

PROGRAMA CURSO: **IF 1300 INTRODUCCIÓN A LA COMPUTACIÓN E INFORMÁTICA**
I Semestre, 2017

Datos Generales

Sigla del curso: IF-1300
Nombre del curso: Introducción a la Computación e Informática
Tipo de curso: Teórico-práctico
Créditos: 4
Horas lectivas semanales: 8
Requisitos: Ninguno
Correquisitos: Ninguno
Ubicación en el plan de estudio: I Ciclo
Horario del curso: L de 17:00 a 21:00 y V de 17:00 a 21:00
Tutoría: No
Suficiencia: No
Modalidad: Bajo virtual

Datos del Profesor

Nombre: Gabriel Quirós Ramírez
Correo electrónico: gabriel.quiros@ucrso.info
Horas consulta presencial: L de 21:00 a 22:00 y V de 21:00 a 22:00
Horas consulta remota: M de 16:00 a 18:00.
Medio oficial para consulta remota: Hangouts (gabriel.quiros@ucrso.info)

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.

Objetivos

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

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

- Enfrentar al estudiante con la temática de resolución de problemas a través del uso de pseudocódigos.
 - 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.
- 1.4. Introducción a los Sistemas de Información:
 - 1.4.1. ¿Qué es un sistema?
 - 1.4.2. Teoría General de sistemas
 - 1.4.3. Conceptos y clasificación de los sistemas.
 - 1.4.4. Desarrollo de los Sistemas de Información.
- 1.5. Datos, información y procesamiento de la información.

2. Algoritmos, Pseudocódigos y Diagramas de Flujo.

- 2.1. Conceptos y características de los algoritmos, pseudocódigos y diagramas de flujo.
- 2.2. Estructura de los algoritmos en pseudocódigos y diagramas de flujo.
- 2.3. Solución de problemas con pseudocódigos y diagramas de flujo

3. Introducción a la programación

- 3.1. Los lenguajes de programación
- 3.2. Concepto y partes constitutivas de un programa
- 3.3. Elementos básicos de un programa
 - 3.3.1. Tipos de datos
 - 3.3.2. Variables, constantes, expresiones
 - 3.3.3. Operadores aritméticos, unarios, asignación, relacionales y lógicos
- 3.4. Estructuras de control
 - 3.4.1. El flujo de control de un programa y estructura secuencial
 - 3.4.2. Estructuras de selección
 - 3.4.2.1. Simple y doble (si-entonces, si-entonces-sino / if, if-else)
 - 3.4.2.2. Múltiple (según sea, caso de / case)
 - 3.4.2.3. Estructuras de selección anidadas
 - 3.4.3. Estructuras repetitivas (while, for, do while)
- 3.5. Métodos, parámetros y argumentos

- 3.5.1. Definición de métodos
 - 3.5.2. Estructura de métodos
 - 3.5.2.1. Variables locales y visibilidad en el programa
 - 3.5.2.2. Valor de retorno
 - 3.5.2.3. Parámetros en los métodos
 - 3.5.3. Llamada a los métodos desde el programa principal y de otros métodos
 - 3.5.4. Argumentos en la llamada de los métodos
4. Introducción a la Programación Orientada a Objetos
- 4.1. Definición y estructura de Clase y Objeto
 - 4.2. Pilares de la programación orientada a objetos
 - 4.2.1. Abstracción
 - 4.2.2. Encapsulación
 - 4.2.3. Herencia
 - 4.2.4. Polimorfismo
 - 4.3. Métodos:
 - 4.3.1. Constructores
 - 4.3.2. Destrucción
 - 4.3.3. Propiedades (set y get)
 - 4.3.4. Visibilidad de métodos y propiedades (privado, publico, etc)
 - 4.3.5. Llamado de métodos entre clases y en la misma clase (modularidad)
 - 4.4. Direcciones de memoria de los objetos, referencia e instancia
 - 4.5. Introducción a la Seguridad en Código

5. Metodología

El profesor impartirá 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 el profesor, tanto en clase como para resolver en sus casas. Además, el estudiante de manera grupal desarrollará tareas programadas que les será asignadas con antelación.

