

Curso Académico: ( 2021 / 2022 )

Fecha de revisión: 04-06-2021

Departamento asignado a la asignatura: Departamento de Informática

Coordinador/a: EXPOSITO SINGH, DAVID

Tipo: Optativa Créditos ECTS : 6.0

Curso : 4 Cuatrimestre :

**REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)**

Sistemas Operativos  
Diseño de Sistemas Operativos  
Arquitectura de Computadores

**OBJETIVOS**

El objetivo de este curso es presentar a los alumnos distintos tópicos relacionados con el dominio de desarrollo de software de sistemas. Estos son, el lenguaje de programación C++, los servicios existentes en el sistema operativo Linux/UNIX y distintas herramientas de desarrollo, monitorización y análisis de rendimiento del software de sistemas. El objetivo de la asignatura es que el alumno desarrolle software que explote de manera eficiente. Para lograr este objetivo el alumno debe adquirir una serie de competencias genéricas, conocimientos, capacidades y actitudes.

**1. Competencias Genéricas:**

- Capacidad de análisis y síntesis (PO a, c, e)
- Capacidad de organización y planificación (PO c, d)
- Resolución de problemas (PO c, g)
- Trabajo en equipo (PO d)
- Capacidad de aplicar los conceptos teóricos (PO a, c)

**2. Competencias Específicas:**

- a. Cognitivas (PO a, c, e)
- Capacidad para diseñar software de sistemas
  - Capacidad para implementar software de sistemas
  - Conocimiento de técnicas de entrada/salida
  - Conocimiento de programación concurrente
  - Empleo de servicios ofrecidos por los sistemas operativos
- b. Procedimentales/Instrumentales (PO a, c, g, k)
- Resolver problemas de concurrencia.
  - Desarrollar software de bajo nivel para sistemas.
- c. Actitudinales (PO: c, e)
- Creatividad
  - Visión crítica del desarrollo de software de sistemas
  - Proactividad
  - Interés por adquirir nuevos conocimientos e información

**3.- Competencias Generales y Transversales**

- Utilizar de forma eficiente medios TIC para redactar informes técnicos y memorias de proyectos y trabajos sobre Informática, así como presentaciones de calidad (CG9)
- Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería (CGB4)

**4.- Competencias Comunes a la Rama de Informática**

- Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios (CECR110)

**5.- Competencias Específicas para la Mención de Ingeniería de Computadores**

- Capacidad de diseñar e implementar software de sistema y de comunicaciones (CEIC4)

Enlace al documento

## DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

Asignatura no impartida este curso académico

- T1. Introducción a la programación de sistemas
  - 1.1. Qué es el software de sistemas
  - 1.2. Historia de los lenguajes de programación
  
- T2. Lenguajes de programación de sistemas: C++
  - 2.1 Objetos, tipos y valores
  - 2.2 Clases y herencia. Sobrecarga de operadores
  - 2.3 Contenedores, arrays y Free store
  - 2.4 Operaciones de copia y movimiento
  - 2.5 Plantillas, excepciones y gestión de recursos
  - 2.6 Metaprogramación
  - 2.7 STL, objetos función y expresiones Lambda
  - 2.8 C++11 y C++14
  
- T3. Concurrencia en C++
  - 3.1 Concurrencia basada en hilos
  - 3.2 Concurrencia basada en tareas
  
- T4. Gestión y optimización de la memoria
  - 4.1 Control de errores y depuración
  - 4.2 Memoria compartida
  - 4.3 Mapeos de memoria
  
- T5. Entrada/salida
  - 5.1 E/S vectorizada
  - 5.2 E/S asíncrona
  - 5.3 Buffering de E/S
  
- T6. Señales y temporizadores
  - 6.1 Multiplexación de E/S
  - 6.2 E/S basada en señales
  - 6.3 Funciones no reentrantes y goto no local
  
- T7. Gestión de bibliotecas y utilidades
  - 7.1 Bibliotecas estáticas y compartidas
  - 7.2 Bibliotecas dinámicas
  
- T8. Monitorización y análisis de prestaciones
  - 8.1 Métricas de rendimiento
  - 8.2 Herramientas de evaluación de rendimiento
  - 8.3 Herramientas de monitorización de Linux

## ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

- 1. Clases magistrales de teoría 1 ECTS (PO a, c, e)
  - Presentación de los conceptos teóricos de la asignatura
  - Para facilitar su desarrollo los alumnos tendrán textos básicos de referencia de teoría y de problemas que les permitirán completar y profundizar en aquellos temas en los cuales estén más interesados.
- 2. Proyectos 1.5 ECTS (PO a, c, d, e, g, k)
  - Se desarrollarán varios proyectos durante el curso. Parte de los mismos se harán en laboratorios con apoyo del profesor.
  - Los estudiantes tienen que analizar los requisitos y ofrecer una solución correcta
  - Los proyectos se harán en grupos de 2 estudiantes para promover el trabajo en equipo
- 3. Resolución de ejercicios por parte del profesor 1 ECTS (PO a, c, e)
  - Solución de ejercicios en clase con participación de los alumnos
- 4. Trabajo del alumno 2 ECTS. (PO a, c, e, k)
  - Estudio para comprender los conceptos teóricos
  - Resolución de los ejercicios propuestos
- 5. Exámenes 0.5 ECTS (PO a, c, e, g)
  - Examen final

## SISTEMA DE EVALUACIÓN

El 40% de la calificación final se obtendrá mediante un examen final de evaluación de los conocimientos adquiridos. El 60% restante será el resultado de un proceso de evaluación continua.

El proceso de evaluación continua incluye:

- Prácticas con un peso del 30% (PO: a, c, d, e, g, k).
- Ejercicios y trabajos de laboratorio, con un peso del 30% (PO: a, c, d, e, g, k).

Se considera que se ha seguido la evaluación continua si se han entregado todas las prácticas y se han realizado, al menos, las dos terceras partes de los ejercicios y trabajos de laboratorio propuestos.

Para aprobar las prácticas será necesario obtener una nota mínima de 2 en cada una. El examen final de evaluación tiene un peso del 40%. Para aprobar el examen será necesario obtener una nota mínima de 4.0

En caso de detección de copia, los alumnos implicados perderán la evaluación continua. La copia se entiende entre compañeros o por copiar las prácticas desde portales de Internet.

Se perderá la evaluación continua si no se entregan todas las prácticas, no se alcanza el mínimo en prácticas o no se realizan los ejercicios y trabajos de laboratorio.

En caso de que no se alcance el mínimo pero la ponderación de la evaluación continua y el examen ordinario de aprobado, el alumno estará suspenso con una nota de 4,5.

Para el cálculo de la calificación final en la convocatoria extraordinaria se contemplarán los siguientes casos:

A-Estudiantes que hayan seguido el proceso de evaluación continuada:

- a-La calificación del examen extraordinario tendrá un peso del 40%.
- b-El 60% restante se obtendrá de la calificación de evaluación continua.
- c-Será necesario sacar una nota mínima de 4 puntos para hacer media con la evaluación continua.

B-Estudiantes que no hayan seguido de forma completa el proceso de evaluación continua:

El examen extraordinario tendrá un peso del 100% y podrá incluir una parte de prácticas para comprobar que se tienen estos conocimientos. El examen ordinario representará el 60% de la calificación para todos aquellos alumnos que decidan no integrarse en el sistema anterior de evaluación continua.

**Peso porcentual del Examen Final:** 40

**Peso porcentual del resto de la evaluación:** 60

## BIBLIOGRAFÍA BÁSICA

- A. Hoover System Programming with C and Unix, Pearson, 2009
- Bjarne Stroustrup Programming: Principles and Practice using C++, Addison-Wesley, 2009
- Michael Kerrisk The Linux Programming Interface. A Linux and UNIX System Programming Handbook, William Pollock, 2010

## BIBLIOGRAFÍA COMPLEMENTARIA

- Bjarne Stroustrup The C++ Programming Language, Addison-Wesley, 2013
- G. R. Andrews Foundations of Multithreaded, Parallel, and Distributed Programming, Addison-Wesley, 1999
- L. L. Beck System Software: An Introduction to Systems Programming, Pearson , 1996
- M. Barr Programming Embedded Systems in C and C++, O'Reilly Media, 1999
- Nicolai M. Josuttis The C++ Standard Library, Addison-Wesley, 2012

## RECURSOS ELECTRÓNICOS BÁSICOS

- Tenouk . The Tenouk's C, C++: <http://www.tenouk.com/>