



Universidad Nacional de Río Cuarto
Facultad de Ciencias Exactas, Físico-Químicas y Naturales
HOMENAJE AL PREMIO NOBEL

2021 - "AÑO DE
DE MEDICINA DR. CÉSAR MILSTEIN"

FORMULARIO PARA LA PRESENTACIÓN DE PROGRAMAS DE ASIGNATURAS en
el CONTEXTO DE PANDEMIA por Covid-19¹
Año Lectivo: 2021

UNIVERSIDAD NACIONAL DE RÍO CUARTO
FACULTAD DE FACULTAD DE CIENCIAS EXACTAS, FÍSICO-QUÍMICAS Y
NATURALES
DEPARTAMENTO DE ...Computación

CARRERA/S: Analista en Computación (Cód. 12), Profesorado en Ciencias de la Computación (Cód. 13), Licenciatura en Ciencias de la Computación (Cód. 14).

PLAN DE ESTUDIOS: 1999 Versión 1 (para las 3 carreras)

ASIGNATURA: Análisis Comparativo de Lenguajes **CÓDIGO:** 1956

MODALIDAD DE CURSADO: Presencial

DOCENTE RESPONSABLE: Ariel Gonzalez, Mg. en Computación, Prof. Adjunto Excl.

EQUIPO DOCENTE:

Maria M. Novaira (Lic. en Cs de la Computación, JTP Excl.)

Luciano Putruele (Lic. en Computación, Ay1 Simple)

Sandra Angeli (Prof en Computación, AY1 Semi-Excl.)

RÉGIMEN DE LA ASIGNATURA: cuatrimestral

UBICACIÓN EN EL PLAN DE ESTUDIO: tercer Año (en todas las carretas)

RÉGIMEN DE CORRELATIVIDADES: (para cursado, según plan de estudio vigente)

Asignaturas aprobadas: (nombre y código)

1948 – Programación Avanzada

1949 – Organización del Procesador

Asignaturas regulares:

1948 – Programación Avanzada

1949 – Organización del Procesador

CARÁCTER DE LA ASIGNATURA:Obligatoria

¹ Res. CS 120/2017 y Res. CD 049/2020



Universidad Nacional de Río Cuarto
Facultad de Ciencias Exactas, Físico-Químicas y Naturales
HOMENAJE AL PREMIO NOBEL

2021 - "AÑO DE
DE MEDICINA DR. CÉSAR MILSTEIN"

CARGA HORARIA TOTAL: 112 horas (según el plan de estudio vigente)

Teóricas:	42 hs	Prácticas:	70 hs	Teóricas - Prácticas:	... hs	Laboratorio:	... hs
------------------	--------------	-------------------	--------------	----------------------------------	---------------	---------------------	---------------

CARGA HORARIA SEMANAL: 8 horas (según el plan de estudio vigente)

Teóricas:	3 hs	Prácticas:	5 hs	Teóricas - Prácticas:	... hs	Laboratorio:	... hs
------------------	-------------	-------------------	-------------	----------------------------------	---------------	---------------------	---------------



1. CONTEXTUALIZACIÓN DE LA ASIGNATURA

La asignatura pertenece al segundo cuatrimestre del tercer año de cursado para las tres carreras (Analista, Licenciatura y Profesorado).

Los temas tratados incluyen el estudio de las herramientas de programación (lenguajes, compiladores, etc.) necesarias para el correcto desenvolvimiento en el ámbito laboral.

Es una de las asignaturas que mas se vincula con el resto de las tres carreras y forma la base para que un egresado puede profundizar los conocimientos de cualquier lenguaje de programación.

2. OBJETIVOS PROPUESTOS

Lograr que el alumno adquiriera la capacidad crítica en base a fundamentos técnicos sobre el análisis y elección del lenguaje de programación mas adecuado de acuerdo al tipo de problema. Además se espera que el alumno consiga adquirir los conocimientos básicos sobre los detalles de implementación de los diferentes aspectos de un lenguaje de programación.

Los temas tratados se fundamentan en la necesidad de comprender los conceptos subyacentes de los lenguajes de programación, para su posterior selección y utilización en la resolución de problemas.

La heterogeneidad y complejidad de los problemas actuales, obligan al profesional a estar al tanto de las herramientas de programación que automatizan los procesos del negocio

En este sentido, los lenguajes de programación son la herramienta principal del programador, y pueden ser categorizados de acuerdo a sus estilos o paradigmas. La comprensión de las características, técnicas de programación y técnicas de verificación de programas en cada uno de estos estilos, es de suma importancia para una selección adecuada del lenguaje que implementará la solución de un problema determinado. La correcta elección de un lenguaje de programación brindará enormes beneficios relacionados con la mantenibilidad de los sistemas, la detección de errores, la legibilidad de los programas y otros más.



Los requisitos mínimos para la comprensión de los temas, incluyen conceptos relacionados a: Administración de Memoria, Estructuras de datos y Algoritmos, una mínima noción de los mecanismos para la definición de lenguajes de programación y la noción de recursividad.

Durante el desarrollo del curso los alumnos resolverán ejercicios prácticos pertinentes, siguiendo el apunte de cátedra elaborado por los propios docentes. La elaboración del apunte sigue el estilo de un libro en cuanto a estructura y descripción de los temas.

Para la comprobación del aprendizaje de los temas, se hará mediante dos exámenes parciales de la parte práctica (con sus respectivos recuperatorios) durante el curso, y posteriormente un examen final que abarcará los conceptos teóricos.

3. EJES TEMÁTICOS ESTRUCTURANTES DE LA ASIGNATURA Y ESPECIFICACIÓN DE CONTENIDOS

3.1. Contenidos mínimos (según plan de estudio vigente)

Los contenidos mínimos son enfocados y divididos en estilos de programación (Paradigmas). El estudio de cada estilo comprende la definición y aplicación de los conceptos subyacentes, y de un conjunto de lenguajes de aplicación de cada paradigma.

Los mismos son estructurados de la siguiente manera:

- Introducción a los Lenguajes y Herramientas de Programación.
- Lenguajes y Modelos de Programación
- El modelo declarativo, Lenguajes Funcionales y Relacionales.
- El modelo con estado (statefull) y Lenguajes de Programación Imperativos.
- Manejo de Memoria.
- Programación Orientada a Objetos.
- Programación Concurrente.

3.2. Ejes temáticos o unidades

Unidad 1: Lenguajes como herramientas de programación. Características generales. Elementos de un lenguaje. Especificación de un lenguaje de programación: Sintaxis y semántica.

Declaraciones, definiciones, expresiones y comandos. Mecanismos para la abstracción funcional



y de datos. Herramientas de programación: intérpretes, compiladores, enlazado (linking). Archivos objeto, bibliotecas y ejecutables.

Unidad 2: Lenguajes y modelos de programación

Modelos o paradigmas de programación. Lenguajes declarativos. Lenguajes con estados. Elementos de un lenguaje de programación. Tipos de datos. Chequeo de tipos. Sistemas de tipos fuertes y débiles. Polimorfismo y tipos dependientes. Seguridad del sistema de tipos. Declaraciones, ligadura y ambientes. Excepciones

Unidad 3: El modelo declarativo. Asignación única. Valores y tipos de datos primitivos y estructurados. Variables e identificadores. Sintaxis y semántica de un lenguaje núcleo declarativo. Adornos sintácticos y abstracciones lingüísticas. Tipado estático y dinámico. Manejo de la memoria. Unificación y ligadura.

Unidad 4: Lenguajes funcionales. El estilo de la programación funcional. Polimorfismo paramétrico. Fundamentos teóricos. Cálculo Lambda y lógica combinatoria. LISP. Lenguajes funcionales modernos: Haskell y ML.