6. Evaluación

<i>Descripción</i>	<i>Porcentaje</i>
Investigación	5%
Laboratorios	10%
Exámenes cortos / Tareas	10%
Tareas Programadas	15%
I Examen Parcial	15%
II Examen Parcial	20%
III Examen Parcial	25%
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.
- 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, Mediación Virtual. El estudiantado debe matricularse en el curso virtual utilizando la dirección de correo electrónico personal (@ucr.ac.cr). En este sitio podrán encontrar el cronograma del curso actualizado, material digital, asignación y entrega de tareas, foros, laboratorios, calificaciones, etc.
- Para comunicación remota con el profesor(a) se utilizará únicamente el correo electrónico (@ucurso.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 aula virtual 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.

- La comprobación de que alguna tarea, laboratorios, proyecto, examen u otra evaluación es una copia, hará que se apliquen las sanciones que contemplan los reglamentos universitarios.
- Se deben mantener los celulares en modo silencioso o apagado durante las lecciones.

7. Cronograma

Semana 1	Actividades
13 de marzo	Presentación y entrega del programa del curso Tema 1: Introducción
17 de marzo	
Semana 2	Actividades
20 de marzo	Tema 1: Introducción Tema 2: Algoritmos, Pseudocódigos y Diagramas de Flujo
24 de marzo	
Semana 3	Actividades
27 de marzo	Tema 2: Algoritmos, Pseudocódigos y Diagramas de Flujo Revisión de Temas de Trabajo de Investigación
31 de marzo	
Semana 4	Actividades
3 de abril	Tema 2: Algoritmos, Pseudocódigos y Diagramas de Flujo
7 de abril	
Semana 5	Actividades
10 de abril	Semana Santa
14 de abril	
Semana 6	Actividades
17 de abril	I Examen Entrega 1er Avance Trabajo de Investigación Revisión de definición de I Tarea Programada
21 de abril	
Semana 7	Actividades
24 de abril	Tema 3: Introducción a la programación Semana Universitaria
28 de abril	
Semana 8	Actividades
1 de mayo	Feriado
5 de mayo	Entrega I Tarea Programada Tema 3: Introducción a la programación
Semana 9	Actividades

8 de mayo	Tema 3: Introducción a la programación
12 de mayo	
Semana 10	Actividades
15 de mayo	Tema 3: Introducción a la programación Revisión de definición de II Tarea Programada
19 de mayo	
Semana 11	Actividades
22 de mayo	Tema 4: Introducción a la Programación Orientada a Objetos <i>II Examen</i>
26 de mayo	
Semana 12	Actividades
29 de mayo	Tema 4: Introducción a la Programación Orientada a Objetos Entrega y comprobación de II Tarea programada
2 de junio	
Semana 13	Actividades
5 de junio	Tema 4: Introducción a la Programación Orientada a Objetos Entrega y Exposición de Trabajo de Investigación
9 de junio	
Semana 14	Actividades
12 de junio	Tema 4: Introducción a la Programación Orientada a Objetos
15 de junio	
Semana 15	Actividades
19 de junio	Tema 4: Introducción a la Programación Orientada a Objetos Revisión de definición de III Tarea Programada
23 de junio	
Semana 16	Actividades
26 de junio	Tema 4: Introducción a la Programación Orientada a Objetos III Examen
30 de junio	
Semana 17	Actividades
3 de julio	Entrega y comprobación de III Tarea programada Entrega de Promedios
7 de julio	
Semana 18	Actividades
10 de julio	Examen de ampliación
14 de julio	

8. Normativa del Curso

Se rige vamos las normas establecidas por la Universidad de Costa Rica en sus diferentes reglamentos. Se insta a los estudiantes a leer y comprender el Reglamento de Régimen académico estudiantil.

9. Bibliografía

- Brookshear, Glenn. *Introducción a la Computación*, Pearson Education, Undécima Edición, España, 2012
- Deitel, Harvey. Deitel, Paul. *Cómo programar en Java*, Pearson Education, Novena Edición, México, 2012

10. Otras referencias

- Norton, Peter. *Introducción a la Computación*, McGraw-Hill, Sexta edición, España, 2006.
- 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
- Beekman, George. *Introducción a la Informática*, Pearson Prentice Hall, Sexta Edición, España, 2005

