

MODALIDAD ACADÉMICA

Asignatura	Tecnología de Software de Base – Electiva Tercer Nivel	
Carrera	INGENIERÍA EN SISTEMAS DE INFORMACIÓN	
Ciclo Lectivo	2018	
Vigencia del programa	Desde el ciclo lectivo 2017	
Plan	2008	
Nivel	☐ 1er. Nivel ☐ 2do. Nivel ☒ 3er. Nivel ☐ 4to. Nivel ☐ 5to. Nivel	
Coordinador de la Cátedra	Ing. Valerio Frittelli	
Área de Conocimiento	☒ Programación ☐ Computación ☐ Sistemas de Información ☐ Gestión Ingenieril ☐ Modelos ☐ Complementaria	
Carga horaria semanal	8 horas cátedra.	
Anual/ cuatrimestral	Cuatrimestral	
Contenidos Mínimos, según Diseño Curricular-Ordenanza 1150 (sólo para asignaturas curriculares, no electivas)	No aplica.	
Correlativas para cursar (según Diseño Curricular-Ordenanza 1150)	Regulares	Aprobadas
	Paradigmas de Programación	Algoritmos y Estructuras de Datos
Correlativas para rendir (según Diseño Curricular-Ordenanza 1150)	Regulares	Aprobadas
	Paradigmas de Programación	Paradigmas de Programación
Objetivos generales de la Asignatura	<i>Que el estudiante:</i> <i>Comprenda, defina y aplique estructuras de datos avanzadas, junto con los algoritmos asociados a ellas, para la resolución de problemas y situaciones complejas.</i> <i>Incorpore elementos de análisis y diseño de algoritmos, para medir la eficiencia relativa de distintos algoritmos que pudieran servir para el mismo problema.</i> <i>Implemente eficientemente las soluciones planteadas, aplicando buenas prácticas y patrones de diseño, mediante una plataforma de programación de uso profesional actual, para facilitar la futura inserción del estudiante en el mundo laboral.</i>	
<u>Programa Analítico</u>		
Unidad Nro. 1: Fundamentos de Programación Orientada a Objetos en Java.		

Resultados de Aprendizaje:

- Dominar conceptos y técnicas de programación orientada a objetos en Java, así como los fundamentos del propio lenguaje, de manera de poder iniciarse en el desarrollo de aplicaciones avanzadas, en el marco de una formación profesional acorde con los requerimientos del mercado.

Contenidos:

- Clases, atributos y métodos. Atributos, modificadores de acceso, constructores, sobrecarga. Herencia. Clases abstractas. Clases de interface. Polimorfismo.
- Estructura general de un programa. Variables, operadores, tipos, conversiones de tipos. Estructura condicional. Condicional múltiple. Estructuras de iteración. Arreglos.
- Aplicaciones sobre listas implementadas en arrays. Diseño de estructuras de datos genéricas: Aplicaciones sobre listas, pilas y colas. Implementación del patrón Iterator mediante las interfaces de la API de Java.
- Control manual de homogeneidad. Generics, control de homogeneidad mediante parametrización de clases en Java.
- Gestión de paquetes de clases en Java.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [\[disponible en biblioteca central\]](#)
- 3.) Documentación oficial de la plataforma Java (Javadocs).
- 4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 6.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [\[disponible en biblioteca del Departamento de Sistemas\]](#)

Evaluación:

Evaluación de contenidos y prácticas en cuestionarios 1, 2 y 3, desafíos de programación 1 y 2, trabajo práctico integrador y parcial único.

Unidad N° 2: Excepciones. Archivos de Texto. Serialización. Expresiones Regulares.

Resultados de Aprendizaje:

- Dominar técnicas de captura, tratamiento y modelado de errores en tiempo de ejecución, para lograr control profesional de las aplicaciones y componentes que desarrolle, considerando un avance confiable en su futuro profesional.
- Mostrar dominio en técnicas de procesamiento de archivos de texto, para usarlos como fuente estándar de almacenamiento para el diseño de pruebas sobre grandes lotes de datos, de forma de asegurar un correcto proceso de prueba de sus soluciones.

