

Facultad de Ingeniería y Ciencias
Escuela de Informática y Telecomunicaciones

PROGRAMA DE ASIGNATURA
Programación

1. Identificación de la asignatura:

Nombre de la Asignatura: Programación	
Códigos: CIT-1100	Créditos: 5
Duración: Semestral	Ubicación en el plan de estudios: Semestre 1
Requisitos: Admisión	
Sesiones cátedras semanales: 2 cátedras	
Sesiones de Ayudantía: 1	

2. Descripción de la asignatura:

Este curso entrega fundamentos de programación y herramientas básicas de análisis de datos utilizando el lenguaje de programación Python. Los estudiantes aprenderán a automatizar tareas comunes de manejo de datos, manipulación de archivos y visualización de información a través de gráficos simples. Se prioriza el desarrollo del pensamiento lógico y la resolución de problemas aplicados al procesamiento de datos reales.

3. Resultados de Aprendizaje:

1. Identifica la sintaxis del lenguaje Python a través del uso de distintos tipos de datos y estructuras de control que permiten la ejecución de un algoritmo.
2. Selecciona estructuras de datos integradas en el lenguaje Python en forma apropiada para la resolución de problemas básicos.
3. Experimenta con diferentes bibliotecas y estructuras de datos reconociendo su función y aplicación.
4. Aplica diferentes bibliotecas especializadas en el manejo y análisis de datos para la manipulación de archivos de texto planos.
5. Diseña funciones específicas para la resolución de problemas básicos, integrando distintas estructuras de control y estructuras de datos.
6. Participa en equipos de trabajo, planificando, coordinando y ejecutando tareas con liderazgo y responsabilidad, comunicándose efectivamente y elaborando informes técnicos que reflejen procedimientos, resultados y análisis del trabajo realizado.

4. Contenidos:

- **Unidad 1: El computador**
 - Definiciones sobre componentes básicas y operación (memoria, disco, sistema de archivos)
- **Unidad 2: Pseudocódigo y ruteo de variables**
 - Descripción de un problemas en sub-etapas a través de un lenguaje natural

- o Técnicas de resolución de un problema
- **Unidad 3: Programación básica en Python**
 - o Introducción a la programación, herramientas para la ejecución de código.
 - o Tipos de datos: enteros, cadenas
 - o Noción y manejo de variables
 - o Uso de operadores lógicos y aritméticos
 - o Estructuras de control: if, else, for, while
 - o Listas, tuplas, sets y diccionarios: creación, acceso, modificación, operaciones básicas
 - o Manejo de archivos: lectura y escritura de archivos CSV
 - Ejercicios: suma, promedio, búsqueda, máximo, mínimo
 - o Visualización básica de datos
 - Gráficos simples: líneas, entre otros.
- **Unidad 4: Manipulación y transformación de datos**
 - o Funciones y funciones lambda
 - o Uso de lambda para filtrado y transformación de datos
 - o Agregación y resumen de datos
 - o Manejo de datos faltantes y duplicados
 - o Operaciones entre listas: suma, promedio, diferencias
 - o Gráficos básicos con datos agrupados: barras, entre otros.

5. Descripción general del método de enseñanza:

Se contempla la realización de 2 sesiones de cátedra en aula de clases. Además, se dispone de ayudantías en laboratorio, donde se aplicará la técnica de estudio de casos, en las que las materias del curso se irán desarrollando en base a los ejemplos aplicados que en cada sesión se planteen.

Durante el semestre se realizarán evaluaciones en laboratorios de las unidades temáticas vistas hasta la fecha de cada evaluación.

Se utilizará la técnica de estudio de casos y ejercicios aplicados, así como el uso de notebooks colaborativos para promover la práctica continua y el trabajo con datos reales.

El curso será apoyado con recursos digitales, ejemplos interactivos y repositorios disponibles en línea.

Descripción de la metodología	Rol de estudiante en el aula	Actividades autónomas del estudiante
Clases expositivas: Presentación de contenidos teóricos y desarrollo de ejercicios orientados a la comprensión conceptual y la introducción de técnicas.	- Participa activamente, atiende, toma apuntes y realiza preguntas.	- Revisa y estudia previamente el material recibido. - Repasa y consolida los contenidos vistos en clase. - Lee y estudia textos guía y complementarios.
Trabajo de laboratorio	- Desarrolla las tareas	- Repasa y resume los

Descripción de la metodología	Rol de estudiante en el aula	Actividades autónomas del estudiante
computacional: Desarrollo de ejercicios o actividades evaluativas asistidas por computador durante la cátedra o ayudantía, que requieren el diseño o implementación de soluciones.	asignadas, de forma individual o colaborativa. - Realiza preguntas asociadas a la ejecución de las tareas asignadas. - Incorpora la retroalimentación recibida desde el/la docente o ayudante, a sus tareas.	contenidos vistos en aula (cátedra y ayudantías). - Investiga sobre herramientas para la resolución de las tareas asignadas. - Desarrolla las tareas asignadas.

