. - Resumen Escrito (guía) estándares ANSI diagramación "Flujo" y "N-S". Realización por parte de los estudiantes de ejercicios. - talleres.	Computador portátil VideoBeam Clase Magistral	6	6
Tipos de datos	Fundamentación teórica, Mapas Conceptuales - Ejemplos - ejercicios resueltos por el docente. - Resumen Escrito (guía) a través de cuadro sinóptico. Optimización por parte de los estudiantes de ejercicios. - talleres.	Computador portátil VideoBeam Clase Magistral Laboratorio asistido Monitor	4	4
Instrucciones	Conferencia IDE's JAVA/C/C++. Y demás herramientas de compilación. Ejemplos prácticos.	Computador portátil VideoBeam Clase Magistral Laboratorio asistido Monitor	4	4
Estructura de un programa	Fundamentación teórica - ejemplos - ejercicios	Clase Magistral Monitor	4	4

	resueltos por el docente. - Lenguaje JAVA. - talleres. Cuadros comparativos de Semejanzas y diferencias con Lenguajes de programación JAVA y C.			
Control de flujo	Mapas conceptuales y cuadros comparativos en diferentes entornos de desarrollo y herramientas de programación, las estructuras de control flujo en JAVA.	Clase Magistral Laboratorio asistido Monitor Portátil - Videobeam	14	14
Funciones y procedimientos	Fundamentación teórica - ejemplos - ejercicios resueltos por el docente. - Optimización por parte de los estudiantes de ejercicios. - talleres. En JAVA.	Clase Magistral Laboratorio asistido Monitor Portátil Videobeam	12	12
Arreglos unidimensionales y bidimensionales	Fundamentación teórica - ejemplos - ejercicios resueltos por el docente. - Optimización por parte de los estudiantes de ejercicios. - talleres. En JAVA.	Clase Magistral Laboratorio asistido Monitor Portátil Videobeam.	8	8
Algoritmos de Ordenamiento y búsqueda	Fundamentación teórica - ejemplos - ejercicios resueltos por el docente. - Optimización por parte de los estudiantes de ejercicios. - talleres. En JAVA.	Clase Magistral Laboratorio asistido Monitor Portátil Videobeam.	6	6
Introducción a la Programación Orientada a Objetos	Evolución, ventajas, desventajas y características. Conceptos Básicos Clase Objeto Atributos y Métodos Alcance y visibilidad de los miembros de una clase: atributos y métodos	Clase Magistral Laboratorio asistido Monitor Portátil Videobeam.	6	6
TOTAL (Horas)			96	96

16. SISTEMA DE EVALUACIÓN

Las pruebas consideradas en el curso para evaluar las competencias son : Prueba oral, Prueba escrita, pruebas apoyadas en guías de observación, escalas de actitudes, cuestionarios, entrevistas, y finalmente las pruebas basadas en el análisis y verificación de la actuación real o simulada o en la apreciación de la calidad de productos terminados.

Estos tipos de pruebas, se clasifican como: PARCIALES, evalúan el desarrollo progresivo del estudiante durante el semestre y FINALES que evalúan el desarrollo de las competencias propuestas por el curso, al final del periodo académico.

1ER CORTE	2DO. CORTE	3ER. CORTE
ÍTEM	ÍTEM	ÍTEM
Parcial 15%	Parcial 15%	Funcionamiento proyecto final. 15%
Talleres, quices 10%	Talleres, quices 10%	Sustentación proyecto final 15%
Laboratorio 10%	Laboratorio 10%	

17. FUENTES BIBLIOGRÁFICAS:

17.1 FUENTES DE CONSULTA BÁSICA:

TÍTULO	AUTOR	EDITORIAL	AÑO	DISPONIBLE UNILLANOS		
				Físico	Virtual	No Disp.
CÓMO PROGRAMAR EN C/C++ Y JAVA 4 EDICION	DEITEL, HARVEY M.	PRENTICE HALL	2004			
COMO PROGRAMAR EN JAVA (QUINTA EDICION)	DEITEL, HARVEY M.	PRENTICE HALL	2004			
INTRODUCCIÓN A LA PROGRAMACIÓN ORIENTADA A OBJETOS Y EL LENGUAJE JAVA	UZURIAGA LÓPEZ, VIVIAN LIBETH; OSPINA SÁNCHEZ, BERNARDO		2005			
MANUAL DE JAVA	NAUGHTON, PATRICK; DE CORDOBA HERRALDE, RICARDO. TR	MAGRAW - HILL	1996			
INTRODUCCIÓN A LA PROGRAMACIÓN CON JAVA: UN ENFOQUE ORIENTADO A OBJETOS	ARNOW, DAVID M; WEISS, GERALD; SEVILLA RUIZ, DIEGO. TR.	ADDISON WESLEY	2001			
JAVA 2: CURSO DE PROGRAMACIÓN 3 EDICION	CEBALLOS SIERRA, FRANCISCO JAVIER	ALFAOMEGA	2006			
JAVA 2: MANUAL DE USUARIO Y TUTORIAL 4 EDICION	FROUTE QUINTAS, AGUSTÍN	ALFAOMEGA	2006			
ALGORITMOS –	CESAR A.		1996			

CONCEPTOS BASICOS (3RA EDICION)	BECERRA SANTAMARIA					
JAVA 2: MANUAL DE USUARIO Y TUTORIAL	FROUFE QUINTAS, AGUSTIN	RA-MA	2008			
PROGRAMACIÓN ORIENTADA A OBJETOS CON JAVA	FRANCISCO GUTIERREZ; FRANCISCO DURÁN; ERNESTO PIMENTL	RA-MA	2007			
JAVA 2: INTERFACES GRAFICAS Y APLICACIONES PARA INTERNET. 3ª EDICIÓN	CEBALLOS SIERRA, FRANCISCO JAVIER	RA-MA	2008			
PROGRAMACION ORIENTADA A OBJETOS CON JAVA	BARNES, DAVID J.; KÖLLING, MICHAEL	RA-MA				
PROBLEMAS RESUELTOS DE PROGRAMACIÓN EN LENGUAJE JAVA	JESÚS CARRETERO PÉREZ; FÉLIX GARCÍA CARBALLEIRA; JOSÉ MANUEL PÉREZ LOBATO; JOSÉ MARÍA PÉREZ MENOR	RA-MA	2006			
EL ARTE DE PROGRAMAR EN JAVA	SCHILDT, HERBERT & HOLMES, JAMES	MC GRAW HILL	2004			
JAVA PARA LA EMPRESA	SAVIT; WILCOX & JAYARAMN	MC GRAW HILL	2000			
JAVA. BIBLIOTECA DEL PROGRAMADOR	JAMSA	MC GRAW HILL	1997			
PROGRAMACIÓN EN JAVA	SOSA SÁNCHEZ-CORTÉS RICARDO; REINOSO PEINADO ANTONIO JOSÉ; MORENO DÍAZ PILAR; FERNÁNDEZ MANJÓN BALTASAR; HUECAS FERNÁNDEZ-	MC GRAW HILL / INTERAMERICAN A DE ESPAÑA, S.A.	2009			

	TORIBIO GABRIEL; SÁNCHEZ JESÚS					
EL LENGUAJE DE PROGRAMACION JAVA 3E	DAVID HOLMES, JAMES GOSLING, KEN ARNOLD	PEARSON EDUCACIÓN	2001			

CIBERGRAFÍA

- www.programacion.net
- www.lawebdelprogramador.com
- www.lavariante.com
- www.programacion.com
- www.programar.net

17.2 FUENTES DE CONSULTA PARA PROFUNDIZACIÓN:

TITULO	AUTOR	EDITORIAL	AÑO	DISPONIBLE NILLANOS		
				Físico	Virtual	No Disp.
UNA HERRAMIENTA PARA LA POO	BECERRA CESAR	EDITORIAL POR COMPUTADOR				
FUNDAMENTOS DEL DISEÑO Y LA PROGRAMACIÓN ORIENTADO A OBJETOS	FERNÁNDEZ SERGIO	MC GRAW HILL				
INGENIERÍA DEL SOFTWARE UN ENFOQUE PRÁCTICO	PRESMAN ROGER	MC GRAW HILL				
ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS	BOOCH GRADY	ADDISON WESLEY				
PROGRAMACIÓN ORIENTADA A OBJETOS	JOYANES LUIS	MC GRAW HILL				

18. RECURSOS Y MEDIOS TECNOLÓGICOS:

- Videobeam
- Computador portátil
- Laboratorios con PC's
- Compilador de Lenguaje JAVA
- Editor de Textos
- Software Curso Multimedia Interactivo (Freeware)
- Acceso a Internet
- Sitios Web descritos en las fuentes bibliográficas.

19. RECURSOS HUMANOS (Equipo profesional que participará en el desarrollo del curso)		
TIPO	PERFIL	FUNCIÓN
Docente	Ing. de Sistemas.	Teoría y práctica.
Estudiante Monitor		
Auxiliar de laboratorio		
Director de Programa Ingeniería de Sistemas		
Docentes Línea Ingeniería de Software FCBI - Unillanos		