

PROGRAMA de Elementos de Programación y Lógica

Carreras:

- Tecnicatura en Programación Informática
- Licenciatura en Informática
- Licenciatura en Bioinformática

Asignatura: Elementos de Programación y Lógica (EPyL)

Núcleo: Ciclo Introductorio

Equipo Docente:

Lic. Silvina Busto, Lic. Silvana La Chiesa, Ing. Erica Wronsky, Lic. Eugenio Cálcena, Lic. Cristián Marchionne, Lic. Natalia Sabella, Anl. Sist. Warmi Guercio, Ing. Juan Carlos Czerwien, Tec. Prog. Mariano Alvarez

Coordinadora: Ing. Erica Wronsky

Asignaturas previas necesarias para favorecer el aprendizaje: Sin requisitos previos

Objetivos Generales:

Fundar los primeros pasos de las personas en tanto su rol como estudiantes en la universidad, propendiendo a su formación en actitudes, prácticas y competencias propias del contexto educativo universitario.

Acercar algunas problemáticas que forman parte del contexto en el que se inscribe la programación informática, trabajando sobre conceptos básicos de programación y de lógica.

Objetivos Específicos:

Se busca que cada estudiante sea capaz de:

- Entender la función de una computadora en la ejecución de un programa.
- Comprender el concepto de lenguaje, identificar sus partes y usos.
- Comprender conceptos básicos de lógica.
- Identificar el lenguaje que se utiliza en los formalismos de la lógica.
- Entender y manejar conceptos básicos de la programación.
- Comprender enunciados de problemas en programación.
- Resolver problemas de programación a través de la búsqueda de estrategias de subdivisión en problemas más pequeños.

- Entrenar capacidades de abstracción para resolver problemas simples, a partir de ejemplos concretos.
- Desarrollar habilidades de trabajo autónomo y grupal.
- Desarrollar capacidades que permitan adaptarse a diferentes situaciones de la vida universitaria.

Contenidos mínimos:

- Lenguajes de marcado: concepto, definiciones básicas, aplicaciones. Casos de estudio Markdown, HTML.
- Representación de información, mediante comandos desarrollados exclusivamente para la ejercitación de niveles de abstracción.
- Resolución de problemas y modelado de situaciones específicas, a través algoritmos de programación.
- Lógica proposicional: nociones introductorias.
- Lógica de primer orden: nociones introductorias.

Carga horaria semanal:

Cuatro horas semanales distribuidas en dos clases de dos horas (o su equivalente), más una hora semanal de consulta (opcional). Se recomiendan cuatro horas semanales de estudio y resolución de problemas fuera de la cursada.

Programa analítico:

Unidad 1: INFORMÁTICA BÁSICA Y LENGUAJES

- Introducción a la materia. Tipos de archivos (documentos, binarios). Extensiones de archivos. Estructura de carpetas. Rutas de acceso. Definición Hardware y Software. Relación entre Hardware y Software.
- Definición de lenguajes de programación. Comparación de diferentes lenguajes. Lenguajes de Mercado. Utilidad. Uso.

UNIDAD 2: LÓGICA

- Introducción a la lógica: razonamientos, tipos.
- Conectivas lógicas, semántica y sintaxis práctica. Tablas de verdad.
- Lógica proposicional. Proposición atómica. Valor de verdad.
- Proposiciones compuestas. Identificación de proposiciones en razonamientos.
- Razonamientos deductivos. Formalización mediante lenguaje de la lógica proposicional.
- Lógica de primer orden. Individuo. Propiedad. Propiedades simples y compuestas. Relaciones. Cuantificadores.

UNIDAD 3: PROGRAMACIÓN

- Qué es la programación, robot humano y Lightbot.
- Introducción a programación (Documentar, propósito y precondition).
- Definición del lenguaje didáctico QDraw.
- División en subtareas.
- Repetición simple.
- Alternativa condicional.
- Contextos de programación. Diferentes conjuntos de instrucciones.

