Universidad Católica de Córdoba

Facultad de Ingeniería

Sistemas de Tiempo Real - Año 2002

Titular: Ing. John Coppens

Prácticos: Ing. Alejandro Butti

1 Objetