

PROGRAMA DE ESTUDIOS: LENGUAJES DE PROGRAMACIÓN

PROTOCOLO

Fechas	Mes/año
Elaboración	05-2010
Aprobación	
Aplicación	09-2010

Clave			Semestre	7°		
Nivel	Licenciatura	X	Maestría		Doctorado	
Ciclo	Integración		Básico		Superior	X
Colegio	H. y C.S.		C. y T.	X	C. y H.	

Plan de estudios del que forma parte: Ingeniería de Software

Propósito(s) general(es):
 El estudiante conocerá los fundamentos, principios y aspectos de diseño e implantación de los principales lenguajes de programación, para que sea capaz de utilizar de modo mas adecuado un lenguaje de programación específico, y determinar qué lenguaje de programación es el más apropiado para utilizarlo en un proyecto de software.

Carácter		Modalidad				Horas de estudio semestral (16 semanas)					
Indispensable	X	Seminario		Taller		Con Docente	Teóricas	36	Autónomas	Teóricas	20
		Curso	X	Curso-taller			Prácticas	36		Prácticas	36
Optativa *		Laboratorio	X	Clínica		Carga horaria semanal: 4.5 + 3.5 = 8			Carga horaria semestral:		128

Asignaturas Previas:	Asignaturas Posteriores:
Construcción y evolución del software, Teoría de la computación	Arquitectura de software

Requerimientos para cursar la asignatura	Conocimientos y habilidades: Saber solucionar problemas mediante la ayuda de una computadora, por medio de programas
---	---

Perfil deseable del profesor:	Ingeniero en computación o alguna carrera afín, con maestría o doctorado de preferencia, en Ciencias de la Computación, o bien, amplios conocimientos y experiencia laboral en el área. Además, amplia experiencia docente y de investigación.
--------------------------------------	--

Academia responsable del programa: Informática	Diseñador (es): Candidato a M.C.C. Marcos López Chimil
--	--

* Aquellas en las que se ofrece la posibilidad de cursar una de las asignaturas, para cubrir un requisito INDISPENSABLE será considerada INDISPENSABLE.

INTRODUCCIÓN

El estudio de lenguajes de programación, al mostrar principios generales de diseño y uso, permiten utilizar de mejor modo un lenguaje específico, y facilitan el estudio y la utilización de un nuevo lenguaje, además, al considerar diferentes lenguajes de programación se logra tener una variedad amplia de posibilidades para expresar la solución a un problema determinado, y por tanto, la determinación del lenguaje más recomendado para ser utilizado en un proyecto de software, será más efectiva. Este conocimiento, proporciona otras ventajas más interesantes, a quien incluso, por necesidades de un problema planteado llegará a necesitar un nuevo modo de expresar la solución a dicho problema, fue así como surgió el lenguaje de programación Simula.

En este curso, en primer lugar, se estudiarán los conceptos y principios básicos de un lenguaje de programación, además, se realizará un estudio histórico enfatizando las causas del surgimiento y los aportes de los nuevos lenguajes de programación, mostrando las características de los principales paradigmas, así como algunos criterios utilizados para la clasificación de los lenguajes de programación, posteriormente, se estudiarán aspectos de diseño comunes en los lenguajes de los paradigmas más relevantes, con el fin conseguir una mejor comprensión del uso y funcionamiento de un lenguaje de programación, y finalmente, se realizará con más detenimiento el estudio de algunos lenguajes representativos de los paradigmas principales.

PROPÓSITOS GENERALES

El estudiante conocerá los fundamentos, principios y aspectos de diseño e implantación de los principales lenguajes de programación, para que sea capaz de utilizar de modo más adecuado un lenguaje de programación específico, y determinar qué lenguaje de programación es el más apropiado para utilizarlo en un proyecto de software.

CONTENIDOS

TEMAS Y SUBTEMAS	PROPÓSITOS ESPECÍFICOS
1 Conceptos y principios de un lenguaje de programación	El estudiante comprenderá los conceptos y principios básicos, que al ofrecer un tipo de solución dieron origen a los principales lenguajes de programación.
1.1 Conceptos básicos 1.2 Historia de los lenguajes de programación 1.3 Paradigmas y clasificación de los lenguajes de programación	
2 Aspectos de diseño de un lenguaje de programación	
2.1 Relación entre los recursos de una computadora y los lenguajes de programación 2.2 Control de datos y de secuencia de instrucciones 2.3 Administración de memoria 2.4 Modularidad 2.5 Sintaxis y Semántica	El estudiante comprenderá y aprenderá los aspectos más importantes que involucra el diseño de un lenguaje de programación.

3 Estudio de paradigmas	El estudiante comprenderá a través de los conocer algunos lenguajes de programación las características de los principales paradigmas en esta área.
3.1 Programación orientada a objetos	
3.2 Programación funcional	
3.3 Programación orientada a eventos	
3.4 Programación lógica	

METODOLOGÍA PARA EL CURSO

Este curso es teórico práctico. La materia se impartirá por medio de clases teóricas, prácticas y asesorías. Se estudiarán casos reales, de preferencia vinculados a procesos cercanos al estudiante, en los que pueda poner en práctica los conocimientos adquiridos en el curso durante el transcurso del semestre. El objetivo de este curso, es que los estudiantes conozcan las características de los lenguajes de programación de los principales paradigmas, incluidos, aspectos de diseño, para que aplique estos conocimientos en casos reales que le sean significativos, de tal modo, que comprenda la importancia de conocer los principios generales de los lenguajes de programación, y le facilite el uso y la utilización de un lenguaje específico en un proyecto de software, y en casos mas relevantes, determine su criterio para elegir el lenguaje mas adecuado para un proyecto de software.

Puesto que la aplicación en la práctica de los conocimientos adquiridos en el aula es indispensable, se tendrán sesiones en laboratorios de cómputo una vez por semana, en ellas, el estudiante pondrá en práctica los conceptos tratados en el curso realizando prácticas que tengan como fin lograr elaborar software de calidad.

EVALUACIÓN DIAGNÓSTICA

Se aplicará un examen escrito sobre los temas contenidos en el curso de Construcción y evolución del software. Esto con el fin de detectar los conocimientos con los que cuentan los estudiantes y sugerir acciones para que adquieran los que no tengan, por medio de un repaso general dentro de la clase o en asesorías individuales.

EVALUACIÓN FORMATIVA

Con el propósito de dar seguimiento al proceso de enseñanza aprendizaje, se propone aplicar tres evaluaciones formativas, correspondientes a los temas del curso, donde el estudiante mostrará sus conocimientos propuestos en dichos temas.

Los instrumentos para las evaluaciones formativas incluirán aspectos teóricos y prácticos, de tal modo que muestren sus avances con relación al conocimiento de los lenguajes de programación, en particular, deberá profundizar en las características de los lenguajes utilizados hasta ese momento en la carrera, mejorando su dominio de los mismos. Además de las evaluaciones formativas se dejarán tareas a los estudiantes al terminar cada tema. Se hará una valoración junto con los estudiantes al final de cada tema para hacer sólida la comprensión de los mismos. Se pondrá especial atención en el cumplimiento de las prácticas en el laboratorio, así como en la realización de las tareas, la participación y asistencia a clases, y las correspondientes asesorías, fortaleciendo la formación del estudiante.

EVALUACIÓN DE CERTIFICACIÓN

El examen de certificación evaluará los conocimientos del estudiante sobre todos los temas tratados durante el curso. Este instrumento consta de dos partes: una evaluación teórica mediante examen escrito y la segunda mediante el desarrollo de un proyecto. En esta, al igual que en otras materias de final de la ingeniería de software, lo más importante es que el estudiante lleve a la práctica el material tratado en clase, de aquí que la certificación escrita tendrá al 50% de la calificación final, mientras que la parte práctica se evaluará por medio de un proyecto y representará el 50% de la calificación final. En este proyecto debe mostrar que realmente ha alcanzado los propósitos del curso, y deberá servir como apoyo para utilizarlos en las materias que le siguen.

BIBLIOGRAFÍA

La bibliografía que se menciona a continuación es la básica para el estudiante y el profesor:

Bibliografía	Temas para los que se recomienda
1. Louden, Kenneth. Programming Languages. Principles and Practice. PWS Publishing Company. USA 1993	1, 2, 3
2. Schalkoff, Robert, J.; Programming Languages and Methodologies, Jones and Bartlett Publishers, 2007	3
3. Friedman, D; Wand, M, Hayness, C. Essentials of Programming Languages. McGraw Hill 1994.	1, 2, 3
4. Moriber, H. Structured Basic Programming. Charles E. Merrill Publishing Co. 1992.	1,2, 3
5. Pratt, T., Zelkowitz, M. Programming Languages, Design and Implementation. Prentice Hall. 1999.	1, 2
6. Sebesta, R. Concepts of Programming Languages. 4o Edition. Addison Wesley. 1999.	1, 3
7. Tucker, A. Noonan, R. Lenguajes de Programación, principios y paradigmas. McGraw Hill. 1998.	1, 2, 3

OTROS RECURSOS

Laboratorio de cómputo con acceso a Internet de alta velocidad, computadoras con un mínimo de 2 Gb en memoria principal, que tenga instalado una base de datos relacional y un Entorno de Desarrollo Integrado (IDE) (Eclipse, NetBeans y/o Visual Studio, cada una en su última versión), una herramienta case para sistemas orientados a objetos (ArgoUML), herramientas de depuración y pruebas (JUnit, Jtiger) y un paquete de oficina (OpenOffice). Además, pintarrón y proyector de video (cañón).