

**UNIVERSIDAD NACIONAL
DE
SANTIAGO DEL ESTERO**

**FACULTAD DE CIENCIAS EXACTAS Y
TECNOLOGÍAS**

PLANIFICACIÓN ANUAL 2023

ASIGNATURA: PROGRAMACIÓN II

**PROGRAMADOR UNIVERSITARIO EN
INFORMÁTICA
Plan de Estudio: 2017**

Equipo cátedra:

Profesor Adjunto: Méndez Analía

Jefe de Trabajos Prácticos: Zarco Raquel



PLANIFICACIÓN DE LA ASIGNATURA

1- IDENTIFICACIÓN:

- 1.1. **Nombre de Asignatura:** Programación II.
- 1.2. **Carrera:** Programador Universitario en Informática.
- 1.3. **Plan de Estudios:** 2017.
- 1.4. **Año académico:** 2023.
- 1.5. **Carácter:** obligatoria.
- 1.6. **Ubicación de la Asignatura en el Plan de Estudios**
 - 1.6.1. **Módulo – Año:** Tercer módulo – Segundo año.
 - 1.6.2. **Área/Bloque/Tramo al que pertenece la Asignatura/Obligación Curricular, según la organización del Plan de Estudios:**

ÁREAS/BLOQUE/TRAMO	CARGA HORARIA PRESENCIAL
Estructuras de Datos	25
Algoritmia básica	25
Paradigmas de Programación	25
CARGA HORARIA TOTAL DE LA ACTIVIDAD CURRICULAR	75

Tabla 1: Carga horaria por área/bloque/tramo

1.6.3. Correlativas

1.6.3.1 Anteriores:

Débiles: Elementos de Álgebra, Programación I.
Fuerte: Fundamentos de la Programación.

1.6.3.2. Posteriores:

Débiles: Programación III, Sistemas Operativos.
Fuertes: Seminario de Ética y Deontología. Configuración y Uso de Software.



1.7. Carga horaria:

1.7.1. Carga horaria semanal total

1.7.1.1. **Presencial:** 5.

1.7.1.2. **No Presencial:** no indicado.

1.7.2. Carga horaria semanal destinada a la formación práctica

1.7.1.3. **Presencial:** 3.

1.7.1.4. **No Presencial:** no indicado.

1.7.3. Carga horaria total dedicada a la formación práctica: 45.

1.8. Ámbitos donde se desarrollan las actividades de formación práctica a las que se hace referencia en el punto anterior.

Las actividades de formación práctica se desarrollan en laboratorios de computadoras pertenecientes al Departamento de Informática.

1.9. Indique si la asignatura se dicta en más de una comisión: una comisión.

2- PRESENTACIÓN

2.1. Ubicación de la Asignatura como tramo de conocimiento de una disciplina.

Esta asignatura está dedicada al estudio, análisis y manejo de estructuras de datos encadenadas y a la aplicación de algoritmos de recorrido, búsqueda y actualización sobre dichas estructuras. En ella se abarcan:

- Estructuras de datos encadenadas, como listas, pilas y colas.
- Representación de datos en memoria y estrategias de representación.
- Algoritmos de recorrido, búsqueda, ordenamiento, y actualización.
- Resolución de problemas y desarrollo de algoritmos mediante la aplicación del Paradigma de Programación Orientado a Objetos.

2.2. Conocimientos y habilidades previas que permiten encarar el aprendizaje de la Asignatura.

- Conocimientos matemáticos adquiridos en Elementos de Álgebra.
- Conocimientos básicos sobre Informática, lenguajes de programación, paradigmas de programación, tipos de datos, y resolución de problemas en computadora, que fueron abordados durante el cursado de la asignatura Fundamentos de la Programación.
- Conocimientos y destrezas adquiridos durante el cursado de Programación I relativos a conceptos tales como modularidad, procedimientos y funciones, tipos abstractos de datos, así como la aplicación práctica del Paradigma de Programación Orientado a Objetos y el modelado de clases en la resolución de ejercicios de computadora.

2.3. Aspectos del Perfil Profesional del Egresado a los que contribuye la asignatura.

Esta asignatura contribuye a:

- La formación técnica del alumno, abarcando los conocimientos destinados al aprendizaje y al uso de lenguajes de programación y herramientas informáticas en entornos específicos para la resolución de los problemas concretos.



