



República Argentina – Universidad Nacional de Moreno
“1985-2025 – 40 Aniversario del Consejo Interuniversitario Nacional”
Departamento de Ciencias Aplicadas y Tecnología

Disposición

Número: UNM-DCAYT 36/25

Ciudad de MORENO
Martes 10 de junio de 2025

Referencia: Modificación PROGRAMA ENTORNO C y JAVA (2043R)- IEL

VISTO el Expediente N° UNM-EXP: 127/2014 del Registro de la UNIVERSIDAD NACIONAL DE MORENO, y CONSIDERANDO:

Que el REGLAMENTO GENERAL ACADÉMICO, aprobado por Resolución UNM-R N° 37/10 y sus modificatorias, el que fuera ratificado por el Acta de la Sesión Ordinaria N° 01/13 del CONSEJO SUPERIOR de fecha 25 de Junio de 2013, establece el procedimiento para la aprobación de las obligaciones curriculares que integran los Planes de Estudios de las carreras que dicta esta UNIVERSIDAD NACIONAL.

Que por Disposición UNM-DCAYT N° 03/14 se aprobó el Programa de la asignatura ENTORNO C y JAVA (2043R), del ÁREA: REDES, correspondiente al CICLO SUPERIOR de la Carrera INGENIERÍA EN ELECTRÓNICA del DEPARTAMENTO DE CIENCIAS APLICADAS Y TECNOLOGÍA, de esta UNIVERSIDAD, con vigencia a partir del 2do. Cuatrimestre del Ciclo Lectivo 2014.

Que conforme lo dispuesto en el citado REGLAMENTO GENERAL, se ha elevado una nueva propuesta de Programa de la asignatura antes referida y en sustitución del vigente, aconsejando su aprobación con vigencia a partir del 1er. Cuatrimestre del Ciclo Lectivo 2025, a tenor de la necesidad de introducir cambios de interés académico y en armonía con el resto de las obligaciones curriculares.

Que la SECRETARÍA ACADÉMICA de la UNIVERSIDAD ha emitido opinión favorable, de conformidad con lo previsto en el artículo 3° de la Parte I del citado REGLAMENTO GENERAL, por cuanto dicho Programa se ajusta a las definiciones enunciadas en el artículo 4° de la Parte I del REGLAMENTO en cuestión, así como también, respecto de las demás disposiciones reglamentarias previstas en el mismo.

Que la SECRETARÍA LEGAL Y TÉCNICA ha tomado la intervención de su competencia.

Que el CONSEJO del DEPARTAMENTO DE CIENCIAS APLICADAS Y TECNOLOGÍA, en Sesión Ordinaria N°05/25 de fecha 09 de junio de 2025, trató y aprobó la decisión propiciada, conforme lo establecido en el artículo 2° de la Parte I del REGLAMENTO GENERAL ACADÉMICO.

Por ello,

EL CONSEJO DEL DEPARTAMENTO DE CIENCIAS APLICADAS Y
TECNOLOGÍA de la UNIVERSIDAD NACIONAL DE MORENO
DISPONE:

ARTÍCULO 1°.- Dejar sin efecto, a partir del 1er. Cuatrimestre de Ciclo Lectivo 2025, la Disposición UNM-DCAYT N° 03/14.

ARTÍCULO 2°.- Aprobar la modificación del Programa de la Asignatura ENTORNO C y JAVA (2043R), del ÁREA: REDES, correspondiente al CICLO SUPERIOR de la Carrera INGENIERÍA EN ELECTRÓNICA del DEPARTAMENTO DE CIENCIAS APLICADAS Y TECNOLOGÍA, de esta UNIVERSIDAD, aconsejando su aprobación con vigencia a partir del 1er. Cuatrimestre del Ciclo Lectivo 2025, el que como Anexo I forma parte integrante de la presente Disposición.

ARTÍCULO 3°.- Regístrese, comuníquese, dese a la SECRETARÍA ACADÉMICA a sus efectos y archívese. -

Disposición UNM-DCAYT N° 36/25

UNIVERSIDAD NACIONAL DE MORENO
PROGRAMA ASIGNATURA: ENTORNO C Y JAVA (2043R)

Carrera: INGENIERÍA EN ELECTRÓNICA (Plan de estudios aprobado por Resolución UNM-R N° 21/10 y sus modificatorias UNM-R N° 407/11, UNM-R N° 239/13, UNM-R N° 39/16 y UNM-R N° 395/17)¹

Área: Redes

Trayecto Curricular: Ciclo Superior

Período: 1° y 2° Cuatrimestre - Año 4

Modalidades: Presencial o semipresencial

Carga Horaria: 96 (noventa y seis) horas con un máximo de hasta 48 (cuarenta y ocho) horas virtuales.

Vigencia: A partir del primer cuatrimestre 2025

Clases: 32 (treinta y dos)

Régimen: de regularidad o libre

Responsable de la asignatura: Ing. Fernando Gabriel Rodríguez

Programa elaborado por: Ing. Fernando Gabriel Rodríguez

Fundamentación:

Esta asignatura pretende introducir a las/os estudiantes de Ingeniería en Electrónica, quienes naturalmente cuentan con conocimientos de programación estructurada propios de la carrera, en la POO (Programación Orientada a Objetos).

Debido a los avances tecnológicos y la aparición de nuevos hardware, cada vez más resulta necesario adquirir este conocimiento para adecuar el desarrollo de componentes que soportan este paradigma.

Objetivos Generales

- Formar al alumno en el conocimiento del lenguaje orientado a Objetos C++.
- Formar al alumno en el conocimiento del lenguaje JAVA.
- Desarrollar actividades prácticas para aplicar los conocimientos adquiridos.

Contenidos Mínimos:

El lenguaje C++ y el framework de .net. Creación de Programas Simples en C++. Programación orientada a objetos. Clases y Objetos. Herencia y polimorfismo. Constructores: definición, llamada, constructor por defecto. Creación de objetos (operador new) . Operaciones de conversión (explícita, uso de as). Definir y Utilizar una Clase. Hacer Override a un Método. Introducción a Java. Herencia y métodos virtuales. Concepto de herencia. Clase base, clase hija, sintaxis de clases derivadas. Polimorfismo. Métodos genéricos. Operador is. Arreglos y Colecciones. Clase StringBuilder. Arrays y Serialización. Arrays unidimensionales y multidimensionales y anidados. Serialización X M L. Vectores. Matrices. Alcance de objetos y Reciclado de Memoria. Control de

¹ Reconocimiento oficial y validez nacional otorgado por Resolución ME No 2287/2013 y su modificatoria R.M N° 534/2021

Acceso. Variables y Métodos. This y super. Métodos Getters y Setters. Argumentos y valores de retorno. Ámbito. Clase Abstractas. Interfaces. Arrays de datos primitivos y de objetos. Serialización de Objetos. Construcción de GUI en Java. Generación de interfaces de usuario con NetBeans. AWT (Abstract Windows Toolkit). SQL. Hilos. Procesos y hebras. Aplets.

Programa:

Unidad 1: Concepto de Clase, Encapsulamiento y Abstracción.

Introducción a la POO y sus pilares. Diferencias, ventajas y desventajas con la programación estructurada clásica. Definición de clase, atributo miembro, método miembro, accesibilidad y encapsulamiento. Concepto del encapsulamiento como un pilar de la POO. Clase miembro (anidada), Concepto de abstracción como un pilar de la POO. Abstracción del mundo real a una clase. Uso y aplicación del operador this. Operadores de accesibilidad. (public, private, protected).

Unidad 2: Instanciación de objetos y Herencia.

Instanciación de objetos de una clase. Ubicación en memoria de un objeto (Stack o Heap). Concepto de herencia como un pilar de la POO. Constructores, multiplicidad de constructores. Cadena de llamadas de constructores. Constructores copia. Definición de accesibilidad de los constructores. Destruidores, utilidad de los destructores. Cadena de llamadas de destructores. Uso de la sentencia delete. Concepto de Overloading (sobrecarga).

Unidad 3: Tipos de Herencia.

Profundización del concepto de herencia, Objeto miembro. Herencia publica, herencia privada y herencia protegida. Herencia múltiple y la razón de su poco uso. Accesibilidad a los miembros heredados en los diferentes tipos de herencia. Concepto de Overwriting (sobre-escritura), Constructores declarados como public, private y protected, su utilidad y forma de uso. Tiempo de vida de un objeto (alcance) y su relación con el destructor.

Unidad 4: Polimorfismo.

Concepto de polimorfismo como un pilar de la POO. Utilidad del polimorfismo. Cuidados en el uso de polimorfismo. Concepto de Memory Leaks. Utilidad de operador new y delete en la aplicación de polimorfismo. Clases abstractas. Miembros virtuales y virtuales puros. Uso de punteros a objetos. Simulación de una interface mediante el uso de una clase abstracta. Objetos en el Stack versus objetos en el Heap.

Unidad 5: Misceláneas C++ y POO.

Concepto y uso de excepciones. Concepto, uso y compilación de DLLs para la reutilización del código. Clases y miembros friends. Concepto de violación del encapsulamiento mediante el uso de friends. Uso de templates para la reutilización de miembros y clases como forma de reutilización dinámica del código. Serialización, Uso de palabra reservada extern, Concepto de namespaces y su uso para la clasificación y agrupamiento lógico de funcionalidades. Uso de Collections para la simplificación y optimización del código. Sobrecarga de operadores. El lenguaje C++ como un lenguaje compilado de portabilidad limitada por la compilación.

Unidad 6: El lenguaje Java.

Estructura del lenguaje Java. El JRE de Java como herramienta de portabilidad. El api de Java utilización y documentación. El SDK de Java como entorno de desarrollo. Ventajas sobre lenguajes compilados sin entorno de ejecución (ej. C++). El IDE Eclipse como entorno de desarrollo.

Unidad 7: Aplicación de la POO en Java.

Aplicación de todos los conceptos aprendidos de la POO, sobre el lenguaje Java. Análisis de las restricciones en este lenguaje con respecto al acceso al hardware, la prohibición de la herencia múltiple. Análisis de la correspondencia en la sintaxis del lenguaje en relación con el lenguaje C++. Aplicación de interfaces como un tipo de Polimorfismo no presente en C++. Uso y aplicación de packages. Compilación e instalación de una aplicación Java.

Objetivos de aprendizaje:

- Comprender los pilares del paradigma de la (POO) Programación Orientada a Objetos a fin de poder aplicarlos en cualquier lenguaje orientado a objetos.
- Distinguir las diferencias, ventajas y desventajas de la POO con respecto a la programación estructurada.
- Ejercitar la práctica de programación en C++ aplicando los pilares de la POO.
- Abordar Práctica de programación en Java aplicando los pilares de la POO.

Bibliografía:

García de Jalón, J., Rodríguez Iñigo Mingo, J., Brazález, A., Larzabal, A., Calleja, J., García, J. (2009). "*Aprenda Java como si estuviera en primero*" - Escuela Superior de Ingenieros Industriales de San Sebastián UNIVERSIDAD DE NAVARRA.

Love, D. (2014). "*Java all in one*". John Wiley & Sons Inc. 4th edition.

Metodología de trabajo:

La materia está organizada por temáticas basadas en fundamentos teóricos, aplicaciones, ejercitación y programación. Para cada temática se plantearán los conceptos teóricos y se analizará una serie de ejercicios prácticos semi-resueltos, provistos por la materia, donde se podrá visualizar la aplicación de cada concepto. Los ejercicios para el primer cuatrimestre están escritos en el lenguaje C++, los mismos irán aumentando en complejidad a medida que se aplique los temas teóricos. Las unidades 1 a 5 se dictarán en el primer cuatrimestre donde se estudiará toda la teoría concerniente a la POO aplicada en el lenguaje C++. Para ello se utilizará el IDE "Dev – C++" versión 5.11 por su simplicidad y robustez.

Las unidades 6 y 7 se dictarán en el segundo cuatrimestre, donde se aplicará todo lo aprendido sobre la POO, esta vez sobre el lenguaje Java. Se abordarán algunos temas específicos del lenguaje relativos a la sintaxis y organización del código. Para ello se utilizará el IDE "Eclipse" en su última versión.

En modalidad virtual sincrónica se desarrollarán las clases teóricas, a través de una plataforma Moodle de la UNM, mientras que los trabajos prácticos se realizarán en el laboratorio, como así también las evaluaciones, de manera presencial.

Esta modalidad, a distancia, permite que las/os estudiantes puedan interactuar con el/la docente de la misma manera que en un salón de clases tradicional con la siguiente ventaja:

* Todas las clases serán grabadas y compartidas por medio del campus virtual de modo tal que, si un/a estudiante por cualquier motivo no puede asistir puede ver el video y no pierde la misma. Además, es un recurso que permitirá atender las dudas y a preparar los exámenes.

* Se utilizará el recurso Foro de Consultas para que las/os alumnas/os puedan plantear las dudas o inquietudes que se les pudieran presentar durante el desarrollo del curso. Además del intercambio que se produzca en las clases presenciales, la plataforma permitirá también mantener una comunicación fluida entre docentes y alumnos/as, a través de chats, mensajes en los foros y, también, mediante comunicaciones sincrónicas a través del recurso previsto en la plataforma.

Evaluación y aprobación:

La evaluación consta de dos exámenes parciales presenciales. Los parciales se aprobarán con una nota mínima de cuatro (4), lo que dará derecho a rendir el examen final que se aprobará con un mínimo de cuatro (4).

Cada parcial podrá ser recuperado un máximo de 2 (dos) veces. Asimismo, el/la alumno/a podrá rendir el examen final hasta en 3 (tres) fechas destinadas a tal efecto.

Para la aprobación de la asignatura, se adoptan las siguientes modalidades:

– Por promoción directa

Para la aprobación de la asignatura por el Sistema de Promoción al finalizar el Curso Lectivo, el/la alumno/a deberá contar con los siguientes requisitos:

- a. Asistencia 80 % (setenta y cinco por ciento) clases prácticas presenciales.
- b. Aprobación 100 % (cien por ciento) de los Trabajos Prácticos (TPs) y parcial/es con calificación 7 (siete) o superior en cada uno. Cada TPs, contará con una instancia de recuperación.

– Con examen final, como alumno regular

Cuando el/la alumno/a cumpla con los requisitos a y b, pero apruebe el/los parcial/es con calificación igual o mayor que 4 (cuatro) y menor que 7 (siete) podrá aprobar la asignatura mediante un examen final de acuerdo con la normativa de la UNM.

– Por examen libre

De acuerdo con normativa vigente.