Contenidos:

- Gestión de excepciones en Java. Jerarquía de clases de excepción. Excepciones chequeadas y no chequeadas. Lanzamiento y captura de excepciones. El bloque try-catch-finally, try with resources y

multicatch. Excepciones del programador (excepciones propias).

- Manejo del Sistema de Archivos sumando las clases de la API NIO.2 en el paquete java.nio.file, Interfaz Path y clases Files y FileSystem. Serialización. La interface Serializable. Clases de entrada y salida para la persistencia de objetos. Control de excepciones. Aplicaciones de la Serialización para dar persistencia a objetos completos.
- Introducción al uso de expresiones regulares para manejo de archivos de texto.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [\[disponible en biblioteca central\]](#)
- 3.) Documentación oficial de la plataforma Java (Javadocs).
- 4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [\[disponible en biblioteca del Departamento de Sistemas\]](#)

Evaluación:

Evaluación de contenidos y prácticas en cuestionarios 4 y 5, desafíos de programación desde el 2 en adelante, trabajo práctico integrador y parcial único.

Unidad N° 3: Streams y Expresiones Lambda..

Resultados de Aprendizaje:

- Mostrar capacidad básica para que aplique elementos de la API de Streams y de Expresiones Lambda para que pueda producir código conciso y significativo, en el contexto de permitir una aproximación a la programación funcional en Java.

Contenidos:

- Métodos anónimos. Interfaces Funcionales. Elementos fundamentales de la API de Streams. El package java.util.stream. Operaciones intermedias y operaciones terminales. Iteración externa vs. Iteración interna. Creación de instancias de streams. Operaciones básicas con streams.
- Expresiones Lambda en Java: funciones como entidades de primer nivel. Paso de comportamientos como parámetro. Clases de interface con métodos implementados por default. Interfaces funcionales. Alcance de expresiones Lambda.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Documentación oficial de la plataforma Java en el sitio de Oracle.

Evaluación:

Evaluación de contenidos y prácticas en cuestionario 6, desafíos de programación desde el 3 en

adelante, y en el trabajo práctico integrador.

Unidad N° 4: Análisis y Diseño de Estructuras para Búsqueda y Ordenamiento.

Resultados de Aprendizaje:

- Analizar la eficiencia de un algoritmo, ya sea en términos de tiempo de ejecución, consumo de memoria o complejidad aparente de código fuente, para poder plantear soluciones profesionales y eficientes a diversos problemas del campo de la ingeniería, teniendo en cuenta que esa capacidad de análisis les permita diferenciarse como programadores y desarrolladores, además de convertirse en referentes sólidos y confiables.
- Dominar el manejo y la aplicación de las principales estructuras de datos para búsqueda y/o ordenamiento y sus algoritmos asociados, para aplicación en diversas situaciones reales, considerando una formación sólida como futuros programadores profesionales.
- Mostrar conocimiento general de las principales clases nativas de Java para gestión de colecciones, de forma de poder comparar rendimientos entre esas estructuras nativas y las implementaciones obtenidas en el curso, en el marco de un proceso de formación que evidencie sentido crítico y capacidad de decisión.

Contenidos:

