

PROGRAMACIÓN DEL SÍLABO DE APRENDIZAJE

I. DATOS GENERALES

1. Facultad : Ingeniería  
2. Departamento Académico : Ingeniería en Informática y Sistemas  
3. Nombre de la Asignatura : Fundamentos de programación  
4. Escuela Profesional : Ingeniería en Informática y Sistemas  
5. Docente(s) : Ing. Hinojosa Ramos, Edwin Antonio  
6. Régimen : Semestre I  
7. Año Académico : 2016  
8. Horas de Clase : Teoría: 04 / Práctica: 02  
9. Número de créditos : 04  
9. Ambiente donde se realiza las sesiones de aprendizaje  
Turno Mañana  
Teoría : Facultad de Ingeniería (Jueves 11.00-13.00hrs. Viernes 11.00-13.00hrs.)  
Práctica : Laboratorio de Informática ESIS (Martes 11.00-13.00hrs.)  
Turno Tarde  
Teoría : Facultad de Ingeniería (Lunes 17.00-19.00hrs. Miércoles 14.00-16.00hrs.)  
Práctica : Laboratorio de Informática ESIS (Jueves 16.00-18.00hrs.)

II. SUMILLA

APORTE DE LA ASIGNATURA AL PERFIL PROFESIONAL

Tiene como propósito brindar al futuro profesional en ingeniería Informática y Sistemas, para comprender los fundamentos de la ciencia de la computación (lógica, teoría de autómatas y lenguajes formales, paradigmas y lenguajes de programación), así como su aplicación al desarrollo de software (métodos formales en ingeniería de software).

SUMILLA

Introducción. Ingeniería de Sistemas, Ciencia de la Informática, Ciencias de la Computación. Programación Basada en métodos formales. Diseño y complejidad algorítmica. Programación estructurada.

DESARROLLO DEL CURSO: Comprende dos unidades. Artículos científicos. Desarrollo software computacional aplicado de 16, 32 Y 64 bits.

III. COMPETENCIA(S) DE LA ASIGNATURA

- Diferencia la ciencia de sistemas, Informática y Computación, expresando mediante un algoritmo computacional, un modelo matemático y el diseño correspondiente del diagrama de flujo y codifica en un lenguaje de programación para resolver las aplicaciones por micro-computadora.
- Conoce los sistemas, tipos de datos y los códigos que se utilizan en la ciencia de la computación.
- Identifica los diferentes recursos necesarios para el desarrollo de páginas web desde una perspectiva actual.

IV. PROGRAMACIÓN DE CONTENIDOS

PRIMER SEMESTRE: del 01 de abril al 01 de Agosto 2016 Total Horas 108

COMPETENCIA (Conceptual, procedimental, actitudinal)	CONTENIDOS SIGNIFICATIVOS	CRONOGRAMA (Del – al)
Analiza, abstrae, crea y diseña algoritmos computacionales orientados a resolver problemas mediante el uso del ordenador. Optimiza el desempeño de los algoritmos usando la recursividad en los procesos.	<b>PRIMERA UNIDAD: ALGORITMOS</b>  <b>Capítulo I: Algoritmos y Técnicas de programación</b> ↳ Programación estructurada ↳ Programación modular ↳ Estructura secuencial ↳ Estructuras selectivas ↳ Estructuras repetitivas ↳ Estructuras anidadas ↳ Procedimientos y funciones ↳ Acoplamiento de módulos y subprogramas ↳ Manejo de punteros ↳ Implementación de algoritmos	Del 01-04-16 al 13-05-16

<p>Aplica técnicas de análisis de algoritmos para estructurar datos en memoria del ordenador. Planea y diseña experiencias y actividades traducidas en algoritmos que luego implementa mediante la Programación Orientada a Objetos</p>	<p><b>Capítulo II; Recursividad</b></p> <ul style="list-style-type: none"> <li>↗ Conceptos de Recursividad</li> <li>↗ Recursividad Vs. Iteración</li> <li>↗ Algoritmos Divide y Vencerás</li> <li>↗ Compilación por descenso recursivo</li> <li>↗ Principios de recursión</li> <li>↗ Ventajas y desventajas de los algoritmos recursivos</li> <li>↗ Eliminación de la recursión.</li> <li>↗ Ejercicios de aplicación</li> </ul> <p style="text-align: center;"><b>Primera Examen Parcial</b></p> <p><b>SEGUNDA UNIDAD: ARREGLOS Y PROGRAMACIÓN ORIENTADA A OBJETOS</b></p> <p><b>Capítulo I :</b> Arreglos unidimensionales y n-dimensionales como estructuras para almacenamiento de datos.</p> <ul style="list-style-type: none"> <li>↗ Arreglos</li> <li>↗ Almacenamiento de arreglos en memoria</li> <li>↗ Implementación de algoritmos de operaciones del álgebra lineal</li> <li>↗ Suma de matrices</li> <li>↗ Multiplicación de matrices</li> <li>↗ Solución de sistemas de ecuaciones algebraicas</li> <li>↗ Uso de los registros para el almacenamiento de los datos</li> <li>↗ Arreglos de registros</li> <li>↗ Registros de Registros</li> <li>↗ Registros con arreglos</li> <li>↗ Ejercicios de aplicación</li> </ul> <p><b>Capítulo II. PROGRAMACIÓN ORIENTADA A OBJETOS</b></p> <ul style="list-style-type: none"> <li>↗ Conceptos de Programación Orientada a Objetos (POO)</li> <li>↗ Objetos y Clases</li> <li>↗ Abstracción de datos</li> <li>↗ Concepto de clase</li> <li>↗ Objetos</li> <li>↗ Ámbito de una clase</li> <li>↗ Especificadores de acceso a los miembros de una clase</li> <li>↗ Funciones miembro de una clase</li> <li>↗ Constructores y destructores</li> <li>↗ Sobrecarga de funciones y Operadores</li> <li>↗ Herencia y Jerarquía de Clases</li> <li>↗ Funciones virtuales y polimorfismo</li> <li>↗ Encapsulamiento</li> <li>↗ Tópicos avanzados en programación</li> <li>↗ Trabajos de Aplicación</li> </ul> <p style="text-align: center;"><b>Segundo Examen Parcial</b></p>	<p>Del 16-05-16 al 10-06-16</p> <p>Del 13-06-16 al 01-07-16</p> <p>Del 04-07-16 al 29-07-16</p>
---	--	---

