



UNIVERSIDAD DEL CAUCA

FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

PROGRAMA DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

ASIGNATURA : INFORMÁTICA PARA TELECOMUNICACIONES
CRÉDITOS : 3
MODALIDAD : TEÓRICA/PRÁCTICA
INTENSIDAD : 4 HORAS SEMANALES
PRERREQUISITOS: ALGORITMOS COMPUTACIONALES
ÁREA : CIENCIAS BÁSICAS DE INGENIERÍA
DEPARTAMENTO : TELEMÁTICA

OBJETIVO

Proporcionar a los estudiantes competencias para resolver eficientemente problemas comunes en el dominio de las telecomunicaciones.

Objetivos Específicos

El estudiante al final del curso estará en capacidad de:

- Seleccionar, adaptar y utilizar algoritmos típicos para resolver problemas comunes en el dominio de las telecomunicaciones.
- Emplear principios y técnicas básicas de complejidad computacional para analizar y evaluar algoritmos comúnmente utilizados en el dominio de las telecomunicaciones.
- Utilizar programación orientada a objetos para implementar algoritmos comúnmente utilizados en el dominio de las telecomunicaciones.

METODOLOGÍA

La asignatura se desarrollará mediante:

Clases orientadas por el profesor en las que se resolverán dudas acerca de los contenidos del curso; tales contenidos deben ser previamente enviados por el profesor y desarrollados por los estudiantes.

- Prácticas guiadas por el profesor.
- Prácticas desarrolladas por los estudiantes
- Proyecto final.
- Consultas soportadas por el profesor y monitores de pregrado/posgrado.

CONTENIDO

1. Introducción
 - 1.1. Algoritmos de enrutamiento
 - 1.2. Algoritmos para multicast
 - 1.3. Algoritmos para gestión de tráfico
2. Algoritmos de enrutamiento no adaptativo
 - 2.1. Sequence Numbers
 - 2.2. Hop Count
 - 2.3. Spanning Tree
 - 2.4. Random Walk
3. Algoritmos de enrutamiento adaptativo
 - 3.1. Hot Potato
 - 3.2. Backward Learning

4. Algoritmos para enrutamiento de camino más corto
 - 4.1. Bellman-Ford
 - 4.2. Dijkstra's
 - 4.3. Floyd Warshall
 - 4.4. Link State Routing
5. Algoritmos para multicast
 - 5.1. Shortest Path Tree
 - 5.2. Reverse Path Forwarding
 - 5.3. Shared Tree
 - 5.4. Center-based Tree
6. Algoritmos para gestión de tráfico
 - 6.1. Round Robin
 - 6.2. Weighted RR
 - 6.3. Deficit RR
 - 6.4. First in First Out
 - 6.5. Fair Queuing
 - 6.6. Weighted FQ
 - 6.7. Adaptive Virtual Queue
 - 6.8. Class-based Queueing
 - 6.9. Credit-based Fair Queueing

EVALUACIÓN

El tipo de evaluación y la respectiva ponderación son concertadas el primer día de clase con los estudiantes, teniendo en cuenta el reglamento estudiantil de la Universidad del Cauca. El sistema de evaluación promueve la eficiencia y calidad del proceso de enseñanza - aprendizaje del curso, detectando el nivel de desempeño de los estudiantes con el fin de realizar los correctivos necesarios durante el transcurso del semestre.

BIBLIOGRAFÍA

1. James F. Kurose. Keith W. Ross. Computer Networking: A Top-Down Approach (6th Edition) 6th Edition. <http://www-net.cs.umass.edu/kurose-ross-ppt-6e/>.
2. Mark Lutz. Learning Python. 2013. ISBN: 978-1-449-35573-9.
3. Rudolf Pecinovský. Learn Object Oriented Thinking and Programming. ISBN 978-80-904661-9-7 (PDF). 2013.
4. Sara Baase, Allen Van Gelder: Computer Algorithms - Introduction to Design and Analysis (3. ed., repr. with corr.). Pearson / Prentice Hall 2000, ISBN 978-0-201-61244-8, pp. I-XIX, 1-688.
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein:
6. Introduction to Algorithms (3. ed.). MIT Press 2009, ISBN 978-0-262-03384-8, pp. I-XIX, 1-1292.
- 7.
8. L., Joyanes. Programación Orientada a Objetos. Segunda Edición. McGraw-Hill. 1998.
9. R., Pressman. Ingeniería del Software, Un Enfoque Práctico. McGraw-Hill. 1998.
10. C., Larman, UML y patrones (México, NJ: Prentice-Hall, 1999).
11. R., Coss, simulación un enfoque práctico (México, NJ: Prentice-Hall, 1999).
12. M., Flower, UML gota a gota (México, NJ: Prentice-Hall, 1997).
13. I., Jacobson, G., Booch, J., Rumbaugh. El Proceso Unificado de Desarrollo de Software. Addison Wesley.
14. I., Jacobson, G., Booch, J., Rumbaugh. El Lenguaje de Modelado Unificado. Addison Wesley.
15. Laura Lemay y Rogers Cadenhead. Aprendiendo Java2 en 21 Días.
16. Java 2 Curso de programación. 2ª. Edición. Francisco Javier Ceballos. Alfaomega-Rama. 2003.
17. Weitzenfeld, Alfredo. Ingeniería de software orientada a objetos con UML, java e internet. Thomson, 2005.