- Revisión de la Recursividad. Algoritmos recursivos y Backtracking. Revisión de la estrategia Divide y Vencerás: Merge Sort y Quick Sort.
- Revisión de elementos básicos de análisis de algoritmos y la notación Big O. Análisis de eficiencia. Comparación de rendimientos. Relaciones de Recurrencia. Teorema Maestro.
- Concepto de grupo de ordenamiento (o montículo binario o Heap) y su implementación en un arreglo. Implementación de Heaps. Operaciones de inserción y remoción. Heap Sort. Análisis de eficiencia. Aplicaciones.
- Árboles binarios y árboles binarios de búsqueda. Árboles equilibrados. Árboles AVL. Inserción y borrado en árboles AVL. Análisis de eficiencia. Revisión de procesos de inserción en árboles AVL: análisis de aspectos del código fuente.
- Búsqueda por dispersión de claves: Hashing. Diversas técnicas: direccionamiento abierto, encadenamiento separado. Funciones de dispersión.
- Treaps. Introducción de elementos de aleatoriedad en la generación de un árbol de búsqueda para simplificar el mantenimiento del equilibrio. Combinación entre árboles binarios de búsqueda y montículos binarios. Implementación de Treaps. Operaciones de inserción, equilibrado y remoción. Análisis de eficiencia.
- Listas de Salto (Skip Lists) como variante para los árboles equilibrados. Implementación de Skip Lists. Algoritmos de inserción, remoción y búsqueda. Análisis de eficiencia.
- Análisis general de paquete java.util. Clases provistas en java.util para la gestión de estructuras de datos típicas: ArrayList, LinkedList, TreeSet, HashSet, HashTable, Vector, Stack.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [\[disponible en biblioteca del Departamento de Sistemas\]](#)

- 2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [\[disponible en biblioteca central\]](#)
- 3.) Documentación oficial de la plataforma Java (Javadocs).
- 4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [\[disponible en biblioteca del Departamento de Sistemas\]](#)

Evaluación:

Evaluación de contenidos y prácticas en cuestionario 7, 8 y 9, desafíos de programación desde el 4 en adelante, en el trabajo práctico integrador y el en parcial único.

Unidad N° 5: Gestión de Bases de Datos con Java (JDBC)

Resultados de Aprendizaje:

- Dominar elementos y técnicas disponibles en el marco de la gestión de bases de datos desde aplicaciones Java, para posibilitar el diseño de aplicaciones profesionales que accedan a bases de datos, considerando un contexto profesional en el que este dominio resulta exigible.

Contenidos:

- JDBC (Java Database Connectivity): definiciones, conceptos y aplicaciones básicas. Análisis de las API de Java para JDBC. Drivers JDBC. Gestión de una conexión a una base de datos.
- Elementos de SQL. Creación y configuración de una base de datos. Sentencias SQL embebidas en una aplicación Java. Sentencias SQL precompiladas. Transacciones.
- Operaciones típicas sobre una base de datos: altas, bajas, modificaciones, consultas y sentencias SQL para su desarrollo. Análisis de patrones comunes en el contexto de JDBC.
- Introducción a JPA y aplicaciones.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Documentación oficial de la plataforma Java (Javadocs).

Evaluación:

Evaluación de contenidos y prácticas en el trabajo práctico integrador.

Unidad N° 6: Interfaces Gráficas de Usuario con JavaFX.

Resultados de Aprendizaje:

- Demostrar dominio para la aplicación de elementos básicos de creación de interfaces visuales en base al framework JavaFX, para posibilitar el diseño general de interfaces de usuario en aplicaciones profesionales, considerando un ambiente de desarrollo profesional en el que predomina el uso interfaces basadas en esta tecnología.

Contenidos:

- Estructura de una aplicación con IGU basada en JavaFX. Creación de formularios en JavaFX.
- Inclusión de objetos visuales en un formulario JavaFX. Control de eventos.
- Creación de la IGU de una aplicación con FavaFX.

- Control de estilo de visualización.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Documentación oficial de la plataforma Java (Javadocs).

Evaluación:

Evaluación de contenidos y prácticas en el trabajo práctico integrador.

Metodología de enseñanza y aprendizaje (Planificar estrategias centradas en el aprendizaje activo del estudiante)	✓ En cada clase se desarrolla un tema central, y el mismo se ejemplifica y analiza con modelos de programas presentados por los profesores. ✓ Los alumnos realizan modificaciones y variantes sobre los modelos, y realizan ejercicios nuevos en base a los temas tratados. ✓ El trabajo práctico único y las tareas semanales (cuestionarios y desafíos) integran los conocimientos de las distintas unidades, y parte del tiempo de clase se usa para analizar dudas y elementos relevantes referidos a esos trabajos. ✓ Todos los materiales del curso (materiales del estudio y evaluaciones de tipos diversos) están siempre disponibles en el aula virtual de la asignatura. El esquema de trabajo al que se orienta toda la Cátedra (desde 2015 en adelante) es el de <i>Clase Invertida</i>, con variaciones que son propias del hecho de contar con un muy elevado número de estudiantes.
Sistema de evaluación (Nombrar y describir cada una de las diferentes instancias de evaluación, pensando en la Evaluación como proceso continuo de recolección de evidencias)	✓ La materia se evalúa con un parcial único (que se puede recuperar), un trabajo práctico único, y una nota promedio entre diversas actividades prácticas (como cuestionarios y tareas especiales de programación designadas como "desafíos de programación"). ✓ El tema de cada actividad evaluable integra conocimientos de las distintas unidades del programa, y tiene complejidad y volumen de trabajo acorde a la necesidad de evaluar a los estudiantes.
Criterios de evaluación (los cuales serán tenidos en cuenta en las correcciones)	• El <i>trabajo práctico único</i> se entrega a través del aula virtual, y es revisado y calificado por los docentes de la Cátedra. • Los <i>cuestionarios y desafíos</i> son de corrección automática. • En casos de <i>parciales</i> y el <i>trabajo práctico único</i>, los alumnos deben presentar a través del aula virtual proyectos desarrollados en Java, que incluyan los archivos fuente implementados. • Cuando se trate de <i>cuestionarios y desafíos de programación</i>, los alumnos simplemente deben subir sus respuestas mediante los recursos disponibles en el aula virtual. • El enunciado del <i>trabajo práctico único</i> incluye consignas a cumplir y recomendaciones adicionales, además de criterios de evaluación tales como: compilación, completitud de consignas cubiertas, diseño de clases, lógica aplicada en la solución de cada ítem, diseño de interfaz de usuario y elementos de control de carga e integridad. Cada criterio se evalúa y aporta un peso a la calificación final. De acuerdo a la suma de esos pesos, se aplica la escala de notas que ya fue indicada en secciones anteriores de este mismo documento (y que forma parte de la consigna mostrada a los

	estudiantes). • Todo lo expresado en el ítem anterior para los trabajos prácticos, vale también para los parciales. • Esencialmente, se aplica un esquema de evaluación continua, con reglas de trabajo firmes pero claras. • En todo momento los estudiantes saben cuál será el esquema de evaluaciones, cómo serán evaluados, y cuándo les serán entregadas sus devoluciones y calificaciones. • El soporte del aula virtual única y centralizada para todos los cursos, con exactamente el mismo y único modelo de intervención para todos los cursos, facilita el proceso de avance y de control.
Regularidad: condiciones (Describir las condiciones necesarias para regularizar. Se sugiere incluir la aclaración que el estudiante en condición de regular puede rendir en el plazo de un ciclo lectivo sin control de correlativas aprobadas)	A partir de 2018 sólo se aplica la escala con aprobación desde 6(seis) (que es la misma que corresponde a la evaluación de exámenes finales). En consecuencia, para aprobar un parcial, un práctico o cualquier otra evaluación normal del ciclo de cursado, el estudiante deberá alcanzar como mínimo una nota de 6(seis). Todo alumno que obtenga al menos una calificación de 6(seis) en cada una de las tres evaluaciones generales: el parcial, el práctico integrador y el promedio de actividades prácticas de aula, obtendrá la aprobación directa: no deberá rendir examen final. Simplemente deberá inscribirse en cualquier turno en que tenga ocasión de hacerlo, y pasar el día del examen a hacer firmar la libreta con su nota final, que en todos los casos será igual al promedio simple y redondeado entre las tres notas de regularidad. Sólo se puede recuperar el parcial único, si la nota en ese parcial fuese menor que 6(seis). En ese caso, el alumno deberá hacer un parcial de recuperación. En caso de aprobar este recuperatorio, <u>la nota máxima del mismo será 6(seis)</u> (esto se hace por causa de estricta justicia, ya que la nota del recuperatorio <u>reemplazará</u> a la nota original del parcial y se procederá luego en forma normal a calcular el promedio final para aprobación directa de la materia). Pero si el alumno obtiene nota menor a 6(seis) en este recuperatorio, quedará libre. También quedará libre el alumno que no llegue al 6(seis) en el trabajo práctico único y/o en el promedio de las notas por actividades prácticas. Estas actividades no cuentan con instancia de recuperatorio: el trabajo práctico se entrega tantas veces como sea necesario hasta aprobarlo, y el promedio de las actividades prácticas abarca un conjunto suficientemente grande de calificaciones como para que cada una de ellas actúe como recuperatorio en el promedio. A partir de 2017, todo alumno puede inscribirse a rendir una asignatura dentro de un período de ciclo lectivo completo a partir del momento de la regularización, sin que afecte el cumplimiento de las correlatividades previas. A partir de vencido ese plazo de un ciclo completo, para poder inscribirse a examen deberá cumplir primero con las correlatividades previas. A los fines estadísticos, se considerará que un alumno abandonó el cursado si <i>no llegó a realizar todas las evaluaciones previstas</i>. Si hizo todas esas evaluaciones y sus calificaciones no alcanzaron a cubrir las exigencias de aprobación, el alumno será considerado libre. Escala de notas aplicable DURANTE el cursado (a partir de 2018, igual a la usada en el examen final) (*)

