

AÑO ACADÉMICO 2020

DEPARTAMENTO: **MATEMÁTICA**

PROGRAMA DE CÁTEDRA: **MÉTODOS COMPUTACIONALES EN INGENIERÍA I**

OBLIGATORIA/OPTATIVA: **OBLIGATORIA**

CARRERA/S: **ING. ELECTRÓNICA, ING. MECÁNICA, ING. ELÉCTRICA, ING. CIVIL, ING. EN PETROLEO e ING. QUÍMICA.**

ÁREA: **CIENCIAS BÁSICAS**

ORIENTACIÓN: -

PLAN DE ESTUDIOS – ORD. N°: **ING. ELECTRÓNICA: 802/97, 481/00, 536/00, 1005/05, 1069/06, 1537/14, 977/17. ING. MECÁNICA: 806/97, 510/00, 536/00, 627/00, 762/05, 1046/05, 1069/06, 192/10, 1537/14. ING. ELÉCTRICA: 807/97, 536/00, 627/00, 482/04, 1069/06, 1375/14, 1578/14. ING. CIVIL : 805/97, 536/00, 1069/06, 192/10, 1537/14. ING. EN PETROLEO: 804/97, 534/00, 536/00, 627/00, 940/01, 1069/06, 192/10, 1537/14. ING. QUÍMICA: 803/97, 536/00, 627/00, 1069/06, 192/10, 1537/14, 1563/14**

CARGA HORARIA SEMANAL: **6HS**

CARGA HORARIA TOTAL: **96HS**

RÉGIMEN: **CUATRIMESTRAL**

CUATRIMESTRE: **PRIMERO**

OBLIGATORIA

EQUIPO DE CÁTEDRA:

Apellido y Nombres	Cargo
CEJAS BOLECEK, NÉSTOR RENÉ	PAD-3
VILUGRON, RENE MARTÍN	AYP-3

ASIGNATURAS CORRELATIVAS (S/Plan de Estudios):

**REGULARIZADA (00802/97, 00803/97, 00804/97, 00805/97, 00806/97 y 00807/97):
ÁLGEBRA Y GEOMETRÍA II.**

* La materia que se exige regularizada para cursar deberá estar aprobada para **aprobar o promocionar** MÉTODOS COMPUTACIONALES EN INGENIERÍA I.

1. FUNDAMENTACIÓN:

El modelado computacional permite reproducir con gran fidelidad la dinámica de sistemas complejos reales. Al explorar en forma metodológica el espacio de los

parámetros del problema se pueden inferir leyes del comportamiento del sistema que se emplean para: tomar decisiones y reducir costos y mejorar la eficiencia de los productos, entre las aplicaciones más destacadas. Además, la simulación de modelos se utiliza en modalidades de capacitación virtual cuyas situaciones reales análogas serían muy riesgosas o de un elevado costo.

El curso introduce el campo de la informática, o computación, haciendo una revisión de las ideas más revolucionarias que llevaron a la creación de la computadora digital que hoy conocemos. Varias de estas ideas permitieron realizar los avances más importantes en el campo de las Neurociencias, la Biología molecular, las Telecomunicaciones y también en desarrollo de nuevas tecnologías que constantemente cambian nuestro modo de percibir la realidad.

En las unidades siguientes se estudia la organización estructurada de las computadoras, dando una descripción detallada de las partes fundamentales de la computadora digital y de su funcionamiento interno. Además, se estudiarán los componentes básicos y los aspectos fundamentales de un sistema, particularizando estos conceptos al sistema operativo LINUX.

Finalmente, las dos últimas unidades tienen como objetivo introducir al alumno en el pensamiento computacional con el objeto de que pueda escribir programas y comprender como es el ciclo del desarrollo de software. Las herramientas y conocimientos computacionales le permitirán al alumno abordar otros lenguajes de programación y complementado con cursos de matemática adecuados, continuar ampliando sus conocimientos en temas más avanzados y de gran interés en las ingenierías como: el Cálculo numérico, la Computación visual y la Inteligencia artificial, entre algunos de los más relevantes.

2. OBJETIVOS:

Se pretende que el alumno:

- Comprenda el valor y la importancia de la información y las tecnologías en nuestra sociedad.

- Adquiera conocimientos de la organización estructurada de las computadoras con la finalidad de comprender cómo se llevan a cabo las operaciones de cálculo y el procesamiento de la información.

- Adquiera conocimientos básicos para evaluar un sistema general: componentes básicos, criticidad, modularidad y testeo.

- Adquiera habilidades básicas de programación: diagramas UML y el uso de un lenguaje base (*Python*).

- Utilice *git* como herramienta colaborativas en línea.

- Aprenda a resolver problemas con la computadora y a hacer que las computadoras realicen la tarea que se le requiera.

3. CONTENIDOS SEGÚN PLAN DE ESTUDIOS:

Introducción a la informática. Arquitectura de computadoras. Nociones de sistemas. Sistemas operativos. Introducción a la programación. Análisis de sistemas. Técnicas de programación. Lenguaje de alto nivel.

4. CONTENIDO PROGRAMA ANALÍTICO:

Unidad 1: Introducción a la Informática

Introducción a la informática. Representación de la información. Teoría de la información. Sistemas de representación de la información. Análogo vs. digital. Definición de Tecnología. Evolución de las computadoras digitales. Ley de Moore. Conceptos básicos de las nuevas tecnologías de: Redes sociales, Blockchain, Inteligencia Artificial, Realidad Virtual y Aumentada y almacenamiento de información en la nube.

Unidad 2: Arquitectura de las computadoras

Introducción a la Organización estructuradas de las computadoras. Unidad Central de Procesamiento: organización y funcionamiento. Ejecución de instrucciones. Paralelismo y multiprocesos. Memoria y direcciones. Ordenamiento de bytes. Jerarquía de memorias y tiempos de acceso.

Unidad 3: Introducción a los sistemas

Nociones de sistemas. Componentes básicos, criticidad, modularidad y testeo. Nociones de diagramas UML. Sistemas operativos: procesos y threads, memoria, sistema de archivos, entradas y salidas de datos, virtualización y la nube. Caso de estudio UNIX, LINUX y ANDROID.

Unidad 4: Introducción a la programación

Definición de computo, instrucción y programa. Estructuras de datos. Control de flujo y Ramificación. Descomposición, abstracción y funciones. Diagrama de actividades (UML). Aspectos generales de los lenguajes de programación.

Unidad 5: Introducción al lenguaje de programación de Python

Introducción a Python. Objetos y tipos de variables. Conversiones y Casteo. Abstracción de Expresiones. Operaciones simples. Control de flujo y ramificación en Python. Manejo de lectura y escritura de archivos. Testeo y debugging de código.

5. BIBLIOGRAFÍA BÁSICA Y DE CONSULTA:

- 1) Tanenbaum, Andrew S. Organización de computadoras: un enfoque estructurado. Pearson educación, 2000. ISBN-10: 9701703995
- 2) Abraham Silberschatz, Greg Gagne, Peter B. Galvin - Operating System Concepts-Wiley (2018).
- 3) McQuaid, Mike. *Git in practice*. Manning, 2015.
- 4) Sylvia Langfield, Dave Duddell - Cambridge International AS and A Level Computer Science Coursebook - Cambridge University Press (2016).
- 5) Martina Seidl, Marion Scholz, Christian Huemer, Gerti Kappel. UML at Classroom: An Introduction to Object-Oriented Modeling. Springer (2014)

Se sugiere la utilización de la bibliografía mencionada y se trabajará con guías de trabajos prácticos y lecturas adicionales.

6. PROPUESTA METODOLÓGICA:

Los temas de la materia se desarrollarán tanto en el aula, donde se dará un abordaje teórico y práctico, y en el laboratorio de informática, donde se brindarán las herramientas computacionales y los trabajos prácticos a resolver. La práctica en el laboratorio de informática está focalizada en el desarrollo del pensamiento crítico del alumno orientado a cuestionar constantemente las nuevas tecnologías de la informática. Además en las unidades 4 y 5 se orientarán las prácticas en el desarrollo del

pensamiento computacional del alumno, esencial para conseguir las habilidades de programación que se requieren en el curso.

Para introducir a los alumnos en el mundo de la programación se utilizará el lenguaje *Markdown* para crear informes en forma individual y colectiva. Este lenguaje permite crear texto enriquecido sin utilizar un editor de texto. Los programas se escribirán en el lenguaje de programación interpretado de alto nivel *Python*. El lenguaje Python ha sido ampliamente adoptado en diversas universidades del mundo y en la industria del software. Esto se debe a su intuitiva y fácil utilización para la escritura de código de computadora y por presentar una gran cantidad de bibliotecas del estado del arte de cálculo numérico, procesamiento de la información y visualización. Con el fin de que los alumnos adquieran experiencia en la utilización de herramientas colaborativas en línea, empleadas actualmente en la industria del software, se propondrán actividades de elaboración de informes y de código en grupos en forma remota.