- Preparar recursos humanos calificados y competentes, de nivel universitario, en el ámbito técnico de la Informática.

En esta asignatura se proporciona al alumno la capacitación básica necesaria para:

- Implementar, probar, documentar y mantener aplicaciones software.
- Participar en las soluciones de programación en el sector de SSI.

2.4. Integración horizontal y vertical con otras asignaturas.

- **Integración vertical.**

La integración vertical de esta asignatura se establece, en principio, con la asignatura Programación I mediante la incorporación de un práctico introductorio que refuerza lo aprendido en Programación I y en el que se efectúa un repaso de la aplicación práctica del Paradigma de POO y el modelado de clases en la resolución de ejercicios de computadora.

En Fundamentos de la Programación, el alumno adquiere los conocimientos básicos de la programación, resuelve problemas y usa un lenguaje de programación como herramienta, aplicando el Paradigma de Programación Imperativo. Con esta asignatura se establece integración horizontal.

Con la asignatura Programación III, que se dicta durante el segundo cuatrimestre del segundo año de la carrera, también se organiza una integración vertical. El estudio de estructuras de datos encadenadas y la aplicación de algoritmos de recorrido, búsqueda, ordenamiento y actualización, así como el desarrollo de destrezas para la creación de aplicaciones de software aplicando el paradigma de POO, preparan al alumno para el cursado de Programación III.

- **Integración horizontal**

A través del aprendizaje de la estructura de datos pilas se brindan los conocimientos que permiten a los alumnos entender la organización interna de una parte de la memoria en la asignatura Organización del Computador.

Así mismo, los temas que se dictan en simultáneo en Organización del Computador le sirven al alumno para poder interpretar como se ejecutan las instrucciones de los programas que se desarrollan en Programación II utilizando el lenguaje de programación Java.

Laboratorio III aborda la programación avanzada en lenguaje Java. La coordinación e integración horizontal con esta asignatura se realiza mediante la preparación de trabajos prácticos en los cuales, además de utilizar lo aprendido sobre las estructuras de datos encadenadas, se aplican las estrategias de programación y los paquetes de clases estudiados en Laboratorio III.

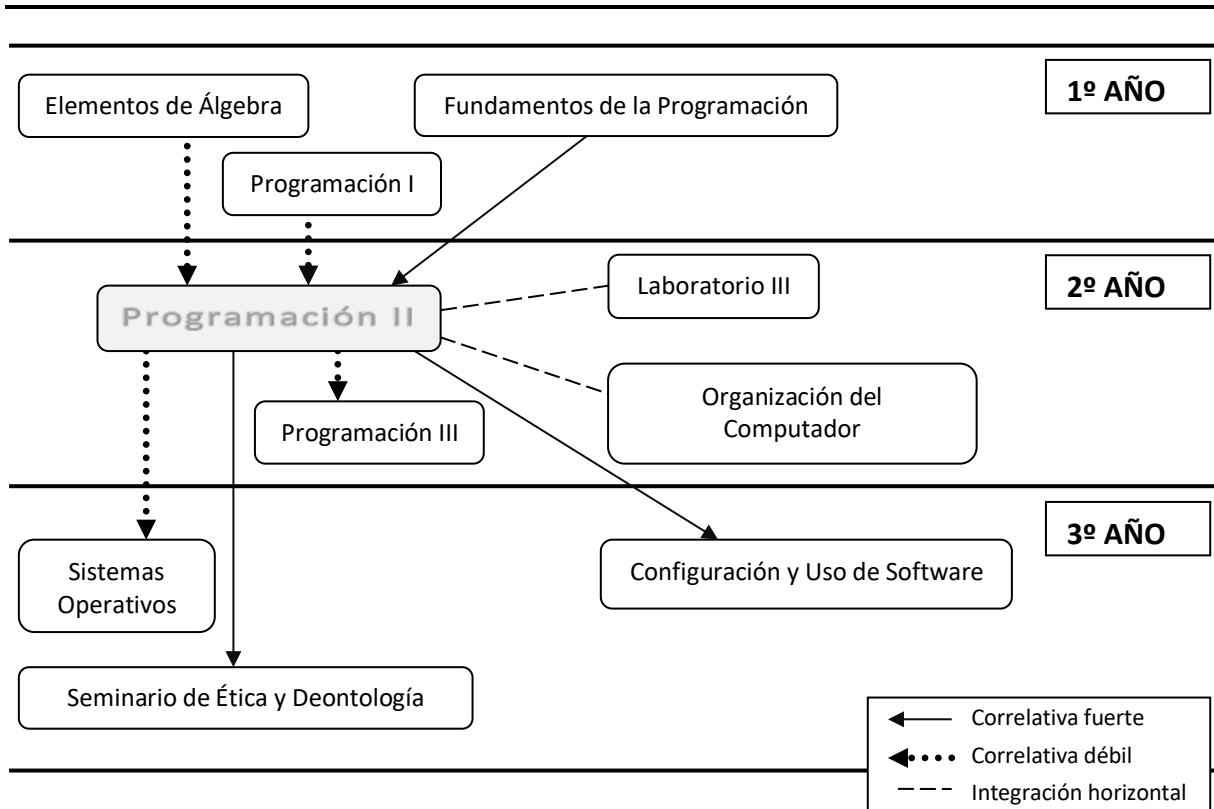


Figura 1. Integración vertical y horizontal con otras asignaturas.

3- OBJETIVOS: Que el alumno desarrolle las siguientes competencias...

Competencias básicas

- Capacidad para buscar, seleccionar y procesar la información necesaria para la resolución de problemas.
- Capacidad para verificar las soluciones encontradas.

Competencias básicas

- Capacidad de investigación.
- Capacidad para actuar en nuevas situaciones.
- Capacidad para buscar, seleccionar y procesar la información necesaria para la resolución de problemas.
- Capacidad para verificar las soluciones encontradas.

Competencias transversales

- Capacidad para relacionar la teoría a la práctica.



- Capacidad para apropiarse de nuevos métodos, paradigmas y tecnologías y adaptarse a nuevas situaciones.
- Capacidad para identificar, plantear y resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
- Capacidad para la comunicación oral y escrita enfocada en el intercambio y transmisión de los conocimientos.

Competencias Específicas: Que el alumno sea capaz de...

- Comparar las características de las estructuras de datos encadenadas implementadas dinámicamente considerando la eficiencia como parámetro para analizar sus semejanzas y diferencias.
- Desarrollar aplicaciones de software en Java usando estructuras de datos encadenadas aplicando el Paradigma de Orientación a Objetos en la resolución de problemas tipo.

4- SELECCIÓN Y ORGANIZACIÓN DE CONTENIDOS

4.1. Contenidos mínimos establecidos en el Plan de Estudios para la Asignatura:

Estructuras de datos encadenadas: listas. Operaciones e implementación de listas. Representación de datos en memoria y estrategias de implementación. Pilas y colas implementadas dinámicamente. Algoritmos de recorrido, búsqueda y actualización aplicados a listas, pilas y colas.

4.2. Programa Sintético sobre la base de los contenidos mínimos

Unidad 1: Estructura de datos Lista.

Estructuras de datos encadenadas: listas. Operaciones e implementación de listas. Representación de datos en memoria y estrategias de implementación. Algoritmos de recorrido, búsqueda y actualización aplicados a listas.

Unidad 2: Pilas dinámicas.

Pilas implementadas dinámicamente. Algoritmos de recorrido, búsqueda y actualización aplicados a pilas.

Unidad 3: Colas dinámicas.

Colas implementadas dinámicamente. Algoritmos de recorrido, búsqueda y actualización aplicados a colas.

4.3. Articulación Temática de la Asignatura /Obligación Curricular

En la figura 2 se presentan los principales conceptos a tratar en la asignatura.

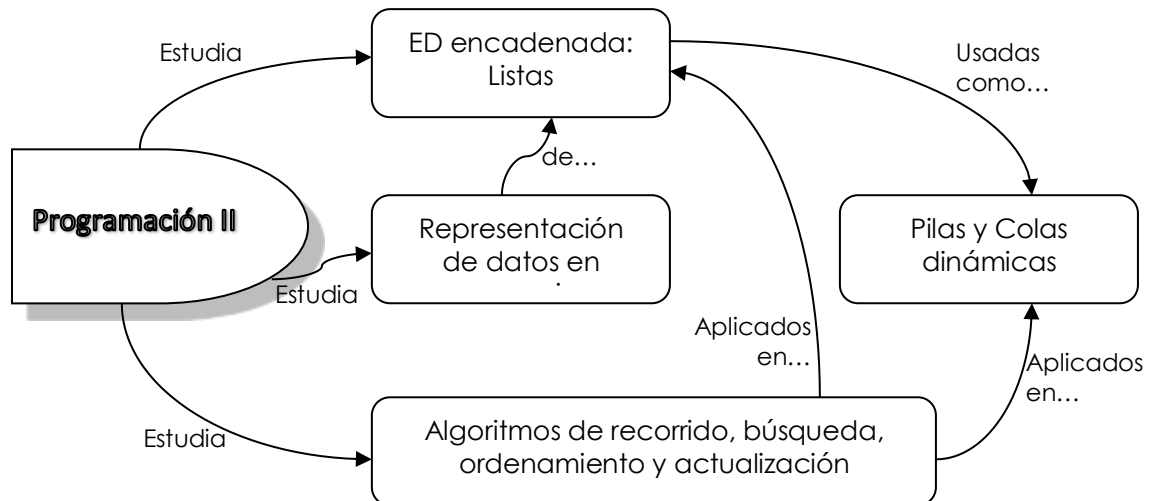


Figura 2. Articulación temática de la asignatura Programación II.

4.4. Programa Analítico

Unidad 1: Estructura de datos Lista.

Estructuras de datos encadenadas: listas. Concepto. Clasificación. TAD Lista. Operaciones e implementación de listas. Representación de datos en memoria y estrategias de implementación. Algoritmos de recorrido, búsqueda y actualización aplicados a listas. Lista ordenada. Lista doble encadenada.

Unidad 2: Pilas dinámicas.

TAD pila. Pilas implementadas dinámicamente. Operaciones básicas. Algoritmos de recorrido, búsqueda y actualización aplicados a pilas. Evaluación de expresiones aritméticas con pilas. Aplicaciones del TAD pila y resolución de problemas.

Unidad 3: Colas dinámicas.

TAD cola. Colas implementadas dinámicamente. Operaciones básicas. Algoritmos de recorrido, búsqueda y actualización aplicados a colas. Colas circulares. Bicolos o colas de doble entrada. Aplicaciones del TAD cola y resolución de problemas.



4.5. Cronograma para el desarrollo de las Unidades Temáticas

UNIDAD	CARGA HORARIA	CRONOGRAMA DE DICTADO
Unidad 1: Estructura de datos Lista	16	Semanas 1 a 8
Unidad 2: Pilas dinámicas	6	Semanas 9 a 10 y 12 a 13
Unidad 3: Colas dinámicas	8	Semanas 11 a 15
TOTAL	30	

Tabla 2: Cronograma para el desarrollo de las unidades temáticas

4.6. Programa y cronograma de formación práctica

ACTIVIDAD	CARGA HORARIA	CRONOGRAMA DE DESARROLLO
TP Listas simples (LSE)	12	Semanas 1 a 3 y 7 a 8
TP Listas simples ordenadas (LSEO)	12	Semanas 4 a 8
TP Pilas implementadas dinámicamente	9	Semanas 9 a 10, 13 y 15
TP Colas implementadas dinámicamente	12	Semanas 11 a 15
TOTAL	45	

Tabla 3: Cronograma para el desarrollo de los trabajos prácticos

5- BIBLIOGRAFÍA.

TÍTULO	AUTOR/ES	EDITORIAL	EJEMPLARES DISPONIBLES	AÑO DE EDICIÓN
Programación orientada a objetos en Java	Blasco F.	Ediciones de la U	Biblioteca Digital eLibro.net	2019
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/222723/		
Estructuras de datos. 3º Edición	Cairo Osvaldo – Guardati Silvia	McGraw-Hill	1 en Centro de Documentación	2006

Tabla 4: Bibliografía



Tabla 4: Bibliografía (continuación)

TÍTULO	AUTOR/ES	EDITORIAL	EJEMPLARES DISPONIBLES	AÑO DE EDICIÓN
Introducción a la programación, lógica y diseño (7° ed.)	Farrel, J.	Cengage Learning	Biblioteca Digital eLibro.net	2013
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/93265/		
Estructuras de Datos y Algoritmos	Fritelli, Valerio - Guzman, A. - Tymoschuk, J.	Jorge Sarmiento Editor - Universitas	Biblioteca Digital eLibro.net	2020
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/175249/		
UML Aplicaciones en Java y C++	Jiménez de Parga C..	RA-MA Editorial	Biblioteca Digital eLibro.net	2014
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/106466/		
Estructuras de datos en Java	Joyanes Aguilar Luis, Zahonero Martinez Ignacio	McGraw-Hill	1 en Centro de Documentación	2008
Programación orientada a objetos en C++ y Java: un acercamiento interdisciplinario	López Goytia J.L.	Grupo Editorial Patria	Biblioteca Digital eLibro.net	2015
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/39461/		
Lógica de programación orientada a objetos	Oviedo Regino E. M.	Ecoe Ediciones	Biblioteca Digital eLibro.net	2015
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/70431/		
Empezar a programar usando Java (3° ed.)	Prieto Sáez, N & Casanova Faus, A.	Editorial de la Universidad Politécnica de Valencia	Biblioteca Digital eLibro.net	2016
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/57434/		
Java: curso práctico	Vegas Gertrudix, J.	RA-MA Editorial	Biblioteca Digital eLibro.net	2020
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/222715/		
Diseñar y programar, todo es empezar: una introducción a la Programación Orientada a Objetos usando UML y Java	Vélez Serrano, J.	Dykinson	Biblioteca Digital eLibro.net	2011
	ENLACE DE ACCESO	https://elibro.net/es/lc/unsebiblio/titulos/63076/		
Estructura de datos en JAVA	Weiss Mark Allen.	Pearson	2 en Centro de Documentación	2013



6- ESTRATEGIAS METODOLÓGICAS.

6.1. Aspectos pedagógicos y didácticos

La metodología de enseñanza que se aplicará en el aula será en un espacio de diálogo y construcción, en el que se trabaje interactuando permanentemente, y de esta manera tanto los alumnos como el docente se consideran fuente de información.

Las actividades de estudio tanto prácticas como de tipo teórico serán apoyadas y desarrolladas mediante la plataforma Moodle del Centro Universitario Virtual (CUV - <http://cuv.unse.edu.ar>). Mediante esta plataforma se acercarán al estudiante diferentes recursos tales como videos tutoriales y materiales de estudio creados por los docentes, se utilizarán herramientas como cuestionarios evaluativos y de diagnóstico, así como la presentación digital de las actividades y proyectos propuestos en las prácticas. Cada contenido desarrollado es mediado para su mejor comprensión y con el fin de propiciar el diálogo y la discusión.

Tanto en las clases teóricas como prácticas se aplicará la metodología de aula invertida. Los alumnos realizarán actividades de aprendizaje y de evaluación formativa previamente a los encuentros con los docentes. Durante las clases se completarán las actividades requeridas para el abordaje de cada tema y se organizarán tareas individuales y grupales colaborativas que favorezcan la apropiación del estudio, y que el alumno deberá completar y enviar mediante la plataforma del CUV, al finalizar la clase.

Cada uno de los trabajos prácticos propuesto en esta asignatura incluye el desarrollo de problemas mediante diagramación UML de clases y diagramación de flujo, así como ejercicios en los que se solicita la programación completa en lenguaje Java de la resolución del enunciado. Los conceptos teóricos y los aspectos metodológicos de la práctica requerida son abordados tanto en los encuentros teóricos como en los prácticos, a fin de fortalecer las habilidades de programación en el alumno.

Se utiliza software libre en la tarea de desarrollar habilidades prácticas en el alumno, necesarias para la resolución de problemas y el diseño de algoritmos y para prepararlos a enfrentar los distintos cambios del mundo tecnológico.

6.2. Mecanismos para la integración de docentes

Se efectuarán reuniones periódicas con la cátedra de la asignatura Laboratorio III, que se dicta en el mismo cuatrimestre y en la cual los alumnos aprenden lenguaje Java avanzado. Estas reuniones permitirán realizar un seguimiento temático acorde.

Por otra parte, con las cátedras de asignaturas como Programación I y Programación III, la coordinación se realiza mediante encuentros anuales en los que se revisan los logros alcanzados en la comprensión y aplicación del paradigma de programación estudiado y las actividades prácticas.

6.3. Recursos Didácticos

Se utilizarán diversos recursos didácticos a fin del logro de los objetivos propuestos, a saber:

- Ambiente integrado de desarrollo (IDE) para la programación en lenguaje Java.