	<table><tr><th>NOTA</th><th>PORCENTAJE</th><th>CALIFICACIÓN</th></tr><tr><td>1</td><td></td><td>Insuficiente</td></tr><tr><td>2</td><td></td><td>Insuficiente</td></tr><tr><td>3</td><td></td><td>Insuficiente</td></tr><tr><td>4</td><td></td><td>Insuficiente</td></tr><tr><td>5</td><td></td><td>Insuficiente</td></tr><tr><td>6</td><td>60% a 68%</td><td>Aprobado</td></tr><tr><td>7</td><td>69% a 77%</td><td>Bueno</td></tr><tr><td>8</td><td>78% a 86%</td><td>Muy Bueno</td></tr><tr><td>9</td><td>87% a 95%</td><td>Distinguido</td></tr><tr><td>10</td><td>96% a 100%</td><td>Sobresaliente</td></tr></table> (*) Escala acordada en reunión de Docentes Coordinadores	NOTA	PORCENTAJE	CALIFICACIÓN	1		Insuficiente	2		Insuficiente	3		Insuficiente	4		Insuficiente	5		Insuficiente	6	60% a 68%	Aprobado	7	69% a 77%	Bueno	8	78% a 86%	Muy Bueno	9	87% a 95%	Distinguido	10	96% a 100%	Sobresaliente
NOTA	PORCENTAJE	CALIFICACIÓN																																
1		Insuficiente																																
2		Insuficiente																																
3		Insuficiente																																
4		Insuficiente																																
5		Insuficiente																																
6	60% a 68%	Aprobado																																
7	69% a 77%	Bueno																																
8	78% a 86%	Muy Bueno																																
9	87% a 95%	Distinguido																																
10	96% a 100%	Sobresaliente																																
Promoción: condiciones (Aclarar si hubiera promoción de alguna parte de la asignatura, las condiciones y si tiene duración, con el mayor detalle posible)	No aplica.																																	
Aprobación Directa: condiciones. (la calificación será la nota registrada como Nota Final en Autogestión) (Se sugiere incluir la aclaración que el estudiante, en esta condición, puede registrar su nota en examen en el plazo de un ciclo lectivo, sin control de correlativas aprobadas, y después de ello se le exigirán correlativas aprobadas)	Los alumnos que hayan obtenido la aprobación directa no deberán rendir examen final. Simplemente deberán inscribirse en cualquier turno en que tengan ocasión de hacerlo, y pasar el día del examen a hacer firmar la libreta con la nota final, que en todos los casos será igual al promedio simple y redondeado entre las tres notas de regularidad. La condición de aprobación directa no tiene vencimiento. A partir de 2017, todo alumno puede inscribirse a rendir una asignatura dentro de un período de ciclo lectivo completo a partir del momento de la regularización, sin que afecte el cumplimiento de las correlatividades previas. A partir de vencido ese plazo de un ciclo completo, para poder inscribirse a examen deberá cumplir primero con las correlatividades previas.																																	
Modalidad de examen final (Describir las características metodológicas del examen final para los distintos estados del estudiante)	Como se indicó, no hay examen final: los alumnos que regularicen tienen aprobada la materia con nota final igual al promedio entre sus notas de regularidad. Escala de Notas para Examen Final (*) <table><tr><th>NOTA</th><th>PORCENTAJE</th><th>CALIFICACIÓN</th></tr><tr><td>1</td><td></td><td>Insuficiente</td></tr><tr><td>2</td><td></td><td>Insuficiente</td></tr><tr><td>3</td><td></td><td>Insuficiente</td></tr><tr><td>4</td><td></td><td>Insuficiente</td></tr><tr><td>5</td><td></td><td>Insuficiente</td></tr><tr><td>6</td><td>60% a 68%</td><td>Aprobado</td></tr><tr><td>7</td><td>69% a 77%</td><td>Bueno</td></tr><tr><td>8</td><td>78% a 86%</td><td>Muy Bueno</td></tr><tr><td>9</td><td>87% a 95%</td><td>Distinguido</td></tr><tr><td>10</td><td>96% a 100%</td><td>Sobresaliente</td></tr></table> (*) Escala acordada en reunión de Docentes Coordinadores	NOTA	PORCENTAJE	CALIFICACIÓN	1		Insuficiente	2		Insuficiente	3		Insuficiente	4		Insuficiente	5		Insuficiente	6	60% a 68%	Aprobado	7	69% a 77%	Bueno	8	78% a 86%	Muy Bueno	9	87% a 95%	Distinguido	10	96% a 100%	Sobresaliente
NOTA	PORCENTAJE	CALIFICACIÓN																																
1		Insuficiente																																
2		Insuficiente																																
3		Insuficiente																																
4		Insuficiente																																
5		Insuficiente																																
6	60% a 68%	Aprobado																																
7	69% a 77%	Bueno																																
8	78% a 86%	Muy Bueno																																
9	87% a 95%	Distinguido																																
10	96% a 100%	Sobresaliente																																
Actividades en laboratorio	✓ Todas las clases de la materia se desarrollan en el laboratorio de																																	