Bibliografía:

- "Bases Conceptuales de la Programación", Pablo E. "Fidel" Martínez López
- "Actividades para aprender a programar", Pablo Factorovich, Federico Sawady O'Connor (Volumen 1, Octubre 2015).
- "Introducción a la Lógica". Copi, I. Buenos Aires: Eudeba, 2009, cap. I y II
- "Deducción y representación: Una introducción a la lógica de primer orden". Legris, J. Buenos Aires, Economizarte, 2001, cap. 1.

Organización de las clases:

La asignatura cuenta con una modalidad presencial de clases teórico-prácticas. Se organiza en dos grandes bloques dentro de los cuales entre cada unidad se realiza una secuencia de integración de los conceptos vistos, en donde se remarcan las limitaciones alcanzadas y la necesidad de incorporar otros conceptos y herramientas para resolver los nuevos problemas planteados. En el primer bloque se trabaja sobre conceptos introductorios de lógica (proposicional y de predicados) como por ejemplo: razonamientos, conectivas lógicas (sintaxis, tablas de verdad), cuantificadores, proposiciones atómicas, etc. En un segundo bloque se tratan conceptos de programación como por ejemplo: autómatas, instrucciones, lenguajes, propósitos y preconditiones, división en subtareas, repetición, alternativa condicional; utilizando el lenguaje didáctico QDraw y ejercicios de contexto específico (conjuntos de instrucciones adaptados a un problema particular).

Las clases presentan una metodología híbrida, entre exposiciones orales dialogadas, así como también clase invertida, en las cuales se propone una lectura previa de contenidos (señalados y guiados por docentes) para luego discutir los conceptos vistos, dudas o interrogantes. Sin importar la metodología implementada, se remarcan los objetivos de la clase, qué se espera por parte de cada estudiante del curso, así como las principales definiciones y conceptos. Se trabaja verificando la aplicación de los conceptos teóricos a través de preguntas, foros de discusión, ejercicios prácticos y discusión de situaciones concretas.

Adicionalmente se cuenta con una guía de ejercicios prácticos (TP) para aplicar los conceptos teóricos vistos en cada eje temático. Estas guías constan de un conjunto de ejercicios agrupados según diferentes temáticas. Las mismas se encuentran listadas en la sección de Trabajos prácticos.

Frecuentemente se pide a grupos de estudiantes que expongan la resolución de ciertos ejercicios seleccionados para lograr una corrección grupal en plenario, en la que participe toda la clase.

Se utiliza el pizarrón como principal recurso didáctico para el dictado de la clase, que se complementa con proyector para mostrar material que pudiese resultar complejo o poco eficaz de reproducir en el pizarrón. Se utilizan recursos virtuales (campus, listas de correo, carpetas compartidas en la nube, etc) para organización de material de lectura y multimedia, cronograma, consultas de resolución de ejercicios y evaluaciones.

Trabajos prácticos

Los trabajos prácticos cuentan con diferentes tipos de actividades, a saber:

- Iniciales, donde se plantea la necesidad de un concepto nuevo o limitación de uno aprendido. Son actividades muy simples y de secuencia corta.
- De desarrollo, son actividades o problemas para analizar y resolver utilizando los conceptos trabajados en la teoría, presentan diferentes escenarios e incorporan complejidad a medida que avanzan con la guía.
- De integración, que buscan trabajar con la interrelación de varios conceptos y herramientas al mismo tiempo, para interpretar, comprender y resolver problemas más complejos.
- De autoevaluación, parcialitos que simulan el momento de evaluación para ayudar a medir tiempos y desempeño autopercebido de cara a la evaluación parcial. Se trabaja con cuestionarios que resaltan los principales conceptos teóricos vistos, junto con una aplicación práctica.

