

EX-2026-00088647- -UNC-ME#FAMAF

PROGRAMA DE ASIGNATURA	
ASIGNATURA: Redes y Sistemas Distribuidos	AÑO: 2026
CARÁCTER: Obligatoria	UBICACIÓN EN LA CARRERA: 3° año 1° cuatrimestre
CARRERA: Licenciatura en Ciencias de la Computación	
RÉGIMEN: Cuatrimestral	CARGA HORARIA: 120 horas

FUNDAMENTOS Y OBJETIVOS

Las redes de computadoras y las aplicaciones basadas en redes de computadoras son fundamentales para el trabajo profesional y son recursos valiosos para quienes hacen investigación y docencia. Para la formación del/de la estudiante no solo se espera que sepan usar las redes de computadoras y las aplicaciones basadas en ellas, sino también comprender cómo se arman las redes, cuáles son sus componentes y los protocolos de software para las mismas; esto les ayudará a eventualmente poder construir y administrar redes de computadoras. Los/as estudiantes aprenderán los fundamentos sobre los sistemas operativos de redes; esto les podría servir en el futuro para participar en el desarrollo de protocolos de redes o de partes de sistemas operativos de redes. En el mundo moderno hay distintos paradigmas de desarrollo de software sobre redes: cliente-servidor, peer to peer, middlewares, etc. Los/as estudiantes adquirirán las primeras experiencias de desarrollo de aplicaciones de redes basándose en algunos de dichos paradigmas y en algunos protocolos de redes. En la materia seguimos el enfoque de organizar los sistemas operativos de redes como una arquitectura de capas donde cada capa tiene sus protocolos y se abstrae de ciertos problemas; esta forma de dar la materia ayuda a organizarla y a que los alumnos la comprendan (la capa de más abajo tiene que ver con el hardware de las redes y las dos capas de más arriba son necesarias para aprender a construir aplicaciones de redes). En cada capa hacemos énfasis en conceptos fundamentales, en cómo resolver los problemas asociados a ella, y en comprender y evaluar los protocolos más importantes usados hoy en día.

Objetivos:

Los/as estudiantes deberán alcanzar los siguientes:

Conocer el hardware de las redes y entender los límites teóricos de velocidad de transferencia.

Comprender los conceptos y problemas a resolver para las distintas capas de sistemas operativos de redes (SOR) arriba del hardware de las redes. Poder hacer razonamientos acerca de protocolos de red (mediante cálculos - usando recursos del álgebra, la aritmética, el análisis matemático, y la probabilidad y estadística – el uso de los conceptos en los que se basan los protocolos, y el empleo de las reglas de los protocolos).

Poder llevar a cabo evaluaciones de cómo se comporta un protocolo de acuerdo a las propiedades que importan para el mismo.

Poder evaluar la cantidad de los recursos que un protocolo de red consume y así como explicar bajo qué circunstancias un protocolo se comporta bien y en cuáles casos se comporta mal.

Poder comparar las alternativas de protocolos para una cierta capa de SOR entre sí desde distintos puntos de vista.

Poder programar aplicaciones distribuidas que usan APIs de comunicación de redes: aquí nuevamente los/as estudiantes deberán conocer los protocolos intervinientes y tener en cuenta las reglas por ellos definidas.

CONTENIDO

1. Introducción

Redes de computadoras. Servicios proporcionados por las redes de computadoras. Tipos de redes. Internet de las cosas. Sistemas operativos de red. Jerarquías de protocolos. Modelos de referencia. Protocolos de internet de las cosas. Cómputo en la nube.

2. La Capa de Aplicación

Enfoques para desarrollar aplicaciones de red. Estilos de arquitectura de aplicaciones de red: cliente-servidor y peer-peer. Protocolos de capa de aplicación. La web: panorama de la arquitectura, navegadores web, plug-ins y aplicaciones de ayuda, servidores web, protocolo de transferencia de hipertexto (HTTP), documentos web estáticos (HTML), páginas dinámicas, generación de páginas web del lado del servidor usando PHP, Cookies, manejo de cookies con PHP.

3. La Capa de Transporte Primitivas y sockets

Elementos de los protocolos de transporte. Conceptos básicos de TCP, problemas elementales sobre envío y recepción de mensajes en TCP, encabezado de TCP. Direccionamiento. Direccionamiento en TCP. Protocolos para transferencia de datos confiable: parada y espera, retroceso N, y repetición selectiva. Control de flujo, protocolos de control de flujo, control de flujo en TCP. Control de congestión. Control de congestión en TCP: distintos protocolos. Establecimiento y fin de conexiones. Establecimiento y liberación de conexiones en TCP. Administración de temporizadores en TCP. Protocolo UDP.

4. La Capa de Red

Aspectos de diseño de la capa de red. Conmutación de paquetes de almacenamiento y reenvío. Servicios proporcionados a la capa de transporte. Servicio no orientado a la conexión. Servicio orientado a la conexión. Algoritmos de enrutamiento: principio de optimización, enrutamiento de ruta más corta, inundación, enrutamiento de vector de distancia, enrutamiento por estado del enlace, enrutamiento jerárquico. Control de congestión: principios generales del control de congestión, políticas de prevención de congestión, control de congestión en subredes de datagramas, desprendimiento de carga. Interconectividad: cómo difieren las redes, conectando redes, fragmentación. Capa de red de Internet: protocolo IP, formatos de direcciones IP, subredes, CIDR, traducción de dirección de red (NAT), Ipv6. Protocolo OSPF (abrir primero la ruta más corta). Protocolos de puerta de enlace exterior, BGP.

5. La Capa de Enlace de Datos

Funciones de la capa de enlace de datos. Tramas. Servicios provistos a la capa de red. El problema de la asignación del canal. Protocolos de acceso múltiple sin detección de portadora, protocolos de acceso múltiple con detección de portadora. Ethernet: cableado Ethernet, formato de trama, cálculo de tamaño de trama mínima, algoritmo de retroceso exponencial binario, Ethernet conmutada. Fast Ethernet.

Gigabit Ethernet. Redes inalámbricas: tipos de redes inalámbricas, problemas de las redes inalámbricas, protocolo CSMA/CA; protocolo 802.11 PCF.

6. La Capa Física

Bases teóricas de comunicación de datos. Análisis de Fourier. Resultados de Niquist y Shannon. Conversiones entre señales digitales y analógicas. Módems. Medios de transmisión guiados y no guiados. Multiplexión. Sistema telefónico público conmutado. DSL. Sistema telefónico móvil. Internet por cable. Fibra a la casa.

BIBLIOGRAFÍA

BIBLIOGRAFÍA BÁSICA

- Andrew S. Tanenbaum and David J. Wetherhall. Computer Networks (6th Edition). Prentice Hall, 2021.
- Kurose, J. F. and Ross, K. W. Computer Networking – A Top Down Approach. Seventh Edition, Pearson, 2017.

BIBLIOGRAFÍA COMPLEMENTARIA

- Douglas E. Comer. Computer Networks and Internets. 5th edition, Prentice Hall, 2009.
- William Stallings. Data and Computer Communications. 8th edition, Prentice Hall, 2007.
- Larry L. Peterson and Bruce S. Davie. Computer Networks. 5th edition, Morgan Kaufmann, 2011.

METODOLOGÍA DE ENSEÑANZA

Formatos empleados para el desarrollo de las clases

Se usan clases teórico-prácticas híbridas (i.e. con asistencia presencial y remota), donde se estudia la teoría de la materia y se desarrollan actividades prácticas bajo supervisión de los/as docentes.

El laboratorio de la materia combina explicaciones conceptuales con instancias de desarrollo de código y de experimentación guiada. Las clases se desarrollan tanto en modalidad presencial física como remota, con instancias sincrónicas para acompañamiento técnico y conceptual. Se pone a disposición además, distintos material (videos, guías y enunciados) a través del Aula Virtual. Cada laboratorio incluye demostraciones, sesiones de preguntas y ejercicios en vivo para afianzar los conceptos antes de la entrega.

Secuenciación de los contenidos

Para el teórico-práctico se comienza con la introducción de la materia; luego se usa enfoque top-down recorriendo las diferentes capas de red desde las más abstractas a las más concretas; estas son: capa de aplicación, capa de transporte, capa de red, capa de enlace de datos y capa física.

Los contenidos del laboratorio se organizan de manera incremental siguiendo el enfoque top-down de la materia. Se inicia con el desarrollo de aplicaciones de red en la capa de aplicación (Lab 1 y Lab 2), centradas en programación con sockets en

Python y protocolos estandarizados o diseñados ad hoc. Luego, se avanza hacia la experimentación en capas inferiores mediante simulaciones con OMNeT++ (Lab 3, Lab 4 y Lab 5), donde se analizan fenómenos de transporte, enrutamiento y acceso al medio. Esta secuencia permite que el/la estudiante transite desde la construcción de aplicaciones distribuidas hasta la modelización y evaluación de mecanismos internos de una red real, tal como se detalla en los enunciados de los laboratorios.

Tareas y actividades que realizan los estudiantes

Para clases teórico-prácticas: i) Derivar protocolos para las diferentes capas. ii) Comparar protocolos que cumplen la misma función dentro de una capa. iii) Hacer cálculos de uso de recursos que involucran los protocolos (latencia, utilización de canal, ventanas, espacios de secuencia, temporizadores, representación interna de la red entre otros). iv) Simular secuencias de mensajes para protocolos bajo ciertas condiciones. v) Construcción de mensajes para ciertas necesidades. vi) Diseñar redes de distintos tipos para diferentes necesidades.

Para el laboratorio de la materia las actividades principales consisten en: (i) implementación de aplicaciones cliente/servidor con sockets (Labs 1 y 2), (ii) desarrollo y extensión de modelos en OMNeT++ para estudiar control de flujo y congestión (Lab 3), enrutamiento estático y dinámico (Lab 4), y protocolos de enlace como ALOHA (Lab 5), (iii) análisis cuantitativo de resultados mediante métricas obtenidas de simulaciones (.sca, .vec), (iv) elaboración de informes técnicos y presentaciones orales grupales de hasta 10 minutos, según lo solicitado en los enunciados. Además, los/las estudiantes realizan lecturas guiadas y resuelven ejercicios de modelización, depuración de código y discusión conceptual sobre el comportamiento observado en sus implementaciones.

Materiales y recursos didácticos usados para la enseñanza y aprendizaje de la materia

Para el teórico práctico: usamos un Aula Virtual en Moodle que contiene: presentaciones audiovisuales de los temas de la materia, videos de clases grabadas, parciales de ejemplo y su resolución, listas de ejercicios y problemas de los capítulos de la materia, ejercicios y problemas resueltos

El laboratorio utiliza: Python 3 para programación con sockets (Labs 1 y 2), OMNeT++ para simulación en capas inferiores (Labs 3 a 5), repositorios Git provistos por la Facultad para entregas, presentaciones multimedia, videos tutoriales y ejemplos de código. El Aula Virtual contiene cuestionarios, foros de consulta, cronogramas, enunciados y material complementario. También se emplean herramientas de línea de comando, Wireshark, contenedores Docker en talleres prácticos y APIs específicas presentadas por la cátedra.

Asimismo, en el entorno Moodle, los/as estudiantes disponen de un foro de consultas a donde pueden hacer preguntas de los temas desarrollados en las clases teórico-prácticas, así como en los laboratorios.

Dinámicas de trabajo propuestas a los estudiantes

Para el teórico-práctico: En las clases se intercalan presentación de contenido teórico con actividades de ejercitación. Estas actividades son supervisadas por el/la docente. Algunas clases son solo de ejercitación donde hay actividades de resolución de guías de ejercicios y problemas. Antes de los parciales siempre hay una clase donde se resuelve un parcial modelo y se atienden consultas sobre los



Universidad
Nacional
de Córdoba



Facultad de Matemática,
Astronomía, Física y
Computación

contenidos a evaluar. Los/as estudiantes pueden usar el canal de Zulip de la materia para hacer consultas sobre la materia o plantear inquietudes.

Para la parte de laboratorio las actividades se realizan en grupos de tres estudiantes, promoviendo la interacción entre pares mediante el diseño colaborativo de código, la discusión de decisiones de ingeniería y el co-análisis de resultados. Las consultas se canalizan a través de foros (Moodle y Zulip) y de sesiones sincrónicas con docentes y ayudantes. En las evaluaciones orales, la interacción es individual, lo que permite verificar la comprensión conceptual y la capacidad de justificar las decisiones de implementación. La dinámica fomenta el aprendizaje activo y el razonamiento crítico sobre el funcionamiento real de las redes de computadoras.

EVALUACIÓN

FORMAS DE EVALUACIÓN

REGULARIDAD

- Aprobación de los 2 parciales, o de 1 parcial y de 1 recuperatorio.
- Aprobar al menos el 60% de los trabajos de laboratorio.

PROMOCIÓN

- Aprobar todas las evaluaciones parciales con una nota no menor a 6 (seis), y obteniendo un promedio no menor a 7.
- Entrega y aprobación de todos los trabajos de laboratorio en las fechas establecidas con nota no menor a 6.