6. Descripción general de la modalidad de evaluación:

Se contempla la realización de evaluaciones parciales (controles, trabajos, prácticas de laboratorio, entre otros), dos pruebas solemnes de igual ponderación y un examen.

Las notas parciales contemplan un mínimo de tres laboratorios y dos controles parciales correspondientes a evaluaciones escritas de los contenidos vistos en clases. El promedio de notas parciales corresponde al promedio simple entre las evaluaciones antes descritas.

La nota de presentación a examen se calcula de la siguiente forma:

Nota de Presentación = (35% Solemne 1 + 35% Solemne 2 + 30% Notas parciales)

A su vez, la nota final del curso está dada por:

Nota Final= 70% Nota de Presentación + 30% Examen.

Según regla general, para aprobar el curso debe tenerse una Nota Final $\geq 4,0$ y para presentarse a Examen se debe cumplir con una Nota de Presentación $\geq 3,5$.

La inasistencia justificada a la Secretaría de Estudio correspondiente a una prueba solemne, implicará reemplazo de su nota con la Nota de Examen. El profesor podrá eximir del examen a aquellos alumnos/as cuyos promedios de notas parciales y de pruebas solemnes sean superiores a 5.0.

Evaluación	Descripción	Tipo de evaluación e instrumentos	Modalidad de retroalimentación	Ponderación
Solemnes	Evaluaciones individuales de carácter formal que integran y aplican los	Prueba escrita con ejercicios de diseño, análisis, implementación y uso de	Retroalimentación grupal mediante revisión de soluciones y discusión de	El promedio corresponderá al 40% de la nota de presentación.

Evaluación	Descripción	Tipo de evaluación e instrumentos	Modalidad de retroalimentación	Ponderación
	contenidos teóricos y prácticos del curso. Se realizan en fechas definidas durante el semestre.	estructuras y algoritmos.	errores frecuentes en clases.	
Controles	Evaluaciones prácticas en algunas sesiones de ayudantía.	Tarea teórica y/o práctica individual de programación con entrega inmediata.	Retroalimentación inmediata del ayudante en la sesión y/o retroalimentación escrita.	El promedio corresponderá al 30% de la nota de presentación.
Laboratorios	Actividades prácticas grupales que requieren la aplicación de los contenidos en problemas de mayor complejidad, con entrega de código funcional.	Desarrollo de código	Retroalimentación escrita sobre el código entregado, con revisión de errores, sugerencias y evaluación del diseño y la implementación.	El promedio corresponderá al 30% de la nota de presentación.
Examen final	Evaluación individual que integra la totalidad de los contenidos teóricos y prácticos del curso. Busca medir el dominio global de las competencias desarrolladas durante el semestre.	Prueba escrita con ejercicios de análisis, diseño e implementación de estructuras de datos y algoritmos.	Retroalimentación individual a través de la revisión de la prueba.	Corresponderá al 30% de la nota final.

7. Planificación de Sesiones:

La planificación de las sesiones dependerá del calendario académico vigente al momento de dictar la asignatura. En cualquier caso, cada docente que la imparta deberá informar a la Escuela (antes del comienzo del semestre) la planificación de sus actividades, evaluaciones, y mecanismos de

diversificación de las mismas.

Esta información será informada al estudiantado durante la primera semana de clases.

8. Bibliografía Básica Obligatoria:

1. "Curso Intensivo de Python" – Eric Matthes
2. McKinney, Wes. *Python for Data Analysis*. O'Reilly Media.
3. Van der Plas, "Python Data Science Handbook: Essential Tools for Working with Data"; O'Reilly Media, 2016.
4. Downey, "Think Python"; Green Tea Press, 2015
5. Apuntes de Cátedra.

Pautas éticas básicas

El aula es un espacio donde los intercambios buscan generar un clima que potencie el aprendizaje, basado en el respeto y el buen trato. Las diferencias, tanto entre estudiantes, como entre estudiantes y docentes, deben abordarse desde este marco de respeto.

La universidad cuenta con dos reglamentos importantes de conocer:

- Reglamento de Convivencia Estudiantil
- Normativa de Prevención y Sanción de Acciones de Discriminación, Violencia Sexual y/o de Género.

Puedes consultar los reglamentos aquí: <https://www.udp.cl/universidad/reglamentos-y-politicas/>

Así también, el plagio es el uso de las ideas o trabajo de otra persona sin el adecuado consentimiento. El plagio puede ser intencional o no. El plagio intencional es el claro intento de hacer pasar el trabajo o ideas ajenas como el suyo propio para su beneficio. El plagio no intencional puede ocurrir si Ud. no conoce el mecanismo adecuado de referenciar la fuente de sus ideas e información. Si no está seguro de los métodos aceptados para referenciar, debería consultar con su profesor, tutor o personal de biblioteca.

El plagio comprobado es una actitud que puede resultar en severas sanciones disciplinarias y/o en la exclusión de la Universidad (Artículo 44, Reglamento del Estudiante de Pregrado).

Elaborado por: Jonathan Frez, Miguel Carrasco.

Fecha revisión: Noviembre de 2025

Fecha vigencia: Marzo de 2026