



Programación 1

1. Características generales

Nombre:	Programación 1
Sigla:	CI-0112
Créditos:	4
Horas lectivas:	5
Requisitos:	CI-0110 Introducción a la Computación
Correquisitos:	-
Clasificación:	Curso propio
Ciclo de carrera:	II ciclo, 1er año
Docente:	Mauricio Pandolfi González
Datos de contacto:	Correo electrónico: mauricio.pandolfi@ucr.ac.cr Telegram: @mpandolfigon
Grupo:	9
Semestre y año:	II ciclo 2023
Horario y lugar de clases:	Lunes de 7 a 9:50 a.m., jueves de 7 a 8:50 a.m., Aula IF 306
Consulta:	Miércoles de 6 a 8 p.m. y viernes de 8:30 a 9 a.m. vía Zoom, con previa confirmación (Calendly)

2. Descripción del curso

Este curso fomenta en el estudiante habilidades generales para la resolución de problemas de programación, con énfasis en la etapa de implementación de soluciones. Se usa el paradigma de programación orientado a objetos porque permite solucionar problemas de forma más natural.

3. Objetivos

Objetivo general

El objetivo general del curso es que el estudiante sea capaz de resolver problemas mediante la programación de computadoras utilizando el paradigma de programación orientado a objetos.

Objetivos específicos

Durante este curso cada estudiante desarrollará habilidades para:

1. Explicar el modelo de ejecución de los programas.
2. Analizar problemas y diseñar soluciones básicas orientadas a objetos.
3. Implementar soluciones utilizando el paradigma de programación orientado a objetos.





4. Validar informalmente la ejecución de los programas mediante la definición de un conjunto básico de casos de prueba.
5. Aplicar buenas prácticas de programación.
6. Implementar y manipular estructuras de datos.
7. Implementar algoritmos básicos.
8. Utilizar un ambiente de desarrollo integrado.
9. Corregir errores en programas utilizando un depurador.

4. Contenidos

Objetivo específico	Eje temático	Desglose
8, 9	Herramientas de desarrollo	Herramientas de desarrollo: compilador, depurador, ambiente de desarrollo integrado (IDE).
1	Modelo de ejecución de programas	Rastreo de memoria de programa.
2	Resolución de problemas	Proceso de resolución de problemas, análisis, diseño, implementación, prueba, patrones, técnicas de resolución de problemas: divide y vencerás.
2, 3	Diseño e implementación de soluciones	Conceptualización y definición de clases, atributos de clases, instancias de clases, tipos y variables, parte pública, parte privada.
2, 3	Diseño e implementación de soluciones	Conceptualización e implementación de métodos (sin parámetros, con parámetros de valor, con parámetros de referencia), constructores y destructores.
6	Estructuras de datos	Estructuras de datos estáticas: arreglos de una y más dimensiones.
6	Estructuras de datos	Estructuras de datos dinámicas: lista enlazada y árbol binario.





Objetivo específico	Eje temático	Desglose
3, 5, 7	Entrada y salida de datos	Flujos de entrada y salida de datos. Validación de datos.
7	Algoritmos	Estructuras básicas de control: secuenciación, bifurcación, iteración.
7	Algoritmos	Algoritmos básicos de búsqueda como: secuencial y binaria, y algoritmos de ordenamiento como: burbuja, selección e inserción.
7	Algoritmos	Funciones recursivas tales como: factorial, Fibonacci, multiplicación de enteros, potencia de dos números, máximo común divisor de dos números, Torres de Hanoi; además, recorrido, inserción y borrado sobre árboles binarios.
4, 5, 8, 9	Calidad de programas	Casos de prueba básicos: casos extremos, intermedios, inválidos. Buenas prácticas de programación: nombres significativos, convenciones de escritura de nombres (clases, variables, métodos, constantes), disposición del texto (tabulación, delimitadores de bloques, espacios y líneas en blanco), documentación interna, orden de las declaraciones (campos, constructores, métodos), modificadores de acceso (privado y público), uso de constructores.

5. Metodología

Se sigue una metodología híbrida constructivista-tradicional. De acuerdo con los créditos del curso, en cada semana el estudiantado dedica a este: 5 horas en lecciones y 7 horas en actividades extra-clase.

Las horas de la semana se dividen en 2 lecciones: una de 3 horas lectivas (lunes) y una de 2 horas lectivas (jueves). En las lecciones teóricas, se utilizan recursos audiovisuales para ilustrar el proceso de resolución de problemas y los conceptos de programación. También, se ven ejemplos en la aplicación de los temas desarrollados y se sugieren prácticas adicionales. En las lecciones de práctica, las y los estudiantes ven ejemplos prácticos y resuelven problemas cortos de programación que requieren la





comprensión de los temas. Para ello, pueden contar con el apoyo del profesor del curso y el o la asistente. Puede haber clases híbridas, en donde se mezcle teoría y práctica.

El curso se respalda en un aula virtual por medio de la plataforma Mediación Virtual UCR (www.mediacionvirtual.ucr.ac.cr). Esta es la forma oficial para acceder a materiales del curso – tanto teóricos como prácticos – y se usará como vía de algunas entregas. El o la estudiante es responsable de atender las asignaciones a tiempo, llevar al día la materia y hacer las entregas correspondientes. En el caso de prácticas cortas que se describen más adelante, estas podrán utilizar la plataforma HackerRank, cuya dinámica y asignación se explicará cuando se utilice por primera vez.

El grupo de estudiantes, así como el profesor y su asistente, compartirán comunicaciones por medio de un grupo de Telegram. El enlace para ser parte de este grupo se compartirá en el encabezado del aula virtual. Este puede ser utilizado para consultas, comentarios generales, discusiones sobre la dinámica del curso, y para anuncios generales sobre evaluaciones, metodología o consultas.

La naturaleza del curso demanda que la materia es acumulativa. Por consiguiente, se recomienda mantenerla al día, e ir evacuando dudas a tiempo. Si hubiese dudas específicas sobre cualquiera de las actividades del curso, es importante hablarlo con el profesor oportunamente para la atención en consulta extraclase. Se ofrecen las siguientes modalidades virtuales:

- A) Enviar un mensaje al profesor usando el chat directo de Telegram con la consulta (@mpandolfigon). Esta será la opción más ágil, pero no siempre es suficiente para comprender las consultas o para recibir una apropiada explicación.
- B) Tener una consulta sincrónica por medio de Zoom. Para preguntas más complejas, es la opción recomendada, pues permite profundizar mejor tanto en las preguntas como en las respuestas. Si requiere utilizar esta consulta sincrónica, el o la estudiante debe reservar una cita dentro del horario estipulado. Para ello, se utilizará la plataforma Calendly, y el link se compartirá en el aula virtual. A una sesión de consulta virtual pueden asistir varias personas a la vez. El link de Zoom que será utilizado para la consulta sincrónica se proporcionará en la confirmación de la cita.

Para la resolución de evaluaciones, se podrá consultar material únicamente que sea producto de su trabajo en el curso o que haya sido proporcionado por el profesor (libros, apuntes o código). En cada enunciado, se especificarán los temas vistos en el curso que están relacionados, y la persona debe limitarse a resolverlo con dichos temas. **Queda terminantemente prohibido emplear código que cubra temas no vistos en clase.** La utilización de librerías, funciones, o código relacionado con conceptos no cubiertos hasta ese momento equivaldrá a la anulación del enunciado que se está evaluando.





6. Evaluación

A continuación, se detalla la evaluación del curso.

Rubro	Peso
Exámenes	10%, 25% y 25%
Prácticas cortas y quices	15%
Tareas programadas	5%, 10% y 10%

Se realizarán 3 exámenes de forma presencial y metodología de resolución a mano y en papel. El primer examen tendrá una duración máxima de 2 horas, y el segundo y tercero tendrán 3 horas como duración máxima. El primero valdrá 10%, mientras que el segundo y tercero valdrán 25% cada uno.

Para la asignación de calificaciones en el caso de los exámenes, el profesor podría solicitar al estudiantado que asistan a una cita sincrónica individual. En esta cita, el profesor podría realizar actividades adicionales como entrevistas con efectos en la calificación de este. Si el profesor lo solicita, asistir a esta reunión corta sería requisito para recibir una calificación. Ante problemas de choque de horarios con las citas disponibles, el estudiante debe dirigirse al profesor con al menos 48 horas de antelación del inicio de las citas, para acordar otro momento en el que ambas coincidan.

Las prácticas cortas serán asignadas en clase, siempre con su fecha y hora de entrega, y estas podrán utilizar la plataforma HackerRank. Los quices se realizan en tiempo de clase, y serán anunciados al menos la clase anterior.

La persona que funge como asistente será la encargada de vigilar las prácticas cortas y revisar las tareas programadas, por lo que ante dudas sobre asignación de puntos se debe acudir a ella en primer lugar. Si se considera que no se atendió el problema satisfactoriamente, puede entonces comunicarlo al profesor y él determinará una solución definitiva de acuerdo con los objetivos del curso.

El o la estudiante debe entregar sus trabajos en la plataforma correspondiente antes de la fecha y hora límite estipulada. Ante entregas a destiempo, se establecen las siguientes reglas:

- Para quices y exámenes, no se aceptan entregas fuera de la hora límite designada.
- Para prácticas cortas, se aceptan entregas hasta con una hora de atraso, sin sanción.
- Para las tareas programadas, ante una entrega de h horas tardías se aplicará un castigo de $h^{3/2}$ puntos sobre 100 (puntaje redondeado al entero más próximo). En caso de que la hora no sea exacta, se redondeará el valor de h al entero mayor más próximo.





Otras reglas de evaluación son:

- En cualquiera de las asignaciones, el profesor podría proponer actividades opcionales por crédito extra en la nota del curso. No obstante, este crédito extra estará sujeto a las indicaciones específicas que se den, tales como tiempos de entrega y otros requisitos.
- Todas las evaluaciones son individuales, excepto en casos en que se indique lo contrario de forma explícita.
- La rúbrica de evaluación será siempre indicada en el enunciado. Es responsabilidad del estudiantado asegurarse de entenderla correctamente y evacuar dudas, antes de hacer una entrega.
- Todo código de programación virtual entregado por la o el estudiante debe compilar. Es decir, debe respetar el lenguaje de programación. De lo contrario, será calificado directamente con nota cero.
- En cualquier asignación en que se sospeche de plagio o suplantación de identidad será calificada con nota cero y se aplicará el debido proceso estipulado en el Reglamento de Orden y Disciplina de los Estudiantes de la Universidad de Costa Rica.
- El uso de inteligencia artificial generativa (IAG) no puede usarse para alcanzar el objetivo principal de las actividades del curso. El profesorado se reserva el derecho de permitir o no el uso de IAG, lo cual será indicado en los enunciados de las evaluaciones correspondientes, así como los mecanismos para delimitar el contenido generado y acreditar su fuente.
- Si en alguna evaluación se permitiese de forma explícita en el enunciado el uso de inteligencia artificial generativa (IAG), la persona estudiante asume las responsabilidades de las implicaciones y errores del contenido generado, no la herramienta. La persona estudiante debe ser capaz de explicar, criticar, y defender cualquier contenido que sea presentado para evaluación.





7. Cronograma

A continuación, se detallan las fechas de los hitos más significativos para la evaluación. Estas fechas podrían cambiar debido a eventos circunstanciales, desarrollo del curso, o solicitudes extraordinarias durante el semestre (de profesor o de estudiantado). En caso de desfase, el profesor hará el ajuste necesario y negociará las fechas con las y los estudiantes.

Evento	Fecha
Examen # 1	Jueves 28 de setiembre, de 7 a 9 a.m.
Entrega máxima de tarea programada # 1	Semana 8 (miércoles 4 de octubre)
Examen # 2	Lunes 30 de octubre, de 7 a 10 a.m.
Entrega máxima de tarea programada # 2	Semana 12 (miércoles 1 de noviembre)
Examen # 3	Lunes 27 de noviembre, de 7 a 10 a.m.
Entrega máxima de tarea programada # 3	Semana 16 (jueves 30 de noviembre)
Entrega de promedios provisionales	Lunes 4 de diciembre
Examen de ampliación	Miércoles 13 de diciembre, de 7 a 10 a.m.

8. Bibliografía

- Barnes, David y Kolling, Michael. Programación orientada a objetos con Java usando BlueJ. Quinta edición. Pearson. 2013.
- Deitel y Deitel. Java How To Program: Early Objects. Décima edición. Pearson. 2014.
- Wu, C. Thomas. An Introduction to Object-Oriented Programming with Java. McGraw-Hill. Quinta edición. 2009.
- Ceballos, Francisco Javier. Java2: Curso de Programación. Cuarta edición. Editorial Alfaomega. 2011.

9. Recursos estudiantiles

Para información sobre recursos estudiantiles disponibles en la UCR, incluyendo el Sistema de bibliotecas y la normativa universitaria vigente, favor visitar la página:

<https://www.ecci.ucr.ac.cr/vida-estudiantil/servicios-institucionales-para-estudiantes/guia-de-recursos-estudiantiles-de-la-ucr>

