



UNIVERSIDAD DE COSTA RICA

**Sede de Occidente**

**Recinto de Tacaes de Grecia**

**Bachillerato en  
Informática Empresarial**

*Programa del Curso*  
**Programación II**

## Datos Generales

---

Sigla: **IF-3000**

Nombre del curso: **Programación II**

Tipo de curso: **Teórico-práctico**

Número de créditos: **4**

Número de horas semanales presenciales: **8**

Número de horas semanales de trabajo independiente del estudiante: **12**

Requisitos: **IF2000 Programación I**

Correquisitos: **Ninguno**

Ubicación en el plan de estudio: **III Ciclo**

Horario del curso: **Lunes y Jueves de 5:00 p.m. a 09:00 p.m.**

Nombre del Profesor: **Mag. Alejandro Ulate Campos**

Correo electrónico: **aulatec@gmail.com**

Horario de Consulta: **Sábados de 1:00 p.m. a 5:00 p.m.**

## Descripción del Curso

---

Este curso profundiza en el diseño y desarrollo de programas computacionales haciendo uso de un lenguaje de programación. Se realiza un estudio exhaustivo del paradigma de programación orientado a objetos y se introduce al estudiante en el análisis y diseño orientado a objetos empleando el lenguaje de modelado UML.

Además, se estudian algunas interfaces de programación de aplicaciones (API) y se introduce al estudiante en la construcción de aplicaciones basadas en el Web.

## Objetivo General

---

Desarrollar en el estudiante la capacidad de construcción de aplicaciones avanzadas, profundizando en el estudio del paradigma de programación orientada a objetos.

## Objetivo Específicos

---

Al finalizar el curso el o la estudiante estará en capacidad de:

- Aplicar los conceptos de la orientación a objetos en la construcción de aplicaciones de software.
- Desarrollar aplicaciones cuya implementación y estructura esté separada en capas lógicas.
- Formular e implementar modelos de clases basados en el lenguaje de modelado UML.
- Construir aplicaciones Web básicas.

## Contenido del Curso

---

- Manejo dinámico de memoria
  - o ArrayList
  - o Punteros
  - o Collection
  - o Iterator
  - o Vector
- Creación, lectura, escritura y actualización de archivos
  - o Archivos de acceso secuencial.
  - o Archivos de acceso aleatorio.
  - o Manejo de excepciones en java.
- Programación orientada a objetos
  - o Composición de clases.
  - o Miembros de clase estáticos.
  - o Definición, implementación y uso de Interfaces.
  - o Patrones de diseño
- Modelado de sistemas orientados a objetos con UML
  - o El análisis/diseño orientado a objetos (ADOO)
  - o El lenguaje de modelado unificado (UML)
  - o Diagramas de Clase: Identificación de Clases (atributos, operaciones), herencia e interfaces.
  - o Asociaciones de clases (uni-direccional, bi-direccional, clase asociación, agregación, reflexiva), multiplicidad y roles.
- Hilos
  - o Creación, administración y destrucción de hilos
  - o Múltiples hilos
  - o Los hilos en las aplicaciones Cliente-Servidor
  - o Sincronización
- Sockets
  - o Conexión de aplicaciones mediante sockets
  - o Sockets en aplicaciones Cliente - Servidor
- Introducción al desarrollo de aplicaciones Web en la plataforma J2EE
  - o Las aplicaciones Web
  - o Introducción al HTML
  - o Formularios HTML como recolectores de información
  - o Javaserver Pages (JSP)
  - o Javabeans
  - o Strut
  - o Java Server Face (JSF)

## Metodología

---

El curso presenta un eje de desarrollo teórico-práctico. El profesor desarrolla clases magistrales y realiza asignaciones (quices y tareas cortas) para que los estudiantes apliquen los conceptos discutidos. Se realizan laboratorios durante el curso. Los estudiantes desarrollan proyectos programados donde ponen en práctica y amplían los conocimientos adquiridos en el curso. Además, se asignarán temas de investigación a los estudiantes.

## Evaluación

---

Descripción	Porcentaje
III Exámenes Parciales	15% 15% 20%
Quices y tareas cortas	5%
5 Laboratorios	15%
I Proyecto	10%
I Trabajo de investigación	10%
2 Tareas Programadas	10%

Al menos un 60% de la evaluación debe realizarse en forma individual.

