Resolución de Consejo Directivo **828 / 2026 - EXA -UNSa**
EXP. 552/2026 Dr. Edgardo Javier TRENTI eleva Programa Analítico y Régimen de Regularidad y Promoción de la Asignatura TEORÍA DE LA COMPUTACIÓN I de Licenciatura en Análisis de Sistemas, Plan 2010
De: EXACTAS-Secretaria Académica y de Investigación

Salta,
02/11/2026

VISTO: La nota presentada por el Dr. Edgardo Javier TRENTI mediante la cual solicita la aprobación del Programa Analítico y Régimen de Regularidad y Promoción de la asignatura TEORÍA DE LA COMPUTACIÓN I de la carrera Licenciatura en Análisis de Sistemas Plan 2010, y

CONSIDERANDO:

Que la Comisión de Carrera de la Licenciatura en Análisis de Sistemas y el Departamento de Informática emiten opinión favorable para la aprobación del Programa Analítico y del Régimen de Regularidad y Promoción.

Que la Comisión de Docencia e Investigación aconseja aprobar el Programa Analítico y el Régimen de Regularidad y Promoción.

Que la RCD 32/2026 EXA-UNSa dispone un formato único para la presentación de los programas de cada una de las asignaturas de las carreras que se dictan en esta Unidad Académica.

Que el Consejo Directivo, en su 18° Sesión Ordinaria realizada el 21 de octubre de 2026, aprobó el despacho de la Comisión de Docencia e Investigación.

Que, el Estatuto de la Universidad Nacional de Salta en el Artículo 117 inciso 8, establece que "*entre los deberes y atribuciones que le confiere al Consejo Directivo, incluye aprobar los programas Analíticos y la reglamentación sobre el régimen de regularidad y promoción propuesto por los módulos Académicos*".

Por ello,

EL CONSEJO DIRECTIVO DE LA FACULTAD DE CIENCIAS EXACTAS

R E S U E L V E:

ARTÍCULO 1°.- Aprobar el Programa Analítico, Régimen de Regularidad y Promoción de la asignatura TEORÍA DE LA COMPUTACIÓN I de la Carrera Licenciatura en Análisis de Sistemas (Plan 2010), que como Anexo forma parte integrante de la presente Resolución.

ARTÍCULO 2°.- Notifíquese al Dr. Edgardo Javier TRENTI, docente responsable de la asignatura TEORÍA DE LA COMPUTACIÓN I. Comuníquese a las Autoridades y Secretarías de esta Unidad Académica, a la Comisión de Carrera de Licenciatura en Análisis de Sistemas, al Departamento de Informática, a la Dirección de Mesa de Entradas, Archivo y Digesto y a la Dirección de Alumnos para su toma de conocimiento, registro y demás efectos. Publíquese en el Boletín Oficial y en la Página WEB de la Facultad. Cumplido, archívese.

MFL

FACULTAD DE CIENCIAS EXACTAS UNSa

LIC. MARCELA F. LÓPEZ
SECRETARIA ACADÉMICA Y DE INVESTIGACIÓN
FACULTAD DE CS. EXACTAS - UNSa

UNIVERSIDAD NACIONAL DE SALTA Facultad Cs. Exactas

Dr. JOSÉ RAMÓN MOLINA
DECANO
FACULTAD DE CS. EXACTAS - UNSa

1. **DEPARTAMENTO DOCENTE O DEPENDENCIA:** Departamento de Informática
2. **CÁTEDRA:** TEORÍA DE LA COMPUTACIÓN
3. **ASIGNATURA:** TEORÍA DE LA COMPUTACIÓN I
4. **CARRERA** Licenciatura en Análisis de Sistemas Plan 2010 (Res CS Nº 262/2012)
5. **ÚLTIMO PROGRAMA VIGENTE:** RESD-EXA Nº 747/2019 – RESCD-EXA Nº 248/2022
6. **EQUIPO DOCENTE**

Profesor Responsable:	Edgardo Javier Trenti
Profesores Auxiliares:	María Laura Massé Palermo
	Darío R. Ledesma

7. **UBICACIÓN EN EL PLAN DE ESTUDIO**

Año de cursado:	Segundo Año
Cuatrimestre:	Primer Cuatrimestre
Carga horaria total:	120hs
Carga horaria semanal:	Teoría 4hs - Práctica 4hs

8. **CONTENIDOS MÍNIMOS**

Lenguajes formales y autómatas. Minimización de Autómatas. Expresiones regulares. Máquinas de Turing. Clasificación de Autómatas. Autómata de Pila. Lenguajes Formales. Jerarquía de Chomsky. Gramáticas e Isomorfismo.