	informática. ✓ Cada alumno debe instalar en su computadora personal las herramientas de software requeridas por la materia, y realizar en forma personal prácticas y pruebas sobre ellas para lograr dominio pleno de su uso. ✓ Es bienvenida la utilización por parte de los alumnos de sus propias notebooks, en el transcurso de cada clase. ✓ En varias clases se prevé la realización de trabajos y tareas pautadas para ser terminadas (en la medida de lo posible) en el transcurso de la misma clase. Esos trabajos se suben al aula virtual para su revisión por parte de los jefes de trabajos prácticos y auxiliares del curso.
Cantidad de horas prácticas totales (en el aula)	64 horas.
Cantidad de horas teóricas totales (en el aula)	64 horas.
Cantidad de horas estimadas totales de trabajo (extra áulicas).	64 horas.
Horas/año totales de la asignatura (en el aula).	128 horas.
Tipo de formación práctica (sólo si es asignatura curricular -no electiva-)	No aplica.
Cantidad de horas cátedras afectadas a la formación práctica indicada en el punto anterior (sólo si es asignatura curricular -no electiva-)	No aplica.
Descripción de los prácticos	✓ Desarrollo de programas basados en diversas estructuras de datos estudiadas y en diversas estrategias de resolución de problemas. ✓ Desarrollo de programas que apliquen técnicas de búsqueda basadas en estructuras de búsqueda rápida. ✓ Desarrollo programas que apliquen tecnologías Java para acceso a bases de datos con JPA y con interfaces visuales de usuario basadas en JavaFx.
Cronograma de actividades de la asignatura (contemplando las fechas del calendario 2019 y para cada unidad)	Ver archivo CRTSB2019.doc anexo a esta presentación.
Propuesta para la atención de consultas y mail de contacto.	Email de los docentes: Ing. Valerio Frittelli: vfrittelli@gmail.com Ing. Gustavo Federico Bett: gfbett@gmail.com Ing. Felipe Steffolani: fsteffolani@gmail.com Ing. Diego Serrano: diegojserrano@gmail.com Ing. Romina Teicher: rteicher@gmail.com Ing. Marcela Tartabini: mtartabini@gmail.com En caso de requerir horarios de consulta, serán pactados en el momento con los alumnos que lo requieran. TODO el trabajo de la cátedra está

