



UNIVERSIDAD DE LAS PALMAS
DE GRAN CANARIA

GUÍA DOCENTE

CURSO: 2011/12

14101 - ARQUITECTURA DE COMPUTADORES

ASIGNATURA: 14101 - ARQUITECTURA DE COMPUTADORES

CENTRO: Escuela de Ingeniería de Telecomunicación y Electrónica

TITULACIÓN: Ingeniero de Telecomunicación

DEPARTAMENTO: INGENIERÍA TELEMÁTICA

ÁREA: Ingeniería Telemática

PLAN: 13 - Año 2000**ESPECIALIDAD:**

CURSO: Cuarto curso **IMPARTIDA:** Primer semestre **TIPO:** Troncal

CRÉDITOS: 9

TEÓRICOS: 6

PRÁCTICOS: 3

Información ECTS

Créditos ECTS: 7,2

Horas de trabajo del alumno: 180

Horas presenciales:

- Horas teóricas (HT): 36
- Horas prácticas (HP): 30
- Horas de clases tutorizadas (HCT): 3
- Horas de evaluación: 5
- otras:

Horas no presenciales:

- trabajos tutorizados (HTT): 40
- actividad independiente (HAI): 66

Idioma en que se imparte: Español

Descriptores B.O.E.

Estructuras en niveles. Máquinas virtuales. Sistemas operativos. Núcleos en tiempo real.

Temario

Tema 1. Introducción a la arquitectura de computadores (2h)

1.1 Sistema computador y sistemas operativos

1.1.1 Sistemas operativos: procesos y ficheros

1.1.2 Comunicación y sincronización de procesos

1.1.3 Planificación de la CPU

1.1.4 Gestión de la memoria

1.2 Arquitecturas paralelas

1.2.1 Multiprocesadores

1.2.2 Multicomputadores

1.3 Núcleos en tiempo real

1.3.1 Planificación para tiempo real

1.3.2 Comunicación y sincronización

1.3.3 Manejo de memoria para tiempo real

1.4 Máquinas virtuales

1.4.1 Introducción a los lenguajes

1.4.2 Análisis léxico

1.4.3 Análisis sintáctico

1.4.4 Generación de código

Tema 2. Sistemas operativos: procesos y ficheros (6h)

2.1 Conceptos básicos (2)

2.1.1 Procesos y threads

2.1.2 Ficheros

2.2 Manejo de procesos y threads (2)

2.2.1 fork()

2.2.2 exec()

2.2.3 wait()

2.2.4 exit()

2.3 Manejo de ficheros (2)

2.3.1 open()

2.3.2 read()

2.3.3 write()

2.3.4 close()

Tema 3. Comunicación y sincronización entre procesos (8h)

3.1 Comunicación mediante pipes (2h)

3.1.1 Pipes bloqueantes

3.1.2 Implementación en Linux: pipe()

3.2 Memoria compartida (2h)

3.2.1 shmget()

3.2.2 shmat()

3.3 Sincronización (2h)

3.3.1 Memoria compartida

3.3.2 Pipes

3.4 Exclusión mútua: semáforos (2h)

3.4.1 Regiones críticas

3.4.2 semget()

3.4.3 wait(), signal()

Tema 4. Arquitecturas paralelas (4h)

4.1 Multiprocesadores (1h)

4.1.1 Arquitectura básica

4.1.2 Modelo de computación

4.2 Multicomputadores (3h)

4.2.1 Arquitectura básica

- 4.2.2 Modelo de computación
- 4.2.3 Message Passing Interface (MPI)

Tema 5. Máquinas virtuales (14)

- 5.1 Introducción (1)
 - 5.1.1 Análisis léxico
 - 5.1.2 Análisis sintáctico
 - 5.1.3 Generación de código
- 5.2 Análisis léxico (3)
 - 5.2.1 Expresiones regulares
 - 5.2.2 Herramienta Flex
- 5.3 Análisis sintáctico (6)
 - 5.3.1 Análisis descendente
 - 5.3.2 Análisis ascendente
 - 5.3.3 Herramienta Bison
- 5.4 Generación de código (4)
 - 5.4.1 Sentencias de asignación
 - 5.4.2 Sentencias condicionales
 - 5.4.3 Sentencias repetitivas

Tema 6. Planificación de la CPU (10h)

- 6.1 Introducción (1)
 - 6.1.1 Planificación para monoprocesadores
 - 6.1.2 Planificación para multiprocesadores
- 6.2 Planificación para monoprocesadores (6h)
 - 6.2.1 Estados de los procesos
 - 6.2.2 First-Come First-Served (FCFS)
 - 6.2.3 Round-Robin (RR)
 - 6.2.4 Shorter Process Next (SPN)
 - 6.2.5 Shorter Remaining Time (SRT)
 - 6.2.6 Highest Response Ratio Next (HRRN)
- 6.3 Planificación para multiprocesadores (3h)
 - 6.3.1 Load Sharing
 - 6.3.2 Gang scheduling
 - 6.3.3 Asignación dedicada