La estructura general de las clases tendrá el siguiente esquema:

Para las unidades 1, 2 y 3:

- 1) Introducción de conceptos fundamentales por parte del docente
- 2) Presentación y discusión de algún tema relacionado a las nuevas tecnologías que involucre algunos de los conceptos explicados
- 3) Espacio para realizar consultas, conjeturas y responder dudas acerca del tema presentado y orientación en la discusión objetiva sobre los usos e implicancias de las tecnologías.
- 4) Asignación de material y orientación para que el alumno elabore un trabajo de investigación corto sobre la temática
- 5) Selección de una parte del material elaborado por el alumno para exponer oralmente en clase y discutir.
- 6) En la unidad 2 se enseñará a los alumnos a utilizar la herramienta de control de versiones git para realizar trabajos colaborativos simultáneamente.

Unidades 4 y 5 (Programación):

- 1) Introducción de conceptos fundamentales por parte del docente
- 2) Presentación de herramientas de programación y orientación en el pensamiento computacional.
- 3) Espacio para realizar consultas, conjeturas y responder dudas acerca del tema presentado y de las herramientas de programación.
- 4) Asignación para la clase siguiente de material bibliográfico y ejercicios prácticos relacionados con los conceptos desarrollados.
- 5) Profundización y resolución de los ejercicios en computadora por parte de los alumnos en forma individual y grupal.

7. EVALUACIÓN Y CONDICIONES DE ACREDITACIÓN:

ALUMNOS REGULARES. Se considerará regularizado el cursado de la materia con la aprobación de dos parciales escritos e individuales con nota de cada parcial igual o superior a 4 (cuatro). En caso de desaprobado alguno de los dos parciales, habrá un recuperatorio por parcial. En caso de no aprobar alguno de los exámenes en esta

segunda instancia el alumno quedará libre en la materia. Al regularizar la asignatura el alumno aprobará la materia al obtener una nota equivalente o superior al 4 (cuatro) en un examen final escrito.

ALUMNOS PROMOCIONALES. Se considerará promocionada la materia con la aprobación de los dos parciales individuales y escritos, la entrega en tiempo y forma de todas las prácticas y la presentación de exposiciones a lo largo del cursado. Todas estas instancias deberán ser aprobadas con una nota igual o superior a 8 (ocho). La desaprobación de un parcial o exposición, la falta de entrega de alguna de las prácticas en tiempo y en forma implicarán la pérdida automática de la posibilidad de promoción. Al aprobar la promoción el alumno quedará eximido de rendir el examen final. La nota final se obtiene del redondeo entero superior (según lo dispuesto en la Ordenanza 273/19, artículo 33) del valor obtenido computado como: $0.15 \times (\text{promedio de la nota de exposiciones}) + 0.25 \times (\text{promedio de notas de las prácticas}) + 0.3 \times (\text{nota del primer Parcial}) + 0.3 \times (\text{nota del segundo Parcial})$.

ALUMNOS LIBRES. Los alumnos en esta situación deberán rendir el examen libre en las mismas fechas de examen final. Esta instancia consiste en la evaluación de un examen escrito práctico y una instancia de exposición oral. En el examen práctico se evaluará la implementación de algoritmos en Python, la utilización de la herramienta de git y la resolución de algún problema utilizando la computadora. Esta instancia se deberá aprobar con una calificación igual o superior a 4 (cuatro) para pasar a la siguiente instancia de examen. En la instancia de exposición oral el alumno deberá desarrollar el contenido que le sea solicitado y responder preguntas conceptuales de cualquier temática abordada en la materia. Esta instancia se aprueba también con una nota igual o superior a 4 (cuatro). En caso de aprobar ambas instancias la nota final se calcula a partir del promedio de estas dos notas.

8. DISTRIBUCIÓN HORARIA:

TEORÍA Y PRÁCTICA: MARTES Y JUEVES DE 19:00 A 20:30HS

PRÁCTICA: LUNES DE 18:00 A 21:00HS

HORAS DE TEORÍA: 3

HORAS DE PRÁCTICA: 3

9. CRONOGRAMA TENTATIVO:

Fechas de desarrollo de temas, actividades y evaluaciones.

MES	UNIDAD
Marzo	I
Abril	II y III
Mayo	IV
Junio	V

Primer Parcial (y fecha de entrega de los prácticos 1, 2 y 3 para promocionar):
4 de Mayo de 2020

Recuperatorio Primer Parcial: 18 de Mayo de 2020

Segundo Parcial (y fecha de entrega de los prácticos 4 y 5 para promocionar):
22 de Junio de 2020

Recuperatorio Segundo Parcial: 2 de Julio de 2020



René Cejas Bolecek

PROFESOR

CONFORMIDAD DEL DEPARTAMENTO



Mg. ALFONSO AGUILAR
Secretario Académico
Centro Regional Univ. Bariloche
Univ. Nacional del Comahue

CONFORMIDAD DEL CENTRO REGIONAL UNIVERSITARIO BARILOCHE