Unidad / TP	Objetivos / Contenido
1. Informática Básica	<p>El objetivo de estas actividades es reforzar los conceptos básicos de informática vistos en la teoría, al mismo tiempo invitar a la investigación en otras fuentes, que se discuten .</p> <p>En estas actividades se ven las diferencias entre hardware y software, la investigación sobre el uso y concepto de software libre y las libertades que implica comparado con el software privativo. Manejo e identificación de los distintos tipos de archivos, en especial los .txt y el uso de editores de texto, que son utilizados luego en la unidad 3 de programación.</p>
2.1 Lógica - Conectivas y Tablas de Verdad	<p>El objetivo de esta guía es lograr reconocer diferentes tipos de razonamientos que presenta la lógica (inductivos / deductivos), y comprender la importancia de los razonamientos deductivos dentro del campo de las ciencias formales.</p> <p>Se busca identificar y distinguir diferentes conectores lógicos, entendiendo a la lógica como un lenguaje de descripción de un universo.</p>

	<p>Se trabaja con la descripción de información en contexto, para, a través del lenguaje y conceptos lógicos poder lograr su interpretación y transformación en distintos formatos haciendo énfasis en la interrelación de los datos. Además se presenta el lenguaje lógico como un generador de preguntas, para lograr obtener y decodificar información de diferentes dominios y formatos. La idea de elemento de denotación de un valor de verdad y conjuntos binarios de valores de verdad, qué son, cómo se aplican y el resultado de conectarlos con el uso de los conectores. Estas actividades se orientan también a aprender a leer, generar e interpretar tablas de verdad de conectores simples (conjunción, disyunción, negación, condicional).</p>
2.2 Lógica - Proposicional	<p>El objetivo de esta práctica es comprender qué es una proposición, cómo se representa en el lenguaje de la lógica proposicional, es decir mostrar el razonamiento como un conjunto de proposiciones relacionadas, para lograr la identificación de sus partes y la distinción de las premisas y de la conclusión. Por último, a partir de un razonamiento dado, en el lenguaje natural o en forma lógica, demostrar su validez interpretando esta información desde la tabla de verdad.</p> <p>A medida que se avanza en las actividades, se busca integrar los conceptos de proposición, razonamiento, diccionario, validez de un razonamiento. Pudiendo transformar un razonamiento desde el lenguaje natural al lenguaje de la lógica proposicional y viceversa e incluso demostrar si se trata de un razonamiento válido o no.</p>
2.3 Lógica - Predicados	<p>El objetivo de esta práctica es comprender la limitación del lenguaje proposicional ante ciertos tipos de razonamientos, entendiendo el análisis más profundo que se realiza en las proposiciones haciendo uso de los conceptos y lenguaje de la lógica de predicados, proponiendo un nuevo nivel de análisis de la información, con énfasis en las relaciones que se generan entre los elementos del dominio.</p> <p>En esta práctica la secuencia de ejercicios busca que se reconozcan los elementos básicos dentro del lenguaje (individuo, propiedad, relación, parámetro y dominio) introduciendolos en etapas como conceptos fundamentales del descripción de universos.</p> <p>Luego, que se pueda traducir proposiciones y sus relaciones a través de los conectores, del lenguaje natural al lenguaje de la lógica de predicados.</p> <p>Por último se introducen cuantificadores (universal, existencial, existencial negado), que permiten lograr nuevas interpretaciones respecto a las relaciones entre los individuos de un dominio.</p>
2.4	<p>Esta práctica integral tiene como objetivo que repasar los conceptos</p>

