

# Pensamiento Computacional

Cátedra Camejo

PROGRAMA  
1º CUATRIMESTRE 2023

**.UBA XXI**

# Fundamentación

El pensamiento computacional es una disciplina que ha sido definida como “el conjunto de procesos de pensamiento implicados en la formulación de problemas y sus soluciones, de manera que dichas soluciones sean representadas de una forma que puedan ser efectivamente ejecutadas por un agente de procesamiento de información”, entendiendo por esto último a un humano, una máquina o una combinación de ambos.

Reconoce antecedentes en trabajos de la Carnegie Mellon University de la década de 1960 y del Massachusetts Institute of Technology de alrededor de 1980, aunque su auge en la educación superior llegó con la primera década del siglo XXI.

Las herramientas básicas en las que se funda el pensamiento computacional son la descomposición, la abstracción, el reconocimiento de patrones y la algoritmia. Está ampliamente aceptado que estas herramientas no sirven solamente a los profesionales de Ciencias de la Computación y de Informática, sino a cualquier persona que deba resolver problemas, con lo cual el pensamiento computacional deviene en una técnica de resolución de problemas. Actualmente, los y las profesionales de la Ingeniería requieren de una capacidad analítica que les permita resolver problemas, y en ese sentido el pensamiento computacional se convierte en un soporte invaluable de esa competencia (cada vez más las ciencias de la computación y la informática constituyen una ciencia básica para todas las ingenierías).

Si bien el pensamiento computacional no necesariamente requiere del uso de computadoras, la programación de computadoras se convierte en su complemento ideal. En primer lugar, porque permite comprobar, mediante la codificación de un algoritmo en un programa, la validez de la solución encontrada al problema, de manera sencilla y prácticamente inmediata. En segundo lugar, porque la programación incentiva la creatividad, la capacidad para la autoorganización y el trabajo en equipo. En tercer lugar, porque la programación constituye un recurso habitual del trabajo en el campo profesional de la ingeniería.

## Objetivos

El objetivo general de la asignatura es que los/as estudiantes adquieran habilidades de resolución de problemas de ingeniería mediante el soporte de un lenguaje de programación multiparadigma.

En tal sentido se pretende que el alumno logre los siguientes objetivos específicos:

- Descomponer problemas complejos en problemas más sencillas utilizando la abstracción como mecanismo fundamental.
- Proveer soluciones concretas valiéndose de las características que provee el lenguaje.
- Utilizar correctamente las estructuras de control y tipos de datos provistos por el lenguaje de programación.

- Crear y utilizar funciones para tomar, procesar información y devolver un resultado como bloques unitarios para la resolución de problemas.
- Manejar información en distintos formatos, utilizar mecanismos de intercambios de datos y procesar dicha información utilizando las herramientas provistas por el lenguaje de programación.
- Identificar y utilizar las bibliotecas adecuadas y sus funciones predefinidas para aplicarlas en la resolución de los distintos tipos de problemas de ingeniería.
- Implementar algoritmos de resolución de problemas de ingeniería.

## Contenidos

### Unidad 1: Introducción a la algoritmia y a la programación

Metodología de resolución de problemas. Concepto de algoritmo. Concepto de Programa. Concepto de lenguaje de programación. El lenguaje de programación Python. Sintaxis y gramática de Python. El intérprete de Python. Variables: concepto y ejemplos.

### Unidad 2: Tipos de datos, expresiones y funciones

Tipos de datos simples. Números: entero, real. Cadenas. Booleano. Funciones: Concepto. Comentarios. Operadores: Operadores Lógicos, Relacionales. Expresiones. Concepto de Expresión. Expresiones booleanas. Funciones. Parámetros. Tipo y valor de retorno.

### Unidad 3: Estructuras de control

Estructuras de control: Concepto. Estructuras condicionales: if, else, elif. Estructuras de control iterativas: while, for, range.

### Unidad 4: Tipos de datos estructurados

Tipos de datos estructurados. Secuencias: tuplas y listas. Listas. Operaciones con listas. Mutabilidad. Diccionarios.

### Unidad 5: Entrada y salida de información

Manejo de archivos. Manejo de errores.