## Notas y Aclaraciones

---

- Los laboratorios se irán desarrollando semanalmente y se realizará una revisión parcial el día de la entrega de los mismos.
- La entrega de proyecto o tarea programada debe ir acompañada de la respectiva documentación. No se recibirá la misma posterior a la entrega.
- La comprobación de que alguna tarea, proyecto o examen es una copia aplicará las sanciones que contemple el Reglamento de Régimen Académico Estudiantil.  
Consultar en:  
[http://cu.ucr.ac.cr/normativ/regimen\\_academico\\_estudiantil.pdf](http://cu.ucr.ac.cr/normativ/regimen_academico_estudiantil.pdf)
- Sólo se recibirán discos etiquetados y cuyo contenido sea exclusivamente la asignación respectiva.
- Todas las tareas, trabajos de investigación y proyectos programados que sean asignados deben ser entregadas a la hora y fecha indicadas. En caso de que se atrase un día, perderán el 30% del valor del trabajo, si son dos días perderán el 60%, y en caso de que no sean entregados dentro de ese lapso de tiempo perderán todo el puntaje.
- El proyecto programado es obligatorio para aprobar el curso. La comprobación del proyecto es individual y obligatoria.
- Los exámenes cortos o quices NO se avisarán con anticipación y se podrán realizar en cualquier momento de la lección. Los exámenes y quices no se reponen en caso de ausencia injustificada.
- Mantener celulares en modo silencioso o apagado durante las lecciones.
- La aparición de un celular durante un examen o quiz anulará automáticamente el mismo.
- Se utilizará la plataforma de Mediación Virtual o cualquier otra definida por el profesor como apoyo del curso.

## Cronograma del Curso

---

Semana	Fechas	Temas y Actividades
1	08/03/10 – 11/03/10	-Entrega carta al estudiante -Presentación del curso -Tema 1: Manejo dinámico de memoria
2	15/03/10 – 18/03/10	-Tema 2: Creación, lectura, escritura y actualización de archivos
3	22/03/10 – 25/03/10	-Continuación tema 2.
4	29/03/10 – 01/04/10	-Semana Santa
5	05/04/10 – 08/04/10	-Tema 3: Programación orientada a objetos
6	12/04/10 – 15/04/10	-Tema 4: Modelado de sistemas orientados a objetos UML
7	19/04/10 – 22/04/10	- Continuación tema 4.
8	26/04/10 – 29/04/10	- Semana Universitaria
9	03/05/10	- <b>I Examen</b>
	06/04/10	- <b>Entrega I Tarea Programada</b> - Tema 5: Hilos.
10	10/05/10 – 13/04/10	- Continuación tema 5.
11	17/05/10 – 20/04/10	- Continuación tema 5. -Tema 6: Sockets
12	24/05/10 – 27/04/10	- Continuación tema 6.
13	31/05/10	- Continuación tema 6.
	03/06/10	-Tema 7: Introducción al desarrollo de aplicaciones Web en la plataforma J2EE
14	07/06/10	- <b>II Examen</b>
	10/06/10	- Continuación tema 7.
15	14/06/10	- <b>Entrega II Tarea Programada</b> - Continuación tema 7.
	17/06/10	- Continuación tema 7.
16	21/06/10 – 24/06/10	- Continuación tema 7.
17	28/06/10	- <b>Entrega y exposición I Trabajo Investigación</b>
	01/07/10	- <b>III Examen</b>
18	06/07/10	- <b>I Proyecto</b>
19	16/07/10	- <b>Ampliación</b>

## Bibliografía

---

- Deitel y Deitel (2008). **¿Cómo programar en Java?**. Séptima edición. México: Prentice Hall.
- Allen Weiss, Mark **Estructura de Datos en Java**, Addison Wesley, Madrid, 2000.
- Craig Larman. **UML y Patrones : Una introducción al análisis y diseño orientado a objetos y al proceso unificado**. Segunda Edición. Prentice Hall, España, 2003.
- Eckel, Bruce **Piensa en Java**, 2da. Edición, Prentice Hall, España, 2000.
- Luis Joyanes, **Problemas de Metodología de la Programación**, Segunda edición, Mc Graw- Hill, 2001.
- Luis Rodríguez B., Matilde Fernández, Luis Joyanes Aguilar. **Libro de problemas, Fundamentos de programación**. Segunda Edición, McGraw-Hill.2003
- Pender, Thomas. **UML: Weekend crash course**. Wiley Publishing, 2002
- Naftalin Maurice , Wadler Philip. **Java Generics and Collections**. USA. O'Reilly, 2006
- O'Docherty Mike. **Object-Oriented Analysis and Design Understanding System Development with UML 2.0**. Inglaterra. John Wiley & Sons, 2005
- Poo Danny, Kiong Derek, Ashok Swarnalatha. **Object-Oriented Programming and Java**. 2da edición. Inglaterra. Springer, 2008.