

Programa de estudios

Datos generales

0. Area Académica

Económico Administrativa

Programa educativo

Licenciatura en Informática

Facultad

Estadística e Informática

Código

Nombre de la Experiencia educativa

Programación de Sistemas

Area curricular

1. Básica general	2. Iniciación a la disciplina	3. Disciplinar X	4. Terminal	5. Electiva
-------------------	-------------------------------	-----------------------------	-------------	-------------

Proyecto integrador

Academia

LGAC o proyecto de vinculación con el que se articula:
 - Tecnología computacional y educativa
 - Ingeniería y tecnología de software

Software de Base

Requisito(s)

a. Prerrequisitos:

b) correquisitos:

Modalidad

Curso

Características del proceso de enseñanza aprendizaje

Individual	Grupal	2.1 Número mínimo: 5
	X	2.2 Número máximo: 30

Número de horas de la experiencia educativa

Teóricas: 4

Prácticas: 2

Total de Créditos

Total de horas

Equivalencias

10

90

Programación de Sistemas

Fecha de elaboración/modificación

Fecha de aprobación

a) Julio 2002

b) Enero de 2005

Nombre de los académicos que participaron en la elaboración y/o modificación

Elaboración:

V. Angélica García Vega.

Modificación:

V. Angélica García Vega
Patricia Carrión Méndez

Perfil del docente

Mínimo maestría en Ciencias de la Computación o equivalente, con formación profesional en Informática o computación o afin: con experiencia docente y preferentemente con experiencia profesional

Espacio

Institucional, taller de soporte de hardware y aula, centro de cómputo

Relación disciplinar

Descripción mínima

Esta experiencia es formativa, se localiza en el área disciplinar, a través de ella los alumnos distinguen los diferentes programas, técnicas y herramientas que soportan el desarrollo de otras aplicaciones, tanto en ambientes estáticos como en dinámicos. Todo lo anterior asumiendo una actitud de colaboración, creatividad, autoaprendizaje, responsabilidad y honestidad

Justificación

Como experiencia formativa dota a los alumnos de los elementos para: comprender la aplicación de métodos y técnicas que subyacen y soportan el desarrollo de aplicaciones de alto nivel, proporcionan los servicios requeridos por sistemas embebidos o empotrados. Refuerza la solución computacional de problemas.

Unidad de competencia

En un ambiente de colaboración, creatividad, autoaprendizaje, responsabilidad y honestidad el estudiante deberá usar herramientas de software (editores, cargadores, ligadores, depuradores y utilerías de desarrollo) previamente desarrolladas en el aula.

Articulación con los ejes

El estudiante al utilizar/escribir software de sistemas concretan los conocimientos adquiridos (eje teórico), usando herramientas de desarrollo de software (eje heurístico) mostrando colaboración, creatividad, autoaprendizaje, responsabilidad y honestidad (eje axiológico)

Saberes

1. Teóricos	2. Heurísticos	3 Axiológicos
1. Introducción - La programación de sistemas - El uso de herramientas en línea o entornos visuales en la programación de	Distinguirá los elementos que intervienen en un programa para poder ejecutarse	- Interés - responsabilidad - Creatividad - Originalidad - Colaboración

<p>sistemas (6 horas)</p> <p>2. Traductores - Interpretes - Generadores (ensambladores/compiladores) (18 horas)</p> <p>3. Cargadores y ligadores - Tipos de cargadores: Cargador absoluto Cargador relocizable Cargador dinámico - Tipos de ligadores Encadenador (12 horas)</p> <p>4. Editores En línea Visuales (10 horas)</p> <p>5. Procesadores de macros - Uso de macros (10 horas)</p> <p>6. Depuradores - Tipos de depuradores Dump Trace Referencias cruzadas - Uso de depuradores En línea Ambiente visual (12 horas)</p> <p>7. Bibliotecas - Bibliotecas estáticas - Bibliotecas dinámicas (10 horas)</p> <p>8. Utilerías -Tipos de utilerías - Desarrollo de utilería (12 horas)</p>	<p>El estudiante analizará los diferentes traductores que existen (sin entrar a los detalles de lenguaje), diferenciando entre lenguaje ensamblador y su traductor. Entenderá el proceso de traducción de lenguajes de diferentes niveles</p> <p>Comprenderá las funciones de los cargadores y ligadores, distinguiendo las ventajas y desventajas de los mismos. Identificación, clasificación, manejo y construcción de un cargador Identificación, clasificación, manejo y construcción de un encadenador</p> <p>Identificar, clasificación y uso de diferentes editores (notepad, Vi, etc) Entenderá y analizará las funciones de los editores</p> <p>Comprenderá la utilidad de las macroinstrucciones. Definirá y usará sus propias macros</p> <p>Identificará los distintos tipos de errores. Identificará y analizará las distintas herramientas para la solución de los diferentes errores</p> <p>Diseño, creación y uso de bibliotecas dinámicas (dll)</p> <p>Investigación, diseño de una utilería que permita establecer comunicación con herramientas externas (telefonos, implementos musicales, lector de código de barras)</p>	
---	--	--

9. Proyecto - Se definirá al inicio del periodo	Pondrá en práctica los conocimientos adquiridos durante todo el curso	
--	---	--

Estrategias metodológicas

De aprendizaje	De enseñanza
<ul style="list-style-type: none"> - Búsqueda y consulta de fuentes de información - Lectura, síntesis e interpretación - Mapas conceptuales - Análisis y discusión de casos - Elaboración de bitácoras personales - Discusiones grupales 	<ul style="list-style-type: none"> - Organización de grupos colaborativos - Dirección de prácticas - Estudio de casos - Seminarios - Tareas para estudios independiente - Exposición con apoyo tecnológico variado - Lectura comentada - Mapas conceptuales - Aprendizaje basado en problemas - Foros

Apoyos educativos

1. Materiales didácticos	2. Recursos didácticos
<ul style="list-style-type: none"> - Bibliografía que soporta a esta experiencia educativa - Manuales de lenguajes y herramientas para - Antología de la experiencia educativa 	<ul style="list-style-type: none"> - proyector - computadoras - Traductores de lenguajes de alto/bajo nivel (ensambladores, compiladores, encadenadores, cargadores (Perl,)) - Herramientas de desarrollo de sistemas, tanto para ambientes Linux, Windows y otros (LEX, BISON, YACC, PALM, WAV)

Evaluación del desempeño

Evidencia(s) de desempeño	Criterios de desempeño	Campo de aplicación	Porcentaje
Síntesis de temas investigados y participación en clase sobre los temas investigados	Documento escrito con claridad, completo y coherente que incluya su propia interpretación del tema(s) estudiado	Individual y en grupo de trabajo dentro del aula, en el centro de cómputo, en bibliotecas, en internet	10%
Prácticas con las diferentes herramientas	Práctica elaborada correctamente de acuerdo a las especificaciones del profesor	Individuales dentro del aula y centro de cómputo	20%

Proyecto de diseño	Proyecto completo de acuerdo a la Ingeniería de software.	Cénro de cómputo o aula taller en donde sea posible usar los lenguajes de programación (C o ensamblador)	20%
Exámenes escritos	Resolución acertada de reactivos de cada examen aplicado. Resolución acertada de ejercicios y/o temas, basados en el estudio de casos, los proyectos desarrollados y las prácticas	Individual en el aula de clases	50%

Acreditación

El estudiante acreditará la experiencia educativa:

Ordinario: con 80% de asistencias, entrega oportuna de tareas y trabajos de investigación bibliográfica, entrega de proyecto y calificación aprobatoria del examen.

Extra ordinarios: 65% de asistencias, entrega oportuna de tareas, trabajos de investigación, entrega del proyecto y calificación aprobatoria del examen.

Fuentes de información

1. Básica

- 1) Donovan John J. (1980) Programación de sistemas.
Edit. "El ateneo" Argentina.
- 2) Beck A. Programación de sistemas Mc Graw Hill

2. Complementarias

- 3). Eckel Bruce (1991) Aplique c++
Edit. Mc Graw-Hill México.
- 4). Pappas Chris and Murria. William III (1993) "Manual de Borland"
Edit Mc Graw Hill México
- 5) Faison Ted. Borland C++ . Programación Orientada a Objetos.
- 6) Manuales de programación de C, Java y ensamblador
- 7) Manuales de equipos de cómputo