

## Taller de Programación Planificación Ciclo Lectivo 2022

Datos administrativos de la asignatura			
Departamento:	Sistemas	Carrera	Ingeniería en Sistemas de Información
Asignatura:	Taller de Programación		
Nivel de la carrera	3er año	Duración	Cuatrimestral
Bloque curricular:	Programación		
Carga horaria presencial semanal:	<b>4</b>	Carga Horaria total:	64
Carga horaria no presencial semanal (si correspondiese)		% horas no presenciales (si correspondiese)	
Profesor/es Titular/Asociado/Adjunto:	Ing. Matías Cassani	Dedicación:	1 DE
Auxiliar/es de 1º/JTP:	Ing. Carolina Poszkus	Dedicación:	1 DSE

### Presentación, Fundamentación

Ante un mundo en constante innovación y descubrimiento, es el desarrollo de software el que nos permite gestionar la tecnología del hardware. Estos avances ofrecen un gran abanico de posibilidades, en donde podemos desempeñarnos como profesionales de la información.

Los algoritmos y la computación tienen una importancia preponderante dentro de la especialidad y por ende en los diseños curriculares de los procesos de formación de cuadros profesionales en esta especialidad.

La programación y los algoritmos tienen un rol multidimensional pues los estudiantes:

- Desarrollan programas durante el diseño de software.
- Ejercitan y modifican programas durante los trabajos de laboratorio.
- Leen programas en el material de estudio de los cursos, textos y publicaciones.

Conforme al perfil de ISI, el área Programación tiene por objetivos: “Formar en las metodologías, técnicas y lenguajes de programación, como herramientas básicas para el desarrollo de software y el estudio de disciplinas que permitan crear nuevas tecnologías”.

Esta disciplina se ha ido fortaleciendo, debiéndose abordar más sólidamente los fundamentos matemáticos y computacionales, mayor capacidad de abstracción y vinculación con el diseño, nuevos paradigmas y patrones, nuevos y diferentes lenguajes y su vinculación con las plataformas asociadas, nuevos entornos de desarrollos y nuevas herramientas (incluso automáticas) para construir y probar programas, así como para realizar reingeniería de código existente.

En consecuencia, el diseño curricular propone una asignatura que aborda Algoritmos, Programas y Estructuras de Datos, Procedimientos, Complejidad, Abstracción de datos con un lenguaje elemental.

Lograr aprendizajes significativos con el desarrollo de estos contenidos, requiere un nivel de alfabetización informática básica asociada. Y si bien la mayor parte de los estudiantes están familiarizados con las múltiples aplicaciones y beneficios que brindan las computadoras, pocos son los que han abordado la complejidad de cómo ordenar que una computadora realice una tarea y que cumpla con lo esperado en un lenguaje de programación específico.

Se necesitan conocimientos básicos de software y hardware, en algunos casos de carácter informativo para el contexto de los aprendizajes y en otros como destrezas o capacidades de razonamiento y abstracción necesarias para el diseño de algoritmos y la construcción de programas.

En el contexto mencionado anteriormente es importante que los alumnos al momento de cursar esta electiva tengan los conocimientos previos respecto a conceptos, metodologías y herramientas básicas necesarias en la resolución de algoritmos, como paso inicial en el desarrollo de software. Con el dictado de esta electiva se busca fundamentalmente que los alumnos puedan conocer y aplicar los conocimientos adquiridos en algoritmos y estructuras de datos sobre un lenguaje de programación específico y que tiene una gran porción de mercado.

- **Relación de la asignatura con el perfil de egreso**

La asignatura contribuye en la interpretación y resolución de problemas vinculados a implementación de tecnologías de procesamiento de la información en lo que respecta a los lenguajes de programación.

- **Relación de la asignatura con los alcances del título.**

Con el aprendizaje el alumno obtendrá los conocimientos para poder identificar y discernir entre diferentes herramientas y técnicas de programación de acuerdo a necesidades de implementación propuestos.

**Relación de la asignatura con las competencias de egreso de la carrera**

Detallar, en la tabla siguiente, la relación de la asignatura con las competencias de egreso específicas, genéricas tecnológicas, sociales, políticas y actitudinales de la carrera. Indicar a cuáles competencias de egreso tributa (aportes reales y significativos de la asignatura) y en qué nivel (0=no tributa, 1=bajo, 2=medio, 3=alto). Agregar un comentario general de justificación. (Este detalle se integrará en una matriz de tributación de la carrera, dictada en la Facultad Regional, en la cual se explicita el desarrollo de las competencias específicas y genéricas de la carrera y el nivel en que tributa cada asignatura).

Competencias específicas de la carrera (CE)	Competencias genéricas tecnológicas (CT)	Competencias genéricas sociales, políticas y actitudinales (CS)
CE1: Especificar, proyectar y desarrollar software (Alta)	CT1: Contribuir a la generación de desarrollos tecnológicos y/o innovaciones tecnológicas (Media)	CS1: Desempeñarse de manera efectiva en equipos de trabajo (Media )

**Propósito**

Introducir y capacitar al alumno en la construcción y programación de sistemas de software, de manera práctica basándose en la teoría que ofrecen materias troncales de los niveles anteriores. Que realice prácticas en herramientas de programación permitiéndole aprender a construir programas en forma manual o con uso de asistentes, en un entorno visual orientado a objetos. De esta manera el alumno puede iniciarse en el mundo de la programación, obteniendo los conocimientos necesarios para lograr la construcción de un proyecto completo usando la programación orientada a objetos.

**Objetivos establecidos en el Diseño Curricular**

- Conocer los lenguajes de programación existentes en el mercado.
- Aprender y adquirir experiencia sobre los principales componentes del proceso de programación.

- Analizar problemas sencillos resolubles con computadora, poniendo énfasis en la modelización y abstracción.
- Iniciarse en el uso de un lenguaje de programación como medio para expresar un algoritmo de manera que la computadora pueda ejecutarlo.
- Obtener una preparación básica para los siguientes cursos del Área Programación de la currícula de ISI.

#### **Resultados de aprendizaje**

Describir y explicar los Resultados de aprendizaje a promover en el desarrollo de la asignatura. Argumentar su cantidad, sus componentes y la manera en que cada resultado de aprendizaje contribuye al desarrollo de las competencias que aborda la asignatura:

- RA 1 Reconocer las sentencias necesarias para implementar soluciones algorítmicas en JAVA.
- RA 2 Emplear los conceptos del paradigma orientados a objetos para diseñar e implementar programas mediante el lenguaje Java.
- RA 3 Adquirir las habilidades para desarrollar sistemas de escritorio desarrollado en un entorno JAVA.

#### **Asignaturas correlativas previas**

Para cursar debe tener cursada:

- Algoritmos y Estructura de Datos

Para rendir debe tener aprobada:

- Algoritmos y Estructura de Datos

#### **Asignaturas correlativas posteriores**

Indicar las asignaturas correlativas posteriores:

- Programación Avanzada

#### **Programa analítico, Unidades temáticas**

##### **Unidad nro. 1: Introducción al lenguaje de programación JAVA (Cap. 1 y 2)**

- ¿Qué es JAVA?
- Historia de JAVA
- ¿Cómo funciona JAVA?
- Herramientas de Desarrollo
- JAVA SE

- Instalación del JDK
- Instalación de un IDE
- Desarrollo de la primera aplicación en JAVA

**Unidad nro. 2: Fundamentos de JAVA (Cap. 3)**

- Tipos de Datos
- Declaración de una Clase
- Declaración de un Método
- Métodos Set y Get
- Constructores
- Variables de Instancia
- Encapsulamiento
- Operadores Lógicos
- Operadores Aritméticos
- Promoción y conversión de argumentos

**Unidad nro. 3: Estructuras de Control (Cap. 4 y 5)**

- Estructuras de control
- if else
- else if
- switch
- while
- do while
- for
- Instrucción break y continue

**Unidad nro. 4: Programación Orientada a Objetos (Cap. 9 y 10)**

- Herencia
- Subclases y Superclases
- La Clase Object
- Modificadores de Acceso
- public, private, protected, default
- Polimorfismo
- Clases y Métodos Abstractos
- Clases y Métodos final

- Creación y Uso de Interfaces

**Unidad nro. 5: Arreglos (Cap. 7)**

- Declaración y creación de arreglos
- Ciclo for mejorado
- Paso de arreglos a métodos
- Arreglos multidimensionales
- Listas de argumentos de longitud variable

**Unidad nro. 6: Interfaz Gráfica de Usuario (Cap. 11)**

- Introducción al desarrollo de GUI
- JOptionPane
- Componentes Swing
- JFrame
- JLabel
- JTextField
- JPasswordField
- JButton
- JCheckBox
- JRadioButton
- JComboBox
- JList
- JDataGrid
- Manejo de Eventos en GUI
- Cómo funciona el manejo de Eventos
- Clases Anidadas
- Creación de Interfaz de Usuario mediante un asistente.

**Unidad nro. 7: Colecciones en Java (Cap. 19)**

- Clase Arrays
- Interfaz Collection y la clase Collections
- Listas
- ArrayList e Iterator
- LinkedList
- Conjuntos

- TreeSet
- HashSet
- Mapas
- Operaciones con Colecciones

**Unidad nro. 8: Manejo de Excepciones (Cap. 13)**

- Generalidades del Manejo de Excepciones
- Jerarquía de Excepciones
- Bloque try, catch y finally
- Declarar nuevos tipos de Excepciones

**Unidad nro. 9: Acceso a Base de Datos con JDBC (Cap. 25)**

- Conectarse a una Base de Datos relacional
- Manipulación de Bases de Datos
- SELECT
- INSERT
- UPDATE
- DELETE
- Sentencias Preparadas
- Procesamiento de Transacciones

**Unidad nro. 10: Generación de Reportes**

- Herramientas de edición de Reportes
- IReport
- Diseño de un Reporte
- Inserción de un Reporte en una aplicación

**Metodología de enseñanza**

**Clases Teóricas:** se lleva a cabo la presentación y el desarrollo de los temas teóricos.

**Actividades prácticas en aula:** destinadas al análisis y resolución de escenarios propuestos e implementados.

**Actividades prácticas en laboratorio:** que comprenden la resolución de problemas de ingeniería y formación experimental, y en las que se abordan e integran los ejes temáticos.

**Consultas:** Se trabaja con consultas en diferentes momentos, antes de la clase, en las actividades prácticas, clases previas a los parciales y/o exámenes.

Durante el desarrollo de los Trabajos Prácticos se contará con la presencia permanente del docente la cátedra para el apoyo a los alumnos y de los becarios que el responsable del Laboratorio designe para asegurar el funcionamiento del equipamiento.

#### **Recomendaciones para el estudio**

Lectura del material bibliográfico con anticipación.

Utilización de trabajos prácticos semanales como instancias de práctica y aprendizaje.

#### **Metodología de evaluación**

##### **EVALUACIÓN**

Durante el cursado de la asignatura se implementará la evaluación formativa, llevando el registro y control de los trabajos prácticos realizados por los alumnos durante las clases prácticas. Esto permite conocer de alguna manera, el proceso de aprendizaje por parte del alumno y su participación en la asignatura.

##### **Escala de Notas (Evaluaciones Parciales y Final)**

Nota	Porcentaje	
1	0-10%	<b>Insuficiente</b>
2	11-20%	
3	21-30%	
4	31-40%	
5	41-59%	
6	60-69%	<b>Aprobado</b>
7	70-79%	<b>Bueno</b>
8	80-89%	<b>Muy Bueno</b>
9	90-95%	<b>Distinto</b>
10	96-100%	<b>Sobresaliente</b>

**Para APROBAR LA CURSADA el alumno deberá:** tener el 75% de asistencia y aprobar todos los prácticos realizados además deberá aprobar los 3 exámenes parciales que se contemplan en la presente planificación. Se contempla además la existencia de un parcial recuperatorio que integrará todos los contenidos evaluados en los parciales anteriores.

**Para la PROMOCION DIRECTA el alumno deberá:** tener el 75% de asistencia y aprobar todos los prácticos realizados además deberá aprobar los 2 exámenes parciales que se contemplan en la presente planificación con nota mayor o igual a 8 ( $\geq 8$ ) en los 3 parciales. Se contempla además la existencia de un parcial recuperatorio que integrará todos los contenidos evaluados en los parciales anteriores. Para poder obtener la promoción directa el alumno que deba asistir a la instancia de recuperación deberá obtener una nota ( $\geq 8$ ) mayor o igual a 8.

### **EXAMEN FINAL**

Constará de una evaluación teórica práctica ante un tribunal que incluye los contenidos dictados por el docente durante el ciclo lectivo. Aquí se evaluarán los conceptos estudiados durante el cursado de la materia.

### **Cronograma de clases/trabajos prácticos/exámenes (tentativo)**

<b>Clase Fecha</b>	<b>Unidad</b>	<b>Temas</b>
21/03	1	Presentación de la materia. Diagnóstico e introducción a JAVA. Instalación del entorno de programación.
28/03	2	Fundamentos de JAVA. Tipos de datos. Declaración de una clase y métodos. Constructores. Variables de Instancia. Operadores. Promoción y Conversión.
04/04	3	Estructuras de Control. (if, switch, while, do while, for, break, continue)
11/04	4	Programación Orientada a Objetos. Herencia. Subclases y Superclases. La clase Object.
18/04	4	EXAMEN PARCIAL Nro. 1 Programación Orientada a Objetos. Modificadores de Acceso. Polimorfismo. Clases y Métodos Abstractos. Clases y Métodos final. Creación de Interfaces
25/04	5	Arreglos. Declaración y creación. Ciclo for mejorado. Paso de arreglos a métodos. Arreglos multidimensionales. Listas de argumentos variable.
02/05	6	Interfaz Gráfica de Usuario. Introducción a GUI. JOptionPane. Componentes Swing
09/05	6	Interfaz Gráfica de Usuario. Manejo de Eventos en GUI.
16/05	7	Manejo de Colecciones. Clase Array. Listar. Conjuntos. Mapas. Operaciones con colecciones.
23/05	8	EXAMEN PARCIAL Nro. 2 Manejo de Excepciones. Generalidades. Jerarquías de Excepciones. Try y catch.
30/05	9	Acceso a Base de Datos. Select, Insert, Update. Sentencias Preparadas. Trabajo Practico Final.

06/07	10	Generación de Reportes. Instalación de entorno de desarrollo. Ireport. Diseño de un Reporte. Trabajo Practico Final.
13/06	10	Generación de Reportes. Inserción de un reporte en una aplicación. Trabajo Practico Final.
20/06	11	Feriado
27/06	11	Trabajo Practico Final.
04/07		EXAMEN RECUPERATORIO INTEGRADOR

### Recursos necesarios

Ámbito de realización: Aula o laboratorio de computación.

Disponibilidad de infraestructura y equipamiento. Los espacios físicos y equipamiento están disponibles para el desarrollo de las actividades prácticas. Eventualmente, los alumnos pueden realizar las actividades prácticas a través del uso de máquinas virtuales en sus propias notebooks dentro del aula sin necesidad de la utilización de laboratorios.

Actividades a desarrollar: Desarrollo de ejercicios prácticos planteados por el docente

Tiempo: Durante el dictado de las clases prácticas. (carga horaria, período que abarca)

Evaluación (de seguimiento y final): Evaluación de seguimiento por medio de entregas pautadas cuya aprobación es requerida para la regularización de la materia

### Referencias bibliográficas (citadas según Normas APA)

#### Bibliografía obligatoria:

- Como programar en JAVA 7ma Edición – Deitel & Deitel – Editorial Prentice Hall – 2008

#### Bibliografía complementaria:

- Manual de JasperSoft.

### Reuniones de asignatura y área

Reuniones periódicas de asignatura y se prevén dos reuniones de área en el año.

### Atención y orientación a las y los estudiantes

**Horario de consulta:**

Los alumnos podrán realizar las consultas sobre los contenidos de la materia ya sean estos teóricos o prácticos en el horario de clases. Además, se establece como horario de consulta los días lunes de 16 a 18 hs. en la facultad.