- Para desarrollar aplicaciones en Java se necesita la herramienta de desarrollo Java Development Kit (JDK). JDK incluye Java Runtime Environment, el compilador Java y las API de Java.
- Bibliografía actualizada digital y apuntes elaborados por el equipo docente de la asignatura, disponibles en la plataforma del CUV y en el Centro de Documentación del Laboratorio de Informática.
- Se utilizará Internet, enunciados tomados de las actividades prácticas propuestas, diapositivas y videos elaborados por los docentes de la asignatura, demos, tutoriales, etc.
- Se contará con el espacio de la asignatura en la plataforma Moodle del Centro Universitario Virtual (CUV), que permitirá al alumno acceder a consultas virtuales (on-line o secuencial) a distancia, acceso a los recursos digitales, apoyo de prácticas, etc. También brindará a los alumnos un canal de comunicación permanente donde podrán acceder a información actualizada de la asignatura: fechas de evaluaciones, resultados de parciales, condición final de la cursada, etc.

7- EVALUACIÓN

7.1. Evaluación Diagnóstica

Teniendo en cuenta que la evaluación diagnóstica no sólo es una estimación, sino que tiene como propósito contribuir al aprendizaje, se llevará a cabo una única evaluación diagnóstica, al comienzo de las clases, cuya finalidad será determinar el nivel de conocimientos y habilidades previas que permitan encarar el aprendizaje de la asignatura.

Los contenidos sobre los que se evaluará serán: Estructuras estáticas: pilas y colas, conceptos básicos sobre la POO: clases, objetos, atributos, mensajes, métodos.

La evaluación diagnóstica será especialmente diseñada, individual, escrita y objetiva.

Se hará una prueba de opción múltiple para que el alumno marque la opción correcta. El nivel de calificación será cuantitativa politómica. Escala: 1 al 10. Se implementará utilizando la herramienta cuestionario de la plataforma del CUV.

7.2. Evaluación Formativa

Se implementará un proceso de evaluación formativa de carácter sistemático, cualitativo y continuo, se pretende resaltar el papel de evaluación orientadora, reguladora y motivadora.

Se busca obtener información relevante respecto al proceso de enseñanza – aprendizaje que permita comprender cómo se produce dicho proceso y tomar las decisiones pertinentes en vistas a mejorar tanto el desarrollo como los resultados del aprendizaje. Permitirá identificar la evolución en el aprendizaje de los alumnos y el grado de impacto de la propuesta educativa que lleva a cabo la cátedra.



Mediante la evaluación formativa se motivará, reforzará y proporcionará ayuda a los estudiantes, reconociendo y observando sus aprendizajes, dificultades y posibilidades.

La evaluación formativa de los contenidos teóricos se llevará a cabo mediante la aplicación de herramientas disponibles en la plataforma Moodle del CUV. En particular se utilizará la herramienta Cuestionario (CD), siguiendo la técnica de la pregunta. Regularmente, y como una actividad de motivación, al comienzo o finalización de clase teórica se implementarán dinámicas de evaluación formativa, tales como escrito de un minuto, tarjetas de aplicación, preguntas abiertas de respuesta breve, cuyos resultados serán luego discutidos en aula.

En cuanto a las competencias relacionadas a la práctica, la evaluación formativa se llevará adelante aplicando la técnica de casos y se estructurará mediante la entrega regulada de diferentes enunciados propuestos en cada trabajo práctico. El seguimiento del desempeño en la práctica por parte de cada alumno podrá ser monitorizado por el docente mediante las herramientas disponibles en la plataforma Moodle del CUV.

7.3. Evaluación Parcial

7.3.1. Programa de Evaluaciones Parciales

EVALUACIÓN	TEMA	MODALIDAD	SEM	FECHA	DEVOLUCIÓN DE RESULTADO
Parcial Práctico 1	Listas	Individual. Escrito. En computadora. Resolución de un problema propuesto mediante: • Diagramación UML de clases aplicando POO. • Diagramación de flujo. • Desarrollo de proyecto de POO con lenguaje Java.	8	11/05	16/05
Recuperatorio 1			10	23/05	01/06
Parcial Práctico 2	Pilas y colas		13	15/06	19/06
Recuperatorio 2			15	29/06	30/06

