

*UNIVERSIDAD DE COSTA RICA
SEDE REGIONAL DE OCCIDENTE
INFORMATICA EMPRESARIAL*

*PROPUESTA
PROGRAMA DEL CURSO
IF-2000 PROGRAMACION I*

II CICLO LECTIVO 2007

Profesor

Lic. Johnny Chaves Darcia

I. ASPECTOS ADMINISTRATIVOS

El curso IF-2000 PROGRAMACION I es semestral y se desarrolla en 16 semanas con la modalidad presencial, aunque se pone mayor énfasis en la práctica. El profesor participante habrá de disponer de al menos 15 horas semanales para atender las actividades que el curso demanda. Tiene como requisito el curso IF-1300 Introducción a la Informática, a su vez es requisito de los cursos IF3000 Programación II e IF3001 Algoritmos y Estructuras de Datos. Es recomendable que el estudiante tenga conocimientos de programación de computadoras. Su nivel corresponde al segundo semestre del primer año. Tiene 4 créditos. El día y la hora es lunes 9-11:50, 13-13:50, jueves 13-16:50. Profesor Johnny Chaves D.

II. DESCRIPCION DEL CURSO:

El curso trata los temas relativos a la programación de computadoras referida a solución de problemas tanto de índole empresarial como científica.

III. PROPÓSITOS DEL CURSO

1. Construir opciones para el desarrollo de software de usuario.
2. Propiciar el estudio del desarrollo de programas desde la perspectiva de la ingeniería de software.
3. Sentar algunas bases del paradigma de la Programación Orientada a Objetos (POO).

IV. OBJETIVOS GENERAL Y ESPECIFICOS DEL CURSO:

Al finalizar el curso el estudiante será capaz de:

Desarrollar programas de usuario de cierto nivel de complejidad utilizando los lenguajes Java y C++, tanto para ambientes **Linux** como **Windows**. Estos programas se basarán en los siguientes paradigmas: 1) Programación Estructurada, 2) Programación Orientada a Objetos, y 3) Programación Web. Por ello, en lo posible, se utilizarán principios de Ingeniería de Software.

1. *Familiarizarse en mayor grado con la lógica de programación de computadoras, a partir de la programación estructurada.*
2. *Utilizar las características principales de la programación orientada a objetos.*
3. *Mostrar aplicaciones web.*

MOTIVACIÓN: Los temas a cubrir en el curso corresponden a un paradigma de programación de vanguardia (Programación Orientada a Objetos), y asimismo a un lenguaje de vanguardia (Java/C++), ambos muy utilizados en la industria nacional e internacional. Al futuro profesional le va a ser muy útil lo que aprenderá y lo comenzará a aplicar tan pronto deje el ambiente del aula.

V. CONTENIDO TEMATICO

El curso se ha organizado a partir de un conjunto de preguntas generadoras de los contenidos que se desarrollan tal como se describen seguidamente:

I ¿ Desarrollo de programas de computadoras con la Programación Estructurada ?

- . ¿ Qué es la programación estructurada como tecnología de software?
- . Desarrollo de algoritmos para problemas comerciales y científicos.
- . Proceso de desarrollo de software usando Java: IDE, compilar, ligar, ejecutar.
- . Estructuras de Control: If/else, switch, break, continue, ciclos (do-while, while, for) operadores de asignación, aritméticos, lógicos, otros.
- . Tipos de datos: primitivos, arreglos, hileras, Tipos de Datos Abstractos (TDA), class
- . Funciones: definición, prototipos de función, archivos de encabezado.
 - . Listas de parámetros, retorno, referencias y parámetros por referencia
 - . Clases para almacenamiento, Reglas de Alcance
 - . Sobrecarga de funciones, plantillas de función
- . Recursividad vs iteración
- . Bits, caracteres, hileras y estructuras
- . Arreglos: dimensiones, declaración, paso de arreglos a funciones
- . Apuntadores
- . Procesamiento de hileras: declaración, operaciones
- . Uniones y estructuras
- . Procesamiento de archivos: organización, métodos de acceso

II ¿ Desarrollo de programas con la Programación Orientada a Objetos ?

- . Introducción a la tecnología de objetos y al Lenguaje Unificado de Modelado (UML)
- . Pensando en objetos, definición de estructuras
- . Clases y abstracción de datos: alcances, miembros de la clase
- . Objetos y funciones miembro const, composición, funciones y clases friend, apuntador this, administración de memoria dinámica (new, delete), static, abstracción de datos y ocultamiento de información, clases contenedoras e iteradores, clases proxy
- . Constructor, destructor

- . Tipos de Datos Abstractos: clases de autorreferencia, listas enlazadas, pilas, colas, árboles
- . Herencia: clases base y derivada, miembros protected, relaciones “Usa un” y “Conoce a”, Ingeniería de software con herencia, herencia simple y múltiple
- . Polimorfismo: relaciones entre objetos, clases abstractas

III *¿ Desarrollo de programas web ?*

- . Introducción a XHTML
- . Introducción a la programación web con CGI (Common Gateway Interface)
- . Tipo de peticiones http, arquitectura multinivel, acceso a servidores web (Apache), CGI, formularios XHTML para enviar datos de entrada, cookies

VI. METODOLOGIA

Con base en la siguiente observación de que “ningún libro enseña a programar”, y sobre todo que la programación de computadoras solo se aprende con práctica y más práctica, en las clases programaremos distintos tipos de problemas continuamente. Ante ello el estudiante deberá practicar el *desarrollo* de dicha programación de computadoras continuamente. Asimismo deberá asistir a las lecciones con el material del texto previamente estudiado, conforme a los contenidos temáticos, para que pueda aplicarlos a los problemas a desarrollar.