DISTRIBUCIÓN DE LOS CONTENIDOS MÍNIMOS EN EL PROGRAMA

Tema	Unidad
Lenguajes formales y autómatas	UNIDAD 1
Minimización de Autómatas	UNIDAD 2
Expresiones regulares	UNIDAD 2
Máquinas de Turing	UNIDAD 4
Clasificación de Autómatas	UNIDAD 1
Autómatas de Pila	UNIDAD 3
Lenguajes Formales	UNIDAD 1

FACULTAD DE CIENCIAS EXACTAS UNSa

Jerarquía de Chomsky	UNIDAD 4
Gramáticas e Isomorfismo	UNIDAD 2

9. CORRELATIVIDADES

Según Res CS N.º 262/2012 (Plan de Estudio vigente).

Para cursar:

- **Regularizadas:** Matemática para Informática, Programación
- **Aprobadas:** ---

Para rendir el examen final:

- **Aprobadas:** Matemática para Informática, Programación

10. JUSTIFICACIÓN

(a) Justificación de la asignatura

Las Ciencias de la Computación proporcionan el marco teórico para el estudio de los algoritmos, los lenguajes de programación y los dispositivos de cómputo. Dentro de este campo se distinguen tres áreas fundamentales: la teoría de autómatas, la teoría de la computabilidad y la teoría de la complejidad computacional.

Esta asignatura introduce los fundamentos de lenguajes formales junto con los dispositivos de aceptación y/o generación de los mismos: autómatas finitos, expresiones regulares, autómatas de pila, máquinas de Turing y las gramáticas asociadas. Estos conceptos permiten comprender distintos niveles de expresividad en los lenguajes formales y su clasificación en la jerarquía de Chomsky.

En la formación del analista de sistemas, estos contenidos brindan bases teóricas para comprender el funcionamiento y diseño de lenguajes de programación, compiladores y sistemas de procesamiento de información, además de constituir el fundamento conceptual para el estudio posterior de la computabilidad y la complejidad computacional.

(b) Articulación Académica

Esta asignatura se articula en forma vertical con Teoría de la Computación II, en donde se desarrollan aplicaciones prácticas de las herramientas teóricas vistas en esta asignatura. También articula con Teoría de la Computación III, donde se repasa el dispositivo de Máquinas de Turing como base para estudiar computabilidad y complejidad computacional. Además, se establece una articulación vertical con la asignatura Programación para el desarrollo de estructuras de datos dinámicas.

(c) Aporte al Perfil Profesional

- Diseño y desarrollo de modelos matemáticos para cualquier sistema de información destinado a resolución de problemas científicos, técnicos o administrativos.

FACULTAD DE CIENCIAS EXACTAS UNSa

- Investigaciones conducentes a la creación y/o mejoramiento de técnicas de desarrollo de sistemas de información y nuevas aplicaciones de la tecnología informática.
- Enseñanza de los conocimientos básicos, técnicos y científicos de los temas contenidos en la carrera, en todos los niveles de conformidad a las respectivas reglamentaciones. Investigaciones relacionadas con estos conocimientos.

(d) Ejes Transversales Abordados

- ET01 Identificación, formulación y resolución de problemas de informática.
- ET04 Utilización de técnicas y herramientas de aplicación en la informática.
- ET10 Fundamentos para el aprendizaje continuo.

11. OBJETIVOS DE LA ASIGNATURA

La asignatura tiene como objetivo abordar los fundamentos teóricos de la computación mediante el estudio de los lenguajes formales y los modelos matemáticos de reconocimiento y generación de lenguajes.

En particular, se propone que el estudiante:

- Comprenda los conceptos fundamentales de lenguajes formales, gramáticas y autómatas como modelos abstractos de sistemas de cómputo.
- Analice las diferentes clases de lenguajes formales y los modelos computacionales asociados.
- Desarrolle habilidades para modelar problemas mediante autómatas, gramáticas y máquinas abstractas.
- Comprenda las capacidades y limitaciones de los distintos modelos de computación.
- Reconozca el rol de estos modelos en áreas de la informática como compiladores, procesamiento de lenguajes, teoría de la computación y diseño de sistemas.
- Desarrolle capacidades de abstracción, formalización y razonamiento matemático aplicadas a problemas informáticos.

12. DESARROLLO DEL PROGRAMA

UNIDAD 1. INTRODUCCIÓN A LOS LENGUAJES FORMALES

Objetivos de aprendizaje

- Comprender el papel de los lenguajes formales y las estructuras discretas en la descripción de procesos computacionales.
- Utilizar la notación y las herramientas matemáticas básicas para representar y analizar lenguajes y estructuras discretas.

FACULTAD DE CIENCIAS EXACTAS UNSa

Contenidos

Motivación para el estudio de los lenguajes formales y los modelos de computación. Conceptos básicos: alfabetos, cadenas y lenguajes formales. Operaciones sobre lenguajes. Representación de estructuras discretas: grafos y árboles. Herramientas matemáticas básicas: relaciones, funciones discretas y conjuntos infinitos. Métodos de demostración. Lenguajes formales. Clasificación de autómatas. Modelos formales de computación: dispositivos de aceptación, generación y traducción.

UNIDAD 2. LENGUAJES REGULARES

Objetivos de aprendizaje

- Analizar el reconocimiento de lenguajes regulares mediante autómatas finitos.
- Aplicar técnicas de transformación y simplificación de autómatas para estudiar su comportamiento.
- Utilizar y relacionar distintas representaciones formales de los lenguajes regulares.

Contenidos

Lenguajes regulares. Autómatas finitos deterministas y no deterministas. Equivalencia entre autómatas deterministas y no deterministas. Máquinas secuenciales: máquinas de Moore y máquinas de Mealy. Minimización de autómatas finitos. Expresiones regulares. Gramáticas regulares y su relación con los lenguajes regulares. Equivalencia entre autómatas finitos, expresiones regulares y gramáticas regulares. Isomorfismos.

UNIDAD 3. LENGUAJES LIBRES DE CONTEXTO

Objetivos de aprendizaje

- Analizar la estructura de los lenguajes libres de contexto mediante gramáticas y derivaciones.
- Aplicar técnicas de representación y simplificación de gramáticas para estudiar sus propiedades.
- Comprender la relación entre gramáticas libres de contexto y los modelos de autómatas de pila.

Contenidos

Gramáticas libres de contexto. Sistemas de producciones. Derivaciones y árboles de derivación. Ambigüedad en gramáticas. Simplificación de gramáticas libres de contexto: eliminación de símbolos inútiles. Autómatas de pila deterministas y no deterministas. Lenguajes aceptados por autómatas de pila deterministas y no deterministas. Equivalencia entre gramáticas libres de contexto y autómatas de pila.

UNIDAD 4. LENGUAJES RECURSIVAMENTE ENUMERABLES

Objetivos de aprendizaje

- Comprender el modelo de máquina de Turing como formalización del concepto de computación.

FACULTAD DE CIENCIAS EXACTAS UNSa

- Analizar la noción de límite de los dispositivos de cómputo y los lenguajes reconocibles por los mismos.
- Reconocer la jerarquía de lenguajes formales como marco de clasificación de los distintos modelos de computación.

Contenidos

Modelo de la máquina de Turing. Definiciones formales y funcionamiento. Construcción de máquinas de Turing. Lenguajes recursivamente enumerables. Funciones computables. Hipótesis de Church–Turing. Clasificación de lenguajes formales: jerarquía de Chomsky.

13. PROGRAMA DE TRABAJOS PRÁCTICOS

Trabajo Práctico N° 1: Nociones de Conjuntos. Nociones de Lenguajes Formales.

Trabajo Práctico N° 2: Autómatas Finitos. Máquinas Secuenciales. Expresiones Regulares. Gramáticas Regulares.

Trabajo Práctico N° 3: Gramáticas Libres de Contexto. Autómatas de Pila.

Trabajo Práctico N° 4: Máquinas de Turing.

Distribución temporal tentativa (15 semanas)

Tener en cuenta que por semana se dictan dos clases prácticas de 2hs cada una.

TP	Tema abordado	Objetivo	Unidad	Horas de Interacción Pedagógica	Clases estimadas
1	Herramientas matemáticas básicas. Lenguajes formales y autómatas.	Repaso de las nociones de Conjuntos, cardinalidad, relaciones y funciones. Introducción a lenguajes formales. Alfabetos, cadenas y lenguajes.	Unidad 1	8 horas	1 a 4
2	Autómatas finitos, expresiones regulares. Equivalencias.	Desarrollar capacidades de abstracción y lógica mediante la construcción de autómatas finitos deterministas y no deterministas. Aprender a minimizar autómatas y distintos dispositivos equivalentes.	Unidad 2	22 horas	5 a 15
3	Lenguajes libres de contexto. Gramáticas.	Practicar la construcción de Gramáticas Libres de Contexto. Conceptos de ambigüedad y simplificación de GLCs. Comprender las relaciones entre GLCs y autómatas de pila.	Unidad 3	20 horas	16 a 25

FACULTAD DE CIENCIAS EXACTAS UNSa

4	Máquinas de Turing y Jerarquía de Chomsky.	Adquirir la capacidad de formalizar procesos a través de máquinas de Turing. Clasificar lenguajes según la jerarquía de Chomsky.	Unidad 4	10 horas	26 a 30

14. LABORATORIOS

Trabajo Práctico Transversal

Durante el cursado se desarrollará a lo largo del cuatrimestre un Trabajo Práctico Transversal (TPT) para la implementación de estructuras de datos y algoritmos relacionados con los modelos formales estudiados en la asignatura.

El objetivo general del TPT es diseñar e implementar, en lenguaje C, un Tipo Abstracto de Datos (TAD) que modele objetos matemáticos como listas y conjuntos, junto con sus operaciones básicas. Este TAD será posteriormente utilizado como soporte para la implementación de algoritmos sobre autómatas finitos, integrando así los conceptos de programación con los modelos formales abordados en la asignatura.

Asimismo, este trabajo establece una articulación vertical con la asignatura previa Programación retomando conceptos de TAD y listas enlazadas y con la asignatura posterior Teoría de la Computación II, en la cual los estudiantes reutilizarán el TAD desarrollado para implementar un pequeño intérprete de un lenguaje de programación que permita manipular listas y conjuntos con distintos niveles de anidamiento. En dicha asignatura se emplearán herramientas de análisis léxico y sintáctico como Flex y Bison integrando el código en C desarrollado en el TAD, lo que permitirá vincular el desarrollo de estructuras de datos y algoritmos con los conceptos de lenguajes formales y compiladores.

Se prevén las siguientes etapas de trabajo.

Etapa 1: Repaso de estructuras dinámicas.
Implementación en lenguaje C de estructuras dinámicas básicas utilizando listas enlazadas y asignación dinámica de memoria. Modelado de cadenas de caracteres y colecciones de elementos.

Etapa 2: Implementación del TAD de listas y conjuntos.
Diseño e implementación de un tipo abstracto de datos que represente listas y conjuntos, incluyendo operaciones básicas como inserción, eliminación, pertenencia, unión e intersección.

Etapa 3: Representación de autómatas finitos.
Utilización de las estructuras desarrolladas para representar autómatas finitos deterministas y no deterministas.

Etapa 4: Implementación de algoritmos sobre autómatas.
Implementación de algoritmos básicos de manipulación de autómatas finitos utilizando el TAD desarrollado.

FACULTAD DE CIENCIAS EXACTAS UNSa

Modalidad de Trabajo

El desarrollo de los laboratorios contará con guías de trabajo que serán introducidas y orientadas durante las horas de clases prácticas. No obstante, la implementación completa de las actividades requerirá trabajo adicional por parte de los estudiantes fuera del aula. Para ello se prevé acompañamiento mediante horarios de consulta, foros y grupos de comunicación vía la plataforma Telegram. En consecuencia, estas actividades suponen una importante carga de estudio y desarrollo autónomo, promoviendo la capacidad de los estudiantes para profundizar contenidos, resolver problemas de programación y consolidar los conceptos teóricos de la asignatura.

El trabajo se desarrollará preferentemente en modalidad grupal, con el propósito de fomentar el intercambio de ideas, la discusión de soluciones y el aprendizaje colaborativo. Esta modalidad busca fortalecer habilidades de trabajo en equipo, organización del desarrollo de software y comunicación técnica, competencias relevantes para la formación profesional en informática.

Objetivos de aprendizaje

Al finalizar el trabajo práctico transversal, se espera que los estudiantes sean capaces de: Reforzar los conceptos de Tipos Abstractos de Datos (TAD) mediante el modelado de cadenas de caracteres, listas y conjuntos; fortalecer el uso de la asignación dinámica de memoria y de punteros en la implementación de estructuras de datos dinámicas.; implementar algoritmos básicos de manipulación de autómatas finitos utilizando las estructuras desarrolladas; integrar herramientas de programación con los conceptos teóricos de lenguajes formales y autómatas; desarrollar estrategias de resolución de problemas de programación de manera autónoma; y trabajar de manera colaborativa en el diseño, implementación y validación de soluciones informáticas.

Resultados esperados

Al finalizar el TPT, los estudiantes deberán contar con un programa en lenguaje C que implemente los tipos abstractos de datos necesarios y permita representar y manipular autómatas finitos, incluyendo algunas de las operaciones y algoritmos estudiados durante el curso. El diseño modular del TAD permitirá además su reutilización en asignaturas posteriores.

Criterios de evaluación del Trabajo Práctico Transversal

La evaluación del trabajo considerará la correcta implementación de las estructuras de datos y operaciones definidas, el adecuado uso de memoria dinámica y punteros, la correcta implementación de los algoritmos sobre autómatas, la claridad, organización y documentación del código, y el funcionamiento correcto del programa ante distintos casos de prueba. Se solicitará al finalizar el desarrollo que los estudiantes realicen una exposición oral del trabajo realizado.

15. ESTRATEGIAS DE ENSEÑANZA

La asignatura se desarrolla mediante una combinación de instancias de construcción teórica y actividades prácticas, orientadas a promover la comprensión conceptual, el razonamiento formal y la aplicación de los modelos estudiados.

Clases de construcción teórica

En estas instancias se introducen y desarrollan los conceptos fundamentales de la asignatura, estableciendo relaciones entre los distintos temas y su relevancia para la informática teórica. El

FACULTAD DE CIENCIAS EXACTAS UNSa

marco conceptual se formaliza mediante definiciones, proposiciones, lemas y teoremas, promoviendo la comprensión de los modelos matemáticos utilizados para describir y analizar procesos computacionales. Se fomenta la participación de los estudiantes a través de preguntas orientadoras, análisis de ejemplos y discusión de los resultados obtenidos.

Clases prácticas

Las actividades prácticas se orientan a la resolución de problemas y la construcción y análisis de modelos formales En estas instancias los estudiantes aplican las herramientas teóricas para representar lenguajes formales, diseñar y analizar autómatas y estudiar propiedades de los distintos modelos de computación. Se promueve el trabajo guiado, la discusión de estrategias de resolución y la puesta en común de resultados.

Integración teoría–práctica

Las actividades se organizan de modo articulado para favorecer la progresiva apropiación de los conceptos y el desarrollo de habilidades de razonamiento formal. A lo largo de la asignatura se utilizan ejemplos, ejercicios integradores y trabajos prácticos que permiten relacionar los distintos contenidos y consolidar los aprendizajes. La inclusión de un proyecto a modo de Trabajo Práctico Transversal apoya la integración de conocimientos.

16. ESTRATEGIAS DE EVALUACIÓN

La evaluación de los aprendizajes se realiza de manera continua e integral, mediante distintos instrumentos que permiten valorar la comprensión conceptual, la capacidad de razonamiento formal y la aplicación de los modelos estudiados.

Instrumentos de evaluación

- **Evaluaciones parciales escritas:** Se toman dos evaluaciones parciales durante el cursado. Cada parcial incluye un bloque teórico, orientado a evaluar la comprensión de los conceptos y resultados fundamentales, y un bloque práctico, destinado a la resolución de problemas y al análisis de modelos formales.
- **Coloquios o instancias de evaluación formativa:** Se realizan coloquios o actividades evaluativas previas a los parciales, orientadas a verificar el avance en la comprensión de los contenidos y a habilitar la instancia de evaluación parcial.
- **Trabajos prácticos**: A lo largo del cursado se desarrollan trabajos prácticos orientados a la resolución de problemas y al análisis de los distintos modelos de computación. Se incluye un trabajo práctico transversal (descrito en la sección 14. LABORATORIOS), orientado a reforzar y vincular conceptos teóricos con prácticas de programación.
- **Examen final:** La aprobación de la asignatura se realiza mediante un examen final integrador que evalúa los principales contenidos del programa. El examen puede ser oral o escrito.

Criterios de evaluación

En las distintas instancias de evaluación se valorará la comprensión de los conceptos fundamentales de la asignatura, la capacidad para modelar y analizar problemas mediante lenguajes formales y dispositivos de cómputo y el uso adecuado de la notación y el lenguaje formal propio del área. Asimismo, se considerará la correcta aplicación de definiciones, teoremas

FACULTAD DE CIENCIAS EXACTAS UNSa

y propiedades, la claridad y coherencia en el desarrollo de demostraciones y resoluciones, y la capacidad de argumentación y justificación de los resultados obtenidos.

17. RECURSOS MATERIALES

Para el normal desarrollo de la asignatura se precisa de aulas con capacidad amplia, pizarrón y un curso en la plataforma Moodle de la Facultad. Instalación en laboratorios de software para el desarrollo del Trabajo Práctico Transversal. Proyector para las clases teóricas. A estos recursos, se suman diapositivas, apuntes y videos educativos elaborados por la cátedra.

18. CONDICIONES PARA REGULARIZAR

Para obtener la condición de estudiante regular se requiere:

- Aprobar dos evaluaciones parciales o sus respectivas instancias de recuperación;
- Aprobar cada uno de los bloques (teórico y práctico) que integran las evaluaciones parciales;
- Participar y aprobar los coloquios o instancias evaluativas e formación establecidos por la cátedra;
- Cumplir con al menos el 75 % de asistencia a las clases prácticas;
- Presentar y aprobar el Trabajo Práctico Transversal.

19. CONDICIONES PARA ACREDITAR

La asignatura se aprueba mediante un examen final, que evalúa de manera integrada los contenidos desarrollados durante el cursado. El examen se organiza en tres bloques temáticos correspondientes a: lenguajes regulares; lenguajes libres de contexto; y lenguajes recursivamente enumerables.

Para aprobar el examen final es necesario aprobar cada uno de los bloques que lo integran.

El estudiante libre deberá aprobar una primera instancia práctica, orientada a verificar competencias equivalentes a las requeridas para la regularización de la asignatura (aprobación de trabajos prácticos, resolución de problemas y Trabajo Práctico Transversal). Una vez superada dicha instancia, accederá a una segunda instancia teórica, cuya modalidad es equivalente a la aplicada a los estudiantes regulares.

20. BIBLIOGRAFÍA

a. Bibliografía específica

Hopcroft, J., Motwani, R. y Ullman, J. (2008). *Introducción a la teoría de autómatas, lenguajes y computación*. 3ª ed. Pearson – Addison Wesley.

Hopcroft, J. y Ullman, J. (1979). *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley.

Linz, P. y Rodger, S. H. (2023). *An Introduction to Formal Languages and Automata*. 7ª ed. Jones & Bartlett Publishers.

Greenberg, N., Jain, S., Ng, K. M., Schewe, S., Stephan, F., Wu, G., & Yang, Y. (Eds.). (2023). *Aspects of computation and automata theory with applications*. World Scientific.

Pettorossi, A. (2022). *Automata Theory and Formal Languages: Fundamental Notions, Theorems, and Techniques*. Springer, Cham. https://doi.org/10.1007/978-3-031-11965-1

b. Bibliografía complementaria

Brookshear, G. (2000). *Teoría de la computación: lenguajes formales, autómatas y complejidad*. Addison-Wesley.

Brena, R. (2014). *Autómatas y lenguajes: un enfoque de diseño*. McGraw-Hill.

Kelley, D. (1995). *Teoría de autómatas y lenguajes formales*. Prentice-Hall.

Viso Gurovich, E. (2008). *Introducción a la teoría de la computación*. México: Las Prensas de Ciencias.

Sipser, M. (2005). *Introduction to the Theory of Computation*. 2ª ed. Boston: Thomson.

FACULTAD DE CIENCIAS EXACTAS UNSa

LIC. MARCELA F. LÓPEZ
SECRETARIA ACADÉMICA Y DE INVESTIGACIÓN
FACULTAD DE CS. EXACTAS - UNSa

UNIVERSIDAD NACIONAL DE SALTA Facultad Cs. Exactas

Dr. JOSÉ RAMÓN MOLINA
DECANO
FACULTAD DE CS. EXACTAS - UNSa