### Unidad 6: Bibliotecas

Bibliotecas: Concepto. Uso de bibliotecas. Import. Introducción al análisis exploratorio de datos con Pandas. Visualizaciones básicas con Matplotlib.

## Estrategias de enseñanza

La asignatura estará planteada desde un enfoque activo, con énfasis en el análisis y la resolución de problemas. Los aportes teóricos cumplirán la función de apuntalar la realización de actividades prácticas, que serán de dificultad creciente.

Los y las estudiantes contarán con apuntes de cátedra y con guías para la resolución de actividades, así como tutorías en línea, en las que se trabajarán los conceptos principales de las distintas unidades del programa, y se comentarán los apuntes de cátedra.

En cuanto a la comunicación entre estudiantes y docentes, ésta se desarrollará a través de distintos foros creados en el campus virtual. Cada una de las sesiones contará con un espacio disponible para resolver dudas, acercar consultas y problemas que puedan surgir durante el proceso de estudio y resolución de las actividades, así como para intercambiar con los docentes de la cátedra y los compañeros acerca del uso de las aplicaciones propuestas para la resolución de las actividades.

Con respecto a las evaluaciones, en el primer parcial se evaluarán las Unidades 1, 2, 3 y 4; mientras que, en el segundo parcial, se evalúan las unidades restantes.

## Evaluación

Las **materias se aprueban** por promoción directa o por medio de un examen final.

**Por promoción directa (sin examen final):** el estudiante deberá tener aprobadas las dos instancias parciales de evaluación con un promedio no inferior a 7 puntos en una escala de calificaciones de 0 a 10.

**Por promoción indirecta (con examen final):** el estudiante que haya obtenido como promedio de las dos instancias parciales de evaluación una calificación comprendida entre 4 puntos y menos de 7 puntos deberá rendir un examen final que será aprobado con un mínimo de 4 puntos en una escala de 0 a 10. Si el estudiante obtuviera una calificación inferior a 4 puntos en esta instancia de evaluación será considerado reprobado. El período de validez de la cursada es de tres turnos de exámenes finales consecutivos, incluido el inmediatamente posterior a la finalización de la cursada.

En los casos en los que fuere necesario expresar el número entero del promedio de notas parciales, se aplicará el número entero superior si la fracción fuere de 0,50 puntos o más y el número entero inferior si fuere de 0,49 o menos; cuando la nota fuese de 3,01 a 3,99, se colocará 3 puntos, de acuerdo con lo establecido en la Resolución (CS) Nº 4994/93.

# Bibliografía

## Unidad 1

Arriazu, A. (2023). Apunte de Cátedra. "Conceptos básicos en programación y pensamiento computacional". UBA XXI, Universidad de Buenos Aires

## Unidad 2

Arriazu, A. (2023). Apunte de Cátedra. "Sentencias básicas y datos simples". UBA XXI, Universidad de Buenos Aires

## Unidad 3

Arriazu, A. (2023). Apunte de Cátedra. "Ciclos y rangos". UBA XXI, Universidad de Buenos Aires

Arriazu, A. (2023). Apunte de Cátedra. "Condicionales". UBA XXI, Universidad de Buenos Aires

## Unidad 4

Arriazu, A. (2023). Apunte de Cátedra. "Recursos con listas". UBA XXI, Universidad de Buenos Aires

Arriazu, A. (2023). Apunte de Cátedra. "Secuencias, tuplas y listas". UBA XXI, Universidad de Buenos Aires

Arriazu, A. (2023). Apunte de Cátedra. "Diccionarios". UBA XXI, Universidad de Buenos Aires

## Unidad 5

Arriazu, A. (2023). Apunte de Cátedra. "Archivos". UBA XXI, Universidad de Buenos Aires

## Unidad 6

Bulacios, J. P. (2023). "Introducción a Pandas"

Bulacios, J. P. (2023). "Introducción a Matplotlib"

## Bibliografía complementaria de la asignatura:

AA.VV. (2018). Algoritmos y Programación I: Aprendiendo a programar usando Python como herramienta. (Archivo PDF)

Bahit, E. (2020). Python para principiantes, Oxford RC

Severance, C. (2020). Python para Todos: Explorando datos con Python 3 (Archivo PDF)