EVALUACIÓN:			
Procedimientos		Porcentaje	Ponderación
Teoría 60%	Examen parcial	45 %	9
	Asistencia, interés y esfuerzo personal.	15 %	3
Práctica 40%	Implementación de algoritmos en Laboratorio	30 %	6
	Trabajos de Investigación.	10 %	2

### **Observación**

- ♣ La calificación será vigesimal (de 00 a 20), siendo la nota mínima de aprobación final 11 (once).
- ♣ Sólo para el promedio final se considerará la fracción 0.5 a favor del estudiante.
- ♣ Los alumnos que no se presenten a rendir sus evaluaciones en las fechas señaladas tendrán la calificación 00. En caso la inasistencia sea debidamente justificada mediante los canales respectivos, según normas vigentes en la universidad, se reprogramará dicha evaluación
- ♣ *La evaluación de los estudiantes es permanente y el contenido de los exámenes es de todo el desarrollo del curso a la fecha de aplicación de la evaluación.*
- ♣ La inasistencia a clases y/o laboratorios puede ser causal de desaprobación del curso, según reglamentos académicos, propios de la universidad.

El promedio Final del curso se obtiene de acuerdo a los criterios de evaluación que se señalaron anteriormente.

### **V. METODOLOGÍA**

#### **Clases Teóricas:**

- a. Método: Analítico y deductivo.
- b. Técnica : Exposición temática en aula propiciando el análisis y participación activa del alumno
- c. Procedimiento: Planteamiento de problemas y la búsqueda de soluciones mediante el planteamiento de algoritmos computacionales. La complejidad de los problemas será progresiva, de acuerdo al avance del curso.

#### **Prácticas de laboratorio y/o en aula**

- a. Método : Análisis y validación de algoritmos como resultado de la solución de problemas.
- b. Técnica : Implementación de algoritmos mediante el uso de lenguajes de alto nivel.
- c. Procedimiento : Planteamiento de problemas, elaboración de algoritmos que soluciones estos problemas, implementación y depuración de los algoritmos. Evaluación de resultados.

### **VI. BIBLIOGRAFÍA**

- ✓ AHO, ALFRED, HOPCROFT J. y ULLMAN, J. (2006). *Estructura de Datos y Algoritmos*. New York: Addison-Wesley Iberoamericana
- ✓ JOYANES, Luis. (2012). *Fundamentos de programación algoritmos y estructura de datos* (Segunda Edición), España, Editorial McGraw-Hill/Interamericana
- ✓ CEVALLOS, Francisco (2006) *Programación en C++* (3ª edición). España, Editorial RA-MA
- ✓ HEILEMAN G. (2004). *Estructura de Datos, Algoritmos y Programación Orientada a Objetos* (2ª ed.). México: Mc Graw Hill
- ✓ JOYANES, L. (2009). *Fundamentos de Programación: Algoritmos y Estructura de Datos* (3ª ed.). México: McGraw-Hill
- ✓ JOYANES , Luis. (2006) *C++ a su alcance* (2ª edición) Universidad Pontificia de Salamanca de Madrid
- ✓ KRUSE, R. (1996). *Estructura de Datos y Diseño de Programas*. México: Prentice-Hall Hispanoamericana
- ✓ MANBER, U. (2001). *Introduction to Algorithms: A Creative Approach*. New York: Addison-Wesley Iberoamericana
- ✓ RAFFO, Eduardo (2004). *Algoritmos y Estructura de Datos*. Lima –Perú. Raffo-Lecca Editores
- ✓ SCHILDT, Herbert (2002). *C++ Guía de referencia completa* (4ª edición) Editorial Reviews
- ✓ THE MATH WORKS (2005) *MATLAB: Guía del Usuario*. España, Ed. Prentice – Hall International Group
- ✓ VASQUEZ PARAGULLA, J ( 2001) *Diseño de programación: 200 algoritmos y un proyecto de aplicación*. Lima – Perú Editorial San Marcos