

1.- DATOS DE LA ASIGNATURA

<p>Nombre de la asignatura : Software de Sistemas I</p> <p>Carrera : Licenciatura en Informática</p> <p>Clave de la asignatura :IFB-9334</p> <p>Horas teoría-Horas práctica-Créditos : 4-0-8</p>
--

2. UBICACION DE LA ASIGNATURA

a) RELACION CON OTRAS ASIGNATURAS DEL PLAN DE ESTUDIOS

A N T E R I O R E S	
ASIGNATURAS	TEMAS
Lenguaje ensamblador.	Arquitectura de computadoras y formato de tust.

P O S T E R I O R E S	
ASIGNATURAS	TEMAS
Software de sistemas II	Con todas las unidades

b) APORTACION DE LA ASIGNATURA AL PERFIL DEL EGRESADO

- Le proporciona el conocimiento necesario para comprender con más profundidad, la manera de que trabajan algunos de los programas más importantes del software de sistemas, como son: Los compiladores, los ensambladores, los interpretes los cargadores y los ligadores.

3. OBJETIVO(S) GENERAL(ES) DEL CURSO

Proporcionar los conocimientos fundamentales de las herramientas de desarrollo de sistemas, su organización y función, para utilizarlos en el desarrollo de sistemas de información más eficientes.

4. TEMARIO

NUM.	TEMAS	SUBTEMAS
I	Introducción a software de sistemas	1.1 Que es el software de sistemas 1.2 Porque estudiarlo 1.3 Diferencias con el software de aplicación
II	Ensambladores	2.1 El nivel del lenguaje ensamblador 2.2 Funciones de un ensamblador 2.3 Tipos de Ensambladores 2.4 El proceso de ensamble de uno, dos o más pasos 2.5 Literales y expresiones 2.6 Macros -Macroinstrucción -Declamación y Expansión -Tablas asociadas -Bibliotecas de macros 2.7 Ensamble condicional
III	Cargadores y Ligadores	3.1 Concepto de cargadores y ligadores 3.2 Funciones de un cargador y de un ligador 3.3 Tipos de cargadores y ligadores 3.4 El proceso de carga absoluta 3.5 El proceso de carga relocizable 3.6 Ligadura de subrutinas
IV	Compiladores	4.1 Funciones de un compilador 4.2 Análisis lexigráfico 4.3 Análisis sintáctico 4.4 Análisis semántico 4.5 Generación de código intermedio 4.6 Generación de código completo 4.7 Optimización de código 4.8 Herramientas generadoras de compiladores
V	Interpretes	5.1 Principios de la interpretación 5.2 Diferencias entre interpretes y compiladores 5.3 Ejemplo de un interprete