Tabla 5: Descripción y cronograma de las Evaluaciones Parciales



7.3.2. Criterios de Evaluación

ESTRUCTURA	OBJETIVO Determinar si los alumnos han adquirido las competencias para:	CRITERIOS DE EVALUACIÓN
Diagramación UML de clases aplicando Orientación a Objetos	<ul style="list-style-type: none">• Modelar y documentar la resolución de un problema mediante diagramación UML de clases, identificando las clases y las relaciones entre ellas.	<ul style="list-style-type: none">• Correcto uso de la diagramación UML, con identificación de relaciones entre clases.
Diagramación de flujo de operaciones básicas de la estructura de datos en estudio	<ul style="list-style-type: none">• Emplear las estructuras de datos estudiadas, y sus operaciones básicas, en la resolución de problemas mediante diagramación de flujo.	<ul style="list-style-type: none">• Correcta representación de las operaciones de la/s estructura/s de datos en estudio.
Diagramación de flujo de la solución de un problema planteado		<ul style="list-style-type: none">• Correcta aplicación de estructuras de diagramación.• Resolución algorítmica en tiempo y forma adecuados.
Desarrollo de proyecto de POO para la resolución de un problema planteado utilizando el lenguaje de programación Java.	<ul style="list-style-type: none">• Desarrollar una solución de programación empleando la orientación a objetos mediante la creación de un proyecto en lenguaje Java utilizando un Entorno de Desarrollo Integrado y aplicando la reutilización de código.	<ul style="list-style-type: none">• Correcto análisis y diseño de la solución al problema propuesto aplicando la POO.• Eficiente codificación en lenguaje Java del enunciado, verificación y funcionamiento de la aplicación.• Capacidad para el análisis crítico y el aporte de mejoras sobre código desarrollado.• Código organizado y claro, incluyendo comentarios, indentación y modularidad.

Tabla 6: Estructura, objetivos y criterios de las Evaluaciones Parciales

7.3.3. Escala de Valoración

La escala de valoración a emplear en los parciales y recuperatorios será cualitativa dicotómica (aprobado – desaprobado).

7.4. Evaluación Integradora

La evaluación integradora se llevará a cabo a través de un Trabajo Integrador Final en el que los alumnos deberán elaborar un resumen y exponer sobre un tema asignado referido a Colas circulares y Bicolos.



7.5. Evaluación Sumativa

7.5.1. Condiciones para lograr la promoción sin Examen Final de la Asignatura.

- Asistir como mínimo al 80% del total de sesiones.
- Aprobar los Cuestionarios Evaluativos con un puntaje ≥ 7 puntos.
- Aprobar los Parciales Prácticos.
- Aprobar el Trabajo Integrador.

7.5.2. Condiciones para lograr la Regularidad de la Asignatura.

- Aprobar los Cuestionarios Evaluativos o sus correspondientes instancias recuperatorias.
- Aprobar los dos (2) Parciales Prácticos previstos, o sus correspondientes Recuperatorios.

7.6. Examen Final.

En el examen final los alumnos serán evaluados sobre los contenidos teóricos y prácticos de la asignatura. El examen podrá ser oral o escrito, e individual.

7.7. Examen Libre.

Para el examen Libre el alumno deberá aprobar las tres (3) instancias que se detallan a continuación, siendo cada una de ellas eliminatorias:

Primera Etapa.

Presentación, prueba y defensa de un planteamiento práctico en lenguaje Java, con evaluación en computador, cuya temática y planteo deberá ser solicitado a la cátedra con al menos 7 (siete) días de anticipación a la fecha de examen. El trabajo deberá ser presentado en soporte digital e impreso, cumplimentando las pautas establecidas en el enunciado. El trabajo se presentará en la fecha del examen y deberá ser aprobado por el tribunal.

Segunda Etapa.

Aprobar una evaluación presencial consistente en:

- Práctica escrita con diagramación UML de clases y diagramación de flujo, utilizando listas, pilas y/o colas, y aplicando Orientación a Objetos.
- Práctica en computador, utilizando el lenguaje Java y Programación Orientada a Objetos sobre un problema aplicando listas, pilas y/o colas.

Tercera Etapa.

Aprobar una evaluación oral/escrita de contenidos teóricos del programa analítico.

Méndez Analía

.....
Profesor responsable de Asignatura