Tema 7. Gestión de la memoria (10h)

- 7.1 Esquemas básicos (4h)
 - 7.1.1 Reubicación: estática y dinámica
 - 7.1.2 Particiones fijas
 - 7.1.3 Particiones dinámicas
 - 7.1.4 Paginación simple
 - 7.1.5 Segmentación simple

- 7.2 Memoria virtual (6h)
 - 7.2.1 Principios básicos
 - 7.2.2 Memoria virtual paginada
 - 7.2.3 Memoria virtual segmentada

Tema 8. Núcleos en tiempo real (6h)

- 8.1 Introducción (1h)
 - 8.1.1 Planificación para tiempo real
 - 8.1.2 Comunicación y sincronización
 - 8.1.3 Manejo de memoria para tiempo real
- 8.2 Planificación para tiempo real (2h)
 - 8.2.1 Earliest Deadline Scheduling (EDS)
 - 8.2.2 Rate Monotonic Scheduling (RMS)
- 8.3 Comunicación y sincronización (2h)
 - 8.3.1 Memoria compartida
 - 8.3.2 Regiones críticas
 - 8.3.3 Semáforos
 - 8.3.4 Señales
- 8.4 Manejo de memoria para tiempo real (1h)
 - 8.4.1 Particiones fijas
 - 8.4.2 Particiones dinámicas
 - 8.4.3 Paginación simple
 - 8.4.4 Segmentación simple
 - 8.4.5 Memoria virtual

Requisitos Previos

Se recomienda que el alumno haya cursado previamente todas las asignaturas anteriores del plan de estudios de la titulación de Ingeniero de Telecomunicación, en particular aquellas relacionadas con la programación.

Objetivos

Objetivos conceptuales (saber):

- 1.1) Comprender los principios básicos de los sistemas operativos.
- 1.2) Conocer los principio básicos de las arquitecturas paralelas.
- 1.3) Comprender los principios básicos de los núcleos en tiempo real.
- 1.4) Conocer los principios básicos de las máquinas virtuales.

Objetivos procedimentales (saber hacer):

- 2.1) Utilizar el lenguaje de programación C.
- 2.2) Utilizar el sistema operativo Linux.

2.3) Programar en Linux utilizando el sistema de archivos.

2.4) Programar en Linux la comunicación y sincronización de procesos.

2.5) Programar con memoria compartida y semáforos en Linux.

2.6) Programar multicomputadores utilizando MPI (Message Passing Interface).

2.7) Utilizar las herramientas Flex y Bison para la construcción de máquinas virtuales.

Objetivos actitudinales (saber ser o estar):

3.1) Elaborar y comunicar de forma oral y escrita un trabajo propio realizado en un grupo de 2 alumnos.

Metodología

La metodología de enseñanza de esta asignatura se fundamenta en los siguientes 4 tipos de clases:

a)Clases de teoría

Actividad del profesor: clases expositivas combinadas con ejercicios.

Actividad del alumno:

Actividad presencial: tomar notas sobre la documentación de la asignatura (disponible en el campus virtual) y plantear dudas.

Actividad no presencial: estudiar la materia.

b)Trabajo de curso en grupo (2 estudiantes por grupo)

Actividad del profesor: tutelar el desarrollo de los trabajos.

Actividad del alumno:

Actividad presencial: asistencia a horas de tutorías y exposición pública del trabajo realizado.

Actividad no presencial: búsqueda y elaboración de la información para desarrollar el trabajo, coordinación con el otro miembro del grupo, y elaboración de la memoria escrita del trabajo.

c)Prácticas de laboratorio

Actividad del profesor: tutelar en el laboratorio el desarrollo de las prácticas.

Actividad del alumno:

Actividad presencial: Realización de las prácticas del proyecto docente, así como plantear dudas sobre la práctica que se está realizando.

Actividad no presencial: Preparación y estudio previo de las prácticas.

d)Tutorías

Actividad del profesor: tutelar cualquier tipo de duda que plantea el alumno.

Actividad del alumno:

Actividad presencial: acudir al despacho del profesor en horario de tutorías.

Actividad no presencial: preparación previa de las dudas antes de acudir al despacho del profesor.

Criterios de Evaluación

Actividades que liberan materia:

La nota correspondiente al examen de la parte práctica.

Actividades que no liberan materia pero puntúan:

a) Aquellos alumnos que asistan regularmente a las clases de laboratorio (se permite un máximo de 2 faltas sin justificar) tendrán una bonificación valorada en 0.5 puntos. Esta bonificación se mantiene hasta la convocatoria extraordinaria.

b) Trabajo de curso en grupo (2 alumnos/grupo): la puntuación del trabajo puede suponer hasta un máximo de 2 puntos (equivale a un 20% de evaluación continua). Esta nota se mantiene hasta la convocatoria extraordinaria.

c) Otras consideraciones:

La evaluación restante de la asignatura se realiza mediante el examen final (por escrito) correspondiente a cada convocatoria. Este examen consta de 2 partes: a) examen correspondiente a la parte teórica (0..6 puntos), b) examen correspondiente a la parte práctica (0..2 puntos). La nota final de la asignatura se obtiene sumando las notas parciales:

$NOTA_FINAL = \text{nota de teoría} + \text{nota de prácticas} + \text{nota del trabajo} + \text{bonificación de laboratorio}$

Descripción de las Prácticas

Las prácticas se desarrollan en el Laboratorio de Arquitecturas del Departamento de Ingeniería Telemática.

Práctica Nº 1 (6 horas)

El alumno toma contacto con el entorno Linux y con unos mínimos de programación C. Para ello desarrolla una serie de ejercicios propuestos por el profesor.

Práctica Nº 2 (2 horas)

El alumno utiliza llamadas al sistema para el manejo de archivos para resolver una serie de ejercicios propuestos por el profesor.

Práctica Nº 3 (4 horas)

El alumno maneja llamadas al sistema para la creación, comunicación y sincronización de

procesos para resolver una serie de ejercicios propuestos por el profesor.

Práctica N° 4 (2 horas)

El alumno maneja llamadas al sistema para el uso de memoria compartida para resolver una serie de ejercicios propuestos por el profesor.

Práctica N° 5 (4 horas)

El alumno maneja llamadas al sistema para el uso semáforos.

Práctica N° 6 (4 horas)

El alumno desarrolla varios programas para multicomputadores (se utilizan las diferentes máquinas del laboratorio) utilizando la librería MPI (Message Passing Interface).

Práctica N° 7 (8 horas)

El alumno diseña e implementa una pequeña máquina virtual utilizando las herramientas Flex y Bison.

Bibliografía

[1 Básico] Operating system concepts /

Abraham Silverschatz, Peter B. Galvin.

Addison-Wesley,, Reading (Massachusetts) : (1998) - (5th ed.)

0201542625

[2 Básico] Compilers: principles, techniques and tools /

Alfred V. Aho ...[et al.].

Addison Wesley,, Boston : (2007) - (2nd ed.)

0321491696 (ed. int.)

[3 Básico] Compiladores: principios, técnicas y herramientas /

Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman.

Addison-Wesley Iberoamericana,, México : (1998)

[4 Básico] Real-time systems design and analysis: an engineer's handbook /

Phillip Laplante.

Institute of Electrical and Electronics Engineers,, New York : (1993)

0780304020

[5 Básico] Sistemas operativos: aspectos internos y principios de diseño /

William Stallings.

Prentice Hall,, Madrid [etc.] : (2005) - (5ª ed.)

84-205-4462-0

[6 Básico] Sistemas operativos: principios de diseño e interioridades /

William Stallings ; traducción Amalia Oñate Gómez, Ángel González del Alba Baraja.

Prentice Hall,, Madrid : (2001) - (4ª ed.)

84-205-3177-4

Kluwer Academic,, Boston :
0792393619

Organización Docente de la Asignatura

Contenidos	Horas					Competencias y Objetivos
	HT	HP	HCT	HTT	HAI	
Tema 1	2	2	0	2	4	1.1,1.2,1.3,1.4,2.1,2.2
Tema 2	2	2	0	3	4	1.1,2.1,2.2,2.3
Tema 3	4	4	0	6	8	1.1,2.1,2.2,2.3,2.4
Tema 4	4	4	1	5	8	1.2,2.4,2.6
Tema 5	12	6	0	7	18	1.4,2.5,2.7
Tema 6	4	4	1	5	8	1.1,2.6
Tema 7	4	4	0	6	8	1.1,2.7
Tema 8	4	4	1	6	8	1.3,2.7,3.1

Equipo Docente

JUAN FRANCISCO PÉREZ CASTELLANO

(COORDINADOR)

Categoría: TITULAR DE UNIVERSIDAD

Departamento: INGENIERÍA TELEMÁTICA

Teléfono: 928451237 **Correo Electrónico:** juanfrancisco.perez@ulpgc.es

WEB Personal: <http://www.dit.ulpgc.es/usuarios/profes/francis/index.html>

Resumen en Inglés

The goal of this subject is to get knowledge on some computer architecture topics: operating systems, parallel architectures, real-time systems, and virtual machines.