	permanentemente disponible para los alumnos en el espacio virtual de la cátedra (http://uv.frc.utn.edu.ar) Desde esa aula virtual se canalizan todas las comunicaciones, trabajos prácticos, subida y bajada de archivos, apuntes, notas de clases, foros, planillas de calificaciones, etc.																								
Plan de integración con otras asignaturas	La asignatura se basa en dos previas: Algoritmos y Estructuras de Datos (de primer año) y Paradigmas de Programación (de segundo año). De ambas, el alumno debe traer un adecuado dominio de la programación de objetos y fundamentos de Java, además de conocimientos sólidos de las estructuras de datos tocadas en esas asignaturas. La materia es la base para el posterior cursado de Diseño de Lenguajes de Consulta (Electiva Cuarto Nivel).																								
Bibliografía Obligatoria	1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [disponible en el aula virtual de la asignatura]																								
Bibliografía Complementaria	1.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [disponible en biblioteca del Departamento de Sistemas] 2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [disponible en biblioteca central] 3.) Documentación oficial de la plataforma Java (Javadocs). 4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [disponible en biblioteca del Departamento de Sistemas] 5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [disponible en biblioteca del Departamento de Sistemas]																								
Distribución de docentes	<table><tr><th>Curso</th><th>Turno</th><th>Día y Horas</th><th>Profesor</th><th>Jefe Trab.Práct.</th><th>Ayudante</th></tr><tr><td>3K1</td><td>mañana</td><td>Mie 1-2-3-4 Jue 4-5-6-7</td><td>Frittelli</td><td></td><td>Serrano</td></tr><tr><td>3K3</td><td>tarde</td><td>Mar 3-4-5-6 Vie 3-4-5-6</td><td>Steffolani</td><td></td><td>Teicher</td></tr><tr><td>3K4</td><td>noche</td><td>Mar 3-4-5-6 Mie 3-4-5-6</td><td>Bett</td><td></td><td>Tartabini</td></tr></table>	Curso	Turno	Día y Horas	Profesor	Jefe Trab.Práct.	Ayudante	3K1	mañana	Mie 1-2-3-4 Jue 4-5-6-7	Frittelli		Serrano	3K3	tarde	Mar 3-4-5-6 Vie 3-4-5-6	Steffolani		Teicher	3K4	noche	Mar 3-4-5-6 Mie 3-4-5-6	Bett		Tartabini
Curso	Turno	Día y Horas	Profesor	Jefe Trab.Práct.	Ayudante																				
3K1	mañana	Mie 1-2-3-4 Jue 4-5-6-7	Frittelli		Serrano																				
3K3	tarde	Mar 3-4-5-6 Vie 3-4-5-6	Steffolani		Teicher																				
3K4	noche	Mar 3-4-5-6 Mie 3-4-5-6	Bett		Tartabini																				

Firma:

Aclaración: