



PROGRAMA CURSO: IF-1300 INTRODUCCIÓN A LA COMPUTACIÓN E INFORMÁTICA
I Semestre, 2014

Datos Generales

Sigla: IF-1300

Nombre del curso: Introducción a la Computación e Informática

Tipo de curso: Teórico-Práctico

Número de créditos: 4

Número de horas semanales presenciales: 8

Número de horas semanales de trabajo independiente del estudiante: mínimo 6

Requisitos: Ninguno

Correquisitos: Ninguno

Ubicación en el plan de estudio: I Ciclo

Horario del curso: K 8:00 a 11:50 y V 8:00 a 11:50

Suficiencia: No

Tutoría: No

Datos del Profesor

Nombre: Mci. Denis González Herrera

Correo Electrónico: Denis.gonzalez@ucurso.info

Horario de Consulta: M 13:00 a 16:50

1. Descripción del curso

Este curso proporciona al estudiante una adecuada introducción a la informática brindándole conocimientos básicos. Se familiariza al estudiante con la historia de las computadoras, su uso, hardware y software actuales. Se aborda el desarrollo de algoritmos en pseudocódigo y programación. Dándole énfasis a la adquisición de destrezas en la solución de problemas con ejercicios desarrollados en un lenguaje de programación.

2. Objetivo General

Desarrollar en el estudiante las habilidades y destrezas necesarias para resolver problemas mediante el uso de un lenguaje de programación.

3. Objetivos específicos

- a. Brindar al estudiante los conceptos básicos relacionados a la computación e informática para que tenga un conocimiento general de esta disciplina.



- b. Enfrentar al estudiante con la temática de resolución de problemas a través del uso de pseudocódigos.
- c. Iniciar al estudiante en la programación a través de un lenguaje de programación de uso didáctico.

4. Contenidos

- 1 Introducción
 - 1.1 ¿Qué es la informática?
 - 1.2 La computadora y su desarrollo histórico.
 - 1.3 Componentes físicos y lógicos de la computadora.
- 2 Teorías
 - 2.1 Sistemas
 - 2.1.1 ¿Qué es un sistema?
 - 2.1.2 Teoría General de sistemas
 - 2.1.3 Conceptos y clasificación de los sistemas
 - 2.1.4 Desarrollo de los sistemas de información
- 3 Algoritmos en Pseudocódigos.
 - 3.1 Conceptos y características de los algoritmos en pseudocódigo
 - 3.2 Estructura de los algoritmos en pseudocódigo
 - 3.3 Resolución de problemas con pseudocódigo
- 4 Introducción a la programación
 - 4.1 Los lenguajes de programación
 - 4.2 Concepto y partes constitutivas de un programa
 - 4.3 Elementos básicos de un programa
 - 4.3.1 Tipos de datos
 - 4.3.2 Variables, constantes, expresiones
 - 4.3.3 Operadores aritméticos, unarios, asignación, relacionales y lógicos
 - 4.4 Estructuras de control
 - 4.4.1 El flujo de control de un programa y estructura secuencial
 - 4.4.2 Estructuras de selección
 - 4.4.2.1 Simple y doble (si-entonces, si-entonces-sino / if, if-else)
 - 4.4.2.2 Múltiple (según_sea, caso de / case
 - 4.4.2.3 Estructuras de selección anidadas
 - 4.4.3 Estructuras repetitivas (while, for, do while)
 - 4.5 Métodos, parámetros y argumentos
 - 4.5.1 Definición de métodos
 - 4.5.2 Estructura de métodos
 - 4.5.2.1 Variables locales y visibilidad en el programa
 - 4.5.2.2 Valor de retorno
 - 4.5.2.3 Parámetros en los métodos
 - 4.5.3 Llamada a los métodos desde el programa principal y de otros métodos
 - 4.5.4 Argumentos en la llamada de los métodos

5. Metodología

Se impartirán lecciones magistrales que el estudiante debe complementar con las lecturas que este recomiende. Se realizarán prácticas en clase. También se empleará el laboratorio de cómputo para



prácticas de diagrama de flujos y de programación. El estudiante debe contar con conocimientos básicos en el uso de la computadora.

Se harán evaluaciones periódicas por medio de exámenes, pruebas cortas, ejercicios programados y tareas.

Deberán participar en los laboratorios del curso, realizando ejercicios dados por la profesora, tanto en clase como para resolver en sus casas. Al finalizar el curso, los estudiantes en parejas desarrollarán una tarea programada que les será asignada.

6. Evaluación

Descripción	Porcentaje
Examen Parcial I	15%
Examen Parcial II	20%
Examen Parcial III	20%
Quices, Tareas Cortas , otros	10%
Laboratorios	10%
Investigación	10%
Tarea Programada	15%
Total:	100%

Consideraciones sobre la evaluación

- Es requerido que el estudiante desarrolle actividades que le permitan construir su conocimiento. Debe ser un creador para instrumentar y operacionalizar lo aprendido mediante las prácticas de laboratorio.
- El estudiante debe asistir a lecciones con el material, que el profesor(a) le asigne, previamente estudiado.
- Durante el curso, el estudiante debe participar activamente de los laboratorios con el propósito de practicar lo aprendido en las clases teóricas.
- Se realizará un proyecto final programado que integre los conocimientos adquiridos durante el curso. Este proyecto deberá ser desarrollado en grupos, no se permiten proyectos individuales. La comprobación del proyecto es individual y tiene un porcentaje específico dentro de la calificación final, la cual puede corresponder hasta el 100% de la nota del proyecto.
- Las calificaciones grupales (ej: avances del proyecto) se podrán calificar INDIVIDUALMENTE a cada miembro del grupo. (En caso que se descubra un encubrimiento de trabajo (fraude) se procederá con el rebajo respectivo en la nota a cada uno de los integrantes del grupo).
- Los estudiantes desarrollarán investigación de temas relacionados con la carrera que estimulen el interés del estudiante por formar su propia experiencia de aprendizaje.
- Los exámenes parciales serán avisados con una semana de antelación.
- Los exámenes cortos y los laboratorios se realizarán sin previo aviso, en cualquier momento de la lección.
- Se utilizará la herramienta de apoyo de Cursos Virtuales, moodle.ucrso.info. El estudiante debe matricularse en el curso digital utilizando la dirección de correo electrónico personal. En este sitio podrán encontrar el cronograma del curso actualizado, material digital, asignación y entrega de tareas, foros, laboratorios, calificaciones, etc.
- Para comunicación con el profesor(a) se utilizará únicamente el correo de la carrera (@ucrso.info). El estudiante deberá estar siempre pendiente de su correo electrónico, ya que es el medio que se utilizará para avisarle sobre actualizaciones en el sitio digital del curso, avisos, asignación de tareas, etc.



- Para cualquier aspecto de evaluación no se aceptarán entregas después de la fecha y hora solicitada, a excepción de que en la especificación del mismo se indique lo contrario. No se harán reposiciones excepto por las disposiciones establecidas en el reglamento de Régimen Académico Estudiantil.
- Las fechas de las evaluaciones del cronograma están sujetas a cambio dependiendo del avance de los temas vistos.
- Mantener celulares en modo silencioso o apagado durante las lecciones.
- La comprobación de que alguna tarea, laboratorio, proyecto, examen u otra evaluación es una copia, hará que se apliquen las sanciones que contemplan los reglamentos universitarios.

7. Cronograma.

	SEMANA	TEMA
1	10 – 16 Marzo	Presentación curso y entrega carta al estudiante Tema 1: Introducción Tema 2: Teorías
2	17– 23 Marzo	Tema 2: Teorías Tema 3: Algoritmos en pseudocódigo
3	24 – 30 Marzo	Tema 3: Algoritmos en pseudocódigo
4	31 – 06 Abril	Tema 3: Algoritmos en pseudocódigo Entrega Trabajo de Investigación Exposiciones Trabajo de Investigación
5	07 – 13 Abril	Exposiciones Trabajo de Investigación Feriado 11 abril
6	14 – 20 Abril	Semana Santa
7	21 – 27 Abril	Tema 4: Introducción a la programación Semana Universitaria
8	28 – 04 Mayo	I Examen Tema 4: Introducción a la programación
9	05 – 11 Mayo	Tema 4: Introducción a la programación
10	12 – 18 Mayo	Tema 4: Introducción a la programación
11	19 – 25 Mayo	Tema 4: Introducción a la programación
12	26 – 01 Junio	Tema 4: Introducción a la programación
13	02 – 08 Junio	II Examen Tema: Introducción a la programación orientada a objetos (Clases y objetos, atributos, métodos de instancia, constructores, set y get, uml básico, llamado de métodos entre clases y en la misma clase (modularidad), paso valor y referencia, sobrecarga, static, this, direcciones de memoria de los objetos, referencia e instancia, atributos tipo objeto, variables anónimas)
14	09 – 15 Junio	Tema: Introducción a la programación orientada a objetos
15	16 – 22 Junio	Tema: Introducción a la programación orientada a objetos
16	23 – 29 Junio	Tema: Introducción a la programación orientada a



6		objetos
1 7	30 – 06 Julio	III Examen
1 8	07 – 13 Julio	Entrega Tarea Programada / Comprobación Tarea Programada Entrega de promedios
1 9	14 – 20 Julio	Ampliación

5 Bibliografía

- Norton, Peter. *Introducción a la Computación*, McGraw-Hill, Sexta edición, España, 2006.
- Deitel, Harvey. Deitel, Paul. *Cómo programar en Java*, Pearson Education, Novena Edición, México, 2012
- Wu, Tomas. *Programación en Java*, McGraw-Hillm, Primera Edición, México, 2008
- Prieto, Alberto. Prieto, Beatriz. *Conceptos de Informática*, McGraw-Hill, Primera Edición, 2005.
- Prieto, Alberto. Lloris, Antonio. Torres, Juan C. *Introducción a la informática*. McGraw-Hill, Cuarta Edición, 2006.
- Jamrich, Parson June y Oja, Dan. *Conceptos de Computación*, Thomson, Sexta Edición, 2006.
- Orozco Martha, Chávez María y Chávez Joaquín. *Informática Uno*, Thomson, Primera Edición, 2006.
- López, Leobardo. *Metodología de la Programación Orientada a Objetos*, Alfaomega, Primera Edición, 2006.
- Joyanes, Luis. *Fundamentos de programación*, MacGraw-Hill, Cuarta Edición, México, 2008
- Brookshear, Glenn. *Introducción a la Computación*, Pearson Education, Undécima Edición, España, 2012
- Beekman, George. *Introducción a la Informática*, Pearson Prentice Hall, Sexta Edición, España, 2005