Unidad 5: Programación relacional. El modelo de computación no determinístico. Sentencias *choice* y *fail*. Árbol de búsqueda. Programación lógica. Prolog. Forma clausal. Mecanismos de inferencia. Resolución. Características extra-lógicas: Cut, aritmética, entrada-salida. El problema de la negación.

Unidad 6: El modelo con estado. Extensión del lenguaje núcleo con celdas mutables. Semántica de celdas. Punteros y referencias. Aliasing. Igualdad. Asignación. Razonando con estado. Abstracción procedural. Sobrecarga. Efectos colaterales. Referencia transparental. Lenguajes de programación imperativos. El lenguaje C. Manejo manual de la memoria y sus problemas: Referencias colgadas y lagunas de memoria. Manejo automático de la memoria: Recolectores de basura y contadores de referencias.

Unidad 7: Lenguajes de programación imperativo



Declaraciones. Expresiones y comandos. Introducción al lenguaje C, estructuras, tipos de datos básicos, declaraciones y definiciones. Proceso de compilación. Funciones. Alcance de las declaraciones. Tiempo de vida de las entidades. Operadores. Sentencias de control. Tipos de datos estructurados. Punteros. Manejo de memoria dinámica.

Unidad 8: Manejo de la Memoria

Manejo de la memoria eficiente. Manejo del stack. Implementación del manejo de alcance de ambientes. Valores creados dinámicamente. Manejo del heap. Manejo automático del heap. Algoritmos de recolección de basura

Unidad 9: Programación orientada a objetos. Objetos y clases. Clases como módulos y tipos. Herencia. Sistemas de tipos. Control de acceso a métodos y atributos. Polimorfismo basado en herencia. Ligadura dinámica. Redefinición de métodos. Clases abstractas e interfaces. Herencia múltiple. Implementación de la ligadura dinámica. Polimorfismo por instanciación. Tipos parametrizados. Generecidad (clases parametrizadas). Funciones parametrizadas. Parámetros de tipo. Plantillas (templates). C++ templates. Generecidad en Java. Programación genérica. Computación estática. Herencia y clases parametrizadas.

Unidad 10: Concurrencia y paralelismo. Modelos de memoria compartida y mensajes. Threads, eventos, corrutinas, procesos secuenciales y concurrentes.

Problemas que plantea la concurrencia: no determinismo, dependencia de velocidad, bloqueos (deadlocks). Progreso finito (starvation). Interacción entre procesos. Procesos independientes, competitivos y comunicantes. Primitivas para la creación y destrucción de procesos y threads. Primitivas de bajo nivel de sincronización para exclusión mutua, semáforos. Constructores para el manejo de concurrencia de alto nivel: monitores y rendezvous. Ejemplos en Java y C++ . Concurrencia en Erlang. Mecanismos de comunicación entre procesos: tuberías (pipes), colas de mensajes, FIFOs, protocolos de comunicación entre procesos remotos (TCP/IP). Sockets.

4. ACTIVIDADES A DESARROLLAR



4.1. Actividades en modalidad virtual (modalidades alternativas a la presencialidad).

CLASES TEÓRICAS: Clases presenciales de 3 horas semanales.

CLASES PRÁCTICAS: Clases presenciales de 5 horas semanales divididas en dos clases de 2 y 3 horas. Las actividades prácticas se desarrollan en conjunto. Los ejercicios prácticos consisten en el planteo de problemas y realización de pequeños experimentos en diferentes lenguajes de programación que deberán implementar en el laboratorio.

EXAMENES: Ver punto 11.

4.2. Actividades en la presencialidad

Especificar el conjunto de actividades, que siendo esenciales e irremplazables, no puedan realizarse en modalidades alternativas a la presencialidad (prácticas de laboratorio, salidas de campo, prácticas pre-profesionales, prácticas docentes, entre otras).

