

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE TECNOLOGÍA
PROGRAMA DE INGENIERÍA DE MANUFACTURA

ASIGNATURA:	PROGRAMACIÓN DE COMPUTADORES
CODIGO:	IMFD43
AREA:	Básicos de ingeniería
REQUISITO:	CB223
HORAS SEMANALES:	4
HORAS TEORICAS:	2
HORAS PRACTICAS:	3
CREDITOS ACADEMICOS:	3
SEMESTRE:	Tercero
FECHA DE REVISION:	Octubre de 2019
Profesores	Ing. Juan Felipe Arroyave Ing. Wilson Pérez

JUSTIFICACIÓN

El Tecnólogo de hoy, dentro de su campo de trabajo, se ve obligado a manejar, en forma directa o indirecta, su información a través de un computador. Por ello es importante que dentro de su vida universitaria, exista un espacio donde adquiera las bases sobre informática, incluyendo los conceptos básicos de algoritmos, procesos computacionales y programación. Con esto se refuerzan la capacidad y habilidad para entender problemas en general y estructurar soluciones para los mismos.

TRANSVERSALIDAD DEL CURSO

El curso de Programación de computadores demanda de los conocimientos y las competencias adquiridas en asignaturas como Álgebra Lineal, con el propósito de generar intercomunicación y enriquecimiento recíproco entre docentes y estudiantes, y propiciar un proceso de enseñanza-aprendizaje integral. No menos importante resulta la transversalidad resultante de la interacción con la industria a través de visitas técnicas y trabajos académicos realizados en las mismas.

COMPETENCIAS

El estudiante debe mostrar la posesión individual de un conjunto de conocimientos, habilidades y actitudes que le permitan un buen desempeño en su función productiva acorde al contexto. La combinación de la aplicación de conocimientos, habilidades o destrezas con los objetivos y contenidos del trabajo a realizar, se expresa en el saber, el saber hacer y el saber ser, manifestadas no sólo en la formación sino en la actuación. El estudiante, apoyado en el proceso de formación, deberá desarrollar y consolidar las siguientes competencias:

Competencias específicas disciplinares

El presente programa contribuirá principalmente a desarrollar en el estudiante las siguientes competencias:

- Aprender la importancia del computador con base en una breve reseña histórica.
- Entender la terminología utilizada en el área de procesamiento electrónico de datos.
- Pensar, crear y manipular algoritmos para la solución de problemas en forma estructurada.
- Adquirir el manejo teórico-práctico y de equipos para la codificación y ejecución de algoritmos lenguaje de programación como Python y Visual Basic.

Competencias específicas profesionales

- Conocer la terminología básica usada en la informática.
- Describir la configuración básica de un computador digital.
- Entender las estructuras de control fundamentales y las Técnicas para buena calidad.
- Conocer la sintaxis y semántica de un lenguaje de programación.
- desarrollar
- Analizar y diseñar soluciones a problemas, codificarlas y ejecutarlas en un lenguaje de programación.

METODOLOGÍA

- Clases magistrales, interactuando con el estudiante, haciéndolo participe de su proceso de formación
- Solución de problemas enfocados a afianzar los conceptos y a desarrollar habilidades analíticas
- Desarrollo dirigido de talleres en forma individual o en grupo
- Ejecución de prácticas para aplicar los conceptos de la clase teórica mediante el uso herramientas computacionales.
- Planteamiento, análisis y solución de problemas en forma dirigida
- Se realiza un proyecto que consiste en crear los planos de trabajo de una maquina aplicando todos los conceptos adquiridos en las asignaturas correspondientes al tema.

CAPÍTULO 1. INTRODUCCIÓN A LA PROGRAMACIÓN

Total horas: 6

Contenido Teórico-Práctico	Hr
1.1 Terminología básica en informática. Componentes de un computador digital	2,0
1.2 Diferentes tipos de Software: Sistemas Operativos, lenguajes de Programación, Paquetes, etc.	2,0

1.3 Estructuras básicas de control. Diagramación estructurada.	2,0
--	-----

CAPÍTULO 2. PROGRAMACIÓN Total horas: 32	
Contenido Teórico-Práctico	
	Hr
2.1 Conceptos básicos: ¿Qué es Python?, Operaciones sencillas, flotantes, cadenas de texto, editor	2,0
2.2 Estructuras de Control	2,0
2.3 Funciones y módulos	2,0
2.4 Excepciones y archivos	6,0
2.5 Programación funcional	8,0
2.6 Programación Orientada a objetos	12,0

CAPÍTULO 3. INTRODUCCIÓN Y UTILIZACIÓN DE ALGUNOS MÓDULOS EXISTENTES EN PYTHON. Total horas: 8	
Contenido Teórico-práctico	
	Hr
3.1 Servicios del sistema operativo 3.2 Software matemático. 3.3 Servicios y aplicaciones web. 3.4 Documentación y test de programas.	8.0

CAPÍTULO 4. BREVE INTRODUCCIÓN A LOS MACROS EN EXCEL Total horas: 24	
Contenido Teórico-práctico	
	Hr
4.1 Introducción	2.0
4.2 Concepto de macro	1.0
4.3 Grabación de macros	1.0
4.4 Uso de referencias absolutas.	4.0
4.5 Administrar las macros Cuadro de diálogo de macros. Ejecutar una macro. Eliminar una macro. Ubicación y opciones	4.0
4.6 Editar el código de las macros.	2.0
4.7 Introducción al entorno de Visual Basic para aplicaciones (vba). El modelo de objetos de Visual Basic: métodos y propiedades. Los módulos, los procedimientos y las palabras reservadas. Conceptos de programación estructurada. Inserción de código en una macro existente.	6.0
4.8 Desarrollo de aplicaciones.	2.0
4.9 Manejo de solver.	2.0

BIBLIOGRAFÍA	
Libros	
<ul style="list-style-type: none"> • Joyanes, Aguilar, Luis. Fundamentos de Programación. Editorial Mc Graw-Hill. • Apuntes y notas de los docentes. • C. A software engineering approach. • Peter A Darnell, Philip E. Margolis. Springer-Verlag, 1991 • Downey, Allen. How to think like a computer scientist : learning with Python. Wellesley: Green Tea Press, 2002. ISBN 9781441419071. • Chun, Wesley. Core Python programming. 2nd ed. Upper Saddle River: Prentice Hall, 2007. ISBN 0132269937. • Guzdial, Mark. Introduction to computing & programming in Python : a multimedia approach. 2nd ed. Upper Saddle River: Pearson : Prentice Hall, 2010. ISBN 9780136060239. • Langtangen, Hans Petter. Python scripting for computational science. 3rd ed. Berlin: Springer, 2008. ISBN 9783540739159. Pilgrim, Mark. Dive into Python. New York: Apress, 2004. ISBN 1590593561. 	
Otros recursos:	
<ul style="list-style-type: none"> • http://wiki.python.org/moin/UsefulModules • http://www.python.org/community/sigs/current/edu-sig/ • ¿Introduction to Programming using Python¿, http://www.pasteur.fr/formation/infobio/python - Manual d'emacs, http://www.gnu.org/software/emacs/manual/emacs.html 	
EVALUACION	
<p>La asistencia y participación en las clases y talleres deberá ser evaluada. La evaluación del curso está compuesta de los siguientes aspectos:</p> <ol style="list-style-type: none"> 1. Primera Evaluación parcial [20%]: se llevará a cabo en la cuarta semana de clase. 2. Segunda Evaluación Parcial [20%]: se llevará a cabo en la décima semana de clase. 3. Tercera Evaluación Parcial [20%]: Última semana de clases. 4. Examen Final [25%]: En la semana de exámenes programada. Se evaluará todo el curso. 5. Trabajos [15%]: contempla evaluaciones en clase, talleres, trabajos. 	