El curso es teórico-práctico, donde el estudiante debe desarrollar problemas en grupo, que permita poner en práctica los conocimientos adquiridos, en donde se enfrente con problemas reales para que les encuentre una solución programada. Con lo que se pretende que el estudiante desarrolle más su iniciativa e inventiva al resolverlos. Las soluciones a los problemas serán discutidos en clase con el fin de permitir que todos los estudiantes evacuen sus dudas.

Las clases serán presenciales y de trabajo colectivo donde se desarrollarán los ejemplos de programas; sin embargo, también para propiciar la creatividad e iniciativa del estudiante se asignarán *exposiciones*, tanto individuales como en equipo.

Las *Tareas Programadas* reforzarán los principios estudiados en clase. En este aspecto es muy importante considerar por parte de los estudiantes, el cumplir cabalmente con los requerimientos que el profesor defina para la elaboración de la tarea, los cuales serán entregados por el profesor en su debido tiempo. Se harán quizzes en cualquier momento.

Todos los trabajos y Tareas Programadas deberán entregarse en la fecha programada, en caso contrario se rebajará un punto por día de atraso hasta el tercer día natural inclusive, porque después no se recibirán. Asimismo deberán entregar la *documentación* respectiva y brindar una *demonstración* en el laboratorio, con valor de 70% y 30% respectivamente.

La *documentación* incluye: 1) *PORTADA*: 1. Nombre y sigla del curso, 2. Nombre, número de carnet y nota interna de los estudiantes si es en grupo, 3. Número de Tarea Programada y breve descripción del problema; 2) *DESARROLLO*: 1. Índice de contenido, 2. Objetivos generales y específicos, 3. Descripción detallada del problema, asimismo información recolectada según la Bibliografía consultada, 4. Algoritmo y listado fuente, 5. Diseño gráfico de la Base de Datos (si hubiera), 6. Diseño de la Interfaz, 7. Salidas o Resultados;

3) *PARTE FINAL*: 1. Conclusiones, 2. En caso de que no le salió explicar las razones técnicas de ello. 3. Bibliografía que incluya las direcciones Internet consultadas.

VII. **EVALUACION** NOTA APROVECHAMIENTO:
2 Parciales * 40% + Quizzes y Exposiciones* 10% + Tareas Programadas * 50%

VIII **BIBLIOGRAFÍA**

- Deitel y Deitel, “**JAVA Cómo Programar**”, Edit. Prentice Hall **5ta Ed.**
- Deitel (Harvey) & Deitel (Paul), “**Cómo programar en C++**”, Cuarta edición, Prentice Hall, 2003.

IX. **CRONOGRAMA IF2000 Programación I, II-2007. (PROPUESTA)**

Se- sión Nº	Fecha	Modalidad de la sesión	Contenido temático	Actividad base
1	13-17 agosto	Presencial	-Reflexión: mercado de trabajo ¿ Cómo se desarrollan programas de computadoras con la programación estructurada ?	Trabajo colectivo
2	20-24	Presencial	¿ Cómo se desarrollan programas de computadoras con la programación estructurada ? -Programa del curso	Trabajo colectivo Conversatorio
3	27-31	Presencial	¿ Cómo se desarrollan programas de computadoras con la programación estructurada ? - Enunciado Tarea Programada	Trabajo individual y grupal
4	3-7 Set.	Presencial	¿ Cómo se desarrollan programas de computadoras con la programación estructurada ?	Discusión y Trabajo colectivo
5	10-14	Presencial	- ¿ Cómo se desarrollan programas de computadoras con la programación estructurada ?	Trabajo colectivo
6	17-21	Presencial	- Enunciado Tarea Programada ¿ Cómo se desarrollan programas de computadoras con la programación estructurada ?	Trabajo colectivo

7	24-28		- I Examen Parcial	
8	1-5 Oct.	Presencial	- ¿ Desarrollo de programas con la programación orientada a objetos ?	Participación colectiva
9	8-12	Presencial	- ¿ Desarrollo de programas con la programación orientada a objetos ? - Enunciado Tarea Programada	Participación colectiva
10	15-19	Presencial	- ¿ Desarrollo de programas con la programación orientada a objetos ?	Participación colectiva
11	22-26	Presencial	- ¿ Desarrollo de programas con la programación orientada a objetos ? - Enunciado Tarea Programada	Participación colectiva
12	29-2 Nov	Presencial	- ¿ Desarrollo de programas con la programación orientada a objetos ?	Participación colectiva
13	5-9	Presencial	- ¿ Desarrollo de programas con la programación orientada a objetos ? - Enunciado Tarea Programada	Participación colectiva
14	12-16	Presencial	- ¿ Desarrollo de programas web ?	Discusión grupal Participación colectiva
15	19-23		- ¿ Desarrollo de programas web ? - Enunciado Tarea Programada	Participación colectiva
16	26-30		- Segundo examen parcial	

<i>3 diciembre</i>	▪ <i>Entrega de promedios</i>
<i>10 diciembre</i>	▪ <i>Ampliación</i>