Lógica - Repaso	vistos de la unidad de lógica que serán evaluados, aprender a leer y comprender los enunciados del parcial detenidamente, organizar el tiempo dedicado a cada ejercicio en el parcial y detectar los temas o conceptos que más dificultades traen al grupo particular de estudiantes, para dedicarles más tiempo de práctica.
3.1 Programación - Primeros autómatas	<p>Esta práctica tiene como objetivo comprender el concepto de autómata y algoritmo (aplicado desde la generalización y la descomposición).</p> <p>Se presentan diferentes actividades en donde se deberá señalar los pasos necesarios para resolver un problema dado. Se espera que se pueda identificar que esos pasos tienen una secuencia determinada, que pueden existir pasos innecesarios y pasos erróneos que invalidan la solución planteada.</p> <p>A partir de analizar un problema y determinar los pasos para su resolución, se espera poder identificar claramente el propósito y la precondition de cada paso para lograr el objetivo buscado.</p>
3.2 Programación - Sintaxis estricta y División de tareas	<p>El objetivo de esta práctica es comenzar a resolver problemas utilizando el lenguaje didáctico de sintaxis estricta QDraw y las primeras estructuras de control: programa y procedimientos. Además, reflexionar sobre la importancia de la documentación en el código, pensando la programación como comunicación tanto con autómatas como con personas.</p> <p>Los ejercicios de tipo inicial son conceptuales, trabajando en pseudocódigo y haciendo énfasis en el algoritmo y la solución en pensamiento computacional. Se comienza con ejercicios prácticos sencillos, que luego se incorporan en problemas más complejos, de forma de reutilizar las soluciones planteadas.</p> <p>Se presentan varias actividades que van complejizando el nivel del problema, hasta llegar a los casos en donde se espera aplicar la división de tareas o método Top down, haciendo uso de la primera estructura de control dada (procedimientos). Se comienza a tratar la importancia de Comunicar en el programa. Por último se trabaja la idea de sintaxis estricta, diferencia entre programa y procedimientos, ventajas y beneficios del uso de estos últimos.</p>
3.3 Programación - Repetición Simple	Los objetivos de esta práctica son comenzar a identificar patrones y cómo los mismos pueden simplificar y mejorar comunicacionalmente el código haciendo un uso correcto de la estructura de control Repetición simple. Se espera poder abstraer problemas de situaciones cotidianas pudiendo generar modelos que se adapten a ser representados con el lenguaje QDraw. Se comienza a trabajar con ejercicios de contexto, que buscan mostrar diferencias y similitudes entre distintos lenguajes

	<p>(entendidos como diferentes conjuntos de instrucciones) que se pueden adaptar mejor a ciertos problemas. Se hace énfasis en el trabajo con las mismas estructuras de control en estos nuevos lenguajes de contexto presentados.</p> <p>Las actividades van creciendo en la complejidad de las estructuras de repetición presentadas, y en los patrones de uso de combinaciones de estructuras de repetición y llamado a procedimientos para división en subtareas.</p> <p>Para cada uno de los ejercicios se comienza a pedir una resolución que contemple el uso de las buenas prácticas de programación dadas (sintaxis estricta, documentación, indentación de código, etc.).</p>
3.4 Programación - Alternativa condicional	<p>Los objetivos de esta práctica incluyen los de la 3.3 adicionando la limitación que se presenta cuando hay información o datos que pueden variar o están ligados a una condición sobre el estado específico del contexto de ejecución. Se espera además relacionar los conceptos vistos en lógica proposicional, específicamente los referidos a valor de verdad y conectores lógicos para generación de condiciones complejas. Se hace énfasis en la importancia de los mismos en los razonamientos para la resolución de problemas de este tipo.</p>
3.5 Programación - Ejercicios integradores para el parcial	<p>Esta práctica integral tiene como objetivo repasar los conceptos vistos de la unidad de programación que serán evaluados, aprender a leer y comprender los enunciados del parcial detenidamente, identificar en cada ejercicio el autómata, contexto e instrucciones, organizar el tiempo dedicado a cada ejercicio en el parcial y detectar los temas o conceptos de mayor dificultad.</p>

Modalidad de evaluación:

Se establece como requisitos necesarios contar con una asistencia no inferior al 75% de las clases y acreditar conocimientos adquiridos de ambos bloques de la materia. La acreditación se obtiene al cumplir alguna de las siguientes condiciones:

- Aprobar cada una de las instancias parciales de evaluación en primera o segunda fecha (recuperatorio) con un mínimo de 6 y un promedio mínimo de 7 (equivalentes al 70 % y 80 % del puntaje máximo respectivamente). En este caso se considera la materia como promocionada.
- Aprobar cada una de las instancias parciales de evaluación en primera o segunda fecha (recuperatorio) y aprobar un examen integrador al finalizar la materia (o en las instancias de pendiente de aprobación establecidas por calendario académico), en todos los casos con un mínimo de 4 (equivalente al 60 % del puntaje máximo).

Notas:

- En el caso de cursada promocionada, la nota final de la cursada se calcula como el promedio de las notas de todas las evaluaciones parciales, redondeada a un dígito. Para el redondeo se tendrá en cuenta una evaluación conceptual en cuanto a trabajo en clase (participación, etc.) y compromiso con las tareas asignadas a resolver fuera del horario de clase (guías de TP, etc.).
- En el caso de estudiantes que rinden examen integrador, la nota final de la cursada se calcula como un promedio ponderado: un 60% lo aporta el promedio de las notas de todas las evaluaciones parciales y un 40 % la nota del examen integrador.

Para aprobar la asignatura se tendrá en cuenta la resolución 201/18 del Consejo Superior sobre el Régimen de estudios de la UNQ y modificatorias.

Modalidad de evaluación libre:

En la modalidad de libre se evaluarán todos los contenidos y lecturas establecidas en el presente programa. Cada estudiante aprueba con la obtención de un mínimo de 4 (cuatro) puntos en el examen. Para obtener un mínimo de 4 (cuatro) puntos se debe responder correctamente al menos un 60% de los contenidos de cada uno de los bloques (Lógica y Programación). La evaluación incluye una primera instancia en forma escrita, problemas y/o ejercicios (teóricos y prácticos) y luego se completará el examen de forma oral. Además se requiere la entrega de un trabajo práctico obligatorio, acordado previamente con la coordinación de la materia a presentar el mismo día en que se rinde el examen.

Aclaración: Se requiere que las personas inscriptas se comuniquen con la coordinación de esta asignatura vía correo electrónico con al menos 5 días hábiles de antelación al inicio de la semana de mesas de exámenes, ya que la instancia de evaluación libre puede incluir realización de TPs que requieren varios días.

Cronograma tentativo

Semana	Unidades		Actividad				
	Clase 1	Clase 2	Teórico	Práctico*			Evaluación
				1	2	3	
1	1 - Presentación materia	2 - Introducción Lógica	X	X			
2	2 - Conectivas y Tablas	2 - Lógica Proposicional	X	X		X	
3	2 - Lógica Proposicional	2 - Lógica Predicados	X	X		X	
4	2 - Lógica Predicados	2 - Lógica Predicados	X	X		X	
5	Repaso - Parcial	1er Parcial - Lógica				X	X
6	3 - Autómatas	3 - Lenguajes - Qdraw	X	X			
7	3 - División de tareas	3 - División de tareas	X	X		X	
8	Repaso - Parcial	Recuperatorio 1er Parcial - Lógica				X	X
9	Semana Buffer - Compensación Feriados					X	
10	3 - Repetición simple	3 - Alternativa condicional	X	X			
11	3 - Alternativa condicional	3 - Ejercicios integradores	X	X			
12	Repaso - Parcial	2do Parcial - Programación				X	X
13	3 - Clase consulta repaso	3 - Clase consulta repaso		X		X	
14	3 - Clase consulta repaso	3 - Clase consulta repaso		X		X	
15	Repaso - Parcial	Recuperatorio 2do parcial - Programación				X	X
16	Repaso - Integrador	Integrador				X	X

17	Reunión Balance	Límite Cierre de Actas				X	
18	Semana Buffer - Compensación Feriados					X	
*Práctico: 1. Trabajos Prácticos 2. Laboratorios 3. Otros: Actividades semanales obligatorias, consultas y seguimiento de foros.							