CLASES TEÓRICAS: (nómina, modalidad, metodología, recursos y carga horaria semanal)

CLASES PRÁCTICAS: (nómina, modalidad, metodología, recursos y carga horaria semanal)

CLASES DE TRABAJOS PRÁCTICOS DE LABORATORIO: (nómina, modalidad, metodología, recursos y carga horaria)

OTRAS: instancias evaluativas, seminarios, talleres, coloquios, etc. (nómina, modalidad, metodología, recursos y carga horaria)

5. PROGRAMAS Y/O PROYECTOS PEDAGÓGICOS INNOVADORES E INCLUSIVOS

INCORPORA AQUÍ EL TEXTO

Consignar actividades como viajes, visitas, foros, ateneos, prácticas socio-comunitarias y todas otras que se instrumentarán como parte del desarrollo de la asignatura o espacio curricular.

Aquí corresponde mencionar muy especialmente, los proyectos para la mejora de la enseñanza de grado (PIIMEG, PELPA) en los que los docentes de la asignatura participan, y todo proyecto o actividad siempre que signifiquen una contribución al desarrollo de la asignatura y a la formación de los estudiantes.



6. CRONOGRAMA TENTATIVO DE CLASES E INSTANCIAS EVALUATIVAS a realizar en la virtualidad y en la presencialidad

INCORPORA AQUÍ EL TEXTO

Que muestre coherencia y consistencia con el logro de los objetivos y las competencias definidas. Las fechas de parciales deberán ser consensuadas con los responsables de las demás asignaturas del cuatrimestre correspondiente, en acuerdo con la Res. C.S. 120/17).

6.1. Cronograma de clases e instancias evaluativas a realizar en la virtualidad.

Semana	Día/Fecha	Teóricos	Prácticos y Laboratorio (*) COM. 1 (lunes y viernes) Día: Tema	Prácticos y Laboratorio (*) COM. 2 (martes y jueves)	Parciales / Recup.
1	Miércoles 25-08: Cap 1	X	Viernes : cap 1	Jueves : Cap 1	
2	Miércoles 01-08: Cap 2	X	Lunes : cap1 Viernes : cap 2	Martes : cap 1 Jueves : cap 2	
3	Miércoles 08-09: Cap 3.	X	Lunes : cap 2 Viernes : cap3	Martes : cap 2 Jueves : cap 3	
4	Miércoles 15-09: Fin del Cap 3 y algo del 4	X	Lunes : cap 3 Viernes : cap 3	Martes : cap 3 Jueves : cap 3	
5	Miércoles 22-09: Cap 4 y cap 5	X	Lunes 21-09: Cap 4 Viernes : cap 4	Martes : Asueto Jueves : cap 4	
6	Miércoles 29-09: Cap. 5 finalizar		Lunes : cap 5 Viernes : cap 5	Martes : cap 5 Jueves : cap 5	
7	Miércoles: consultas		Lunes : cap 5	Martes : cap 5 Jueves : Primer parcial	Viernes 08-10: feriado con fines turísticos
8	Miércoles 13-10: Practico Cap 6	X	Lunes : 11-10: Feriado Viernes: Cap 6	Martes : Teórico Cap 6 Jueves: Cap 6	Jueves 14 : Recup. 13hs
9	Miércoles 20-10: Cap 7 y 8	X	Lunes: cap 6 Viernes : cap 6 y 7	Martes : cap 6 Jueves : Cap 6 y 7	
10	Miércoles:27/10 Fin del cap 8 e inicio del Cap 9	X	Lunes : cap 8 Viernes : Cap 8	Martes : Cap 8 Jueves : Cap 8	
11	Miércoles 03-11: Cap. 9	X	Lunes : cap 8/9	Martes : cap 8/9	



			Viernes : cap 9	Jueves : cap 9	
12	Miércoles 10-10: Teoría Cap 10		Lunes : cap 9 Viernes : Cap 10	Martes : cap 9 Jueves : día de la ciudad	
13	Miércoles 17-11: Fin Teoría Cap 10		Lunes : cap 10	Martes : : cap 10 Jueves : : cap 10	
14			Lunes 22: Feriado Viernes: Consultas	Martes 23: Seg Parcial Jueves : consultas	Recuperatorio: Martes 30-11

6.2. Cronograma de clases e instancias evaluativas a realizar en la presencialidad.

Semana	Día/Horas	Actividad: tipo y descripción*

*Teóricos, teóricos-prácticos, trabajos de laboratorios, seminarios, talleres, coloquios, instancias evaluativas, consultas grupales y/o individuales, otras.

7. BIBLIOGRAFÍA

7.1. Bibliografía obligatoria y de consulta (por lo menos algún material bibliográfico debe ser de edición 2012 o posterior).

◦ Concepts, Techniques and Models of Computer Programming. Van Roy and Haridi. The MIT Press. ISBN: 0-262-22069-5.

◦ Programming Languages Design and Implementation (Third Edition). Terrence Pratt, Marvin Zelkowitz. Prentice Hall. 1996. ISBN: 0-13-678012-1

◦ Foundations on Logic Programming. J. W. Lloyd. Second, extended edition. Springer Verlag. ISBN: 3-540-18199-7.

◦ Programming Language Concepts. Carlo Guezzi, Medhi Jazayeri. 3rd edition. ISBN:0-471-10426-4.

◦ Apunte de la Materia, elaborado por los docentes de la Asignatura.

7.2. Plataformas/herramientas virtuales; materiales audiovisuales, enlaces, otros.

Plataforma de Educación a Distancia EVELIA: <https://www.evelia.unrc.edu.ar/evelia/>



8. DÍA Y HORARIOS DE CLASES VIRTUALES y PRESENCIALES

Teóricos:

Miércoles de 11 a 14hs.
Modalidad: Virtual

Comisión 1:

Lunes de 9 a 12 hs. Modalidad Virtual
Viernes de 10 a 12 hs. Modalidad Virtual

Comisión 2:

Martes de 16 a 19hs. Modalidad Virtual
Jueves de 14 a 16 hs. Modalidad Virtual

9. DÍA Y HORARIO DE CLASES DE CONSULTAS VIRTUALES y PRESENCIALES

- **Miércoles 19hs**
- **Jueves a las 9hs**
- **Viernes a las 13hs**
- **Miércoles a las 10hs**

10. REQUISITOS PARA OBTENER LA REGULARIDAD Y LA PROMOCIÓN

- **CONDICIONES DE REGULARIDAD:**

Aprobación de los dos exámenes parciales o sus recuperatorios.

- **CONDICIONES DE PROMOCIÓN:**

Promoción: No tiene.

11. CARACTERÍSTICAS, MODALIDAD Y CRITERIOS DE LAS INSTANCIAS EVALUATIVAS

Evaluaciones Parciales:

Dos exámenes parciales con sus respectivos exámenes recuperatorios, los cuales consisten en ejercicios prácticos del tipo a los vistos en la parte práctica.



Universidad Nacional de Río Cuarto
Facultad de Ciencias Exactas, Físico-Químicas y Naturales
HOMENAJE AL PREMIO NOBEL

2021 - "AÑO DE
DE MEDICINA DR. CÉSAR MILSTEIN"

Los exámenes serán por videoconferencias controlados. El examen estará disponible en la Plataforma Evelia el día y hora especificado en el programa para su descarga y los alumnos deberán habilitar sus respectivas cámaras. Tendrá una duración de 2hs aproximadamente. Si algún alumno tiene inconvenientes con la conexión se replanificará otra instancia de evaluación.

Evaluación Final:

Alumnos regulares: evaluación, oral o escrita sobre conceptos teóricos.

Alumnos libres: en una primera instancia deberán aprobar ejercicios prácticos, y luego se procede a una evaluación similar para los alumnos regulares.

En ambos casos la modalidad será virtual por videoconferencias.

Firma Profesor/a Responsable

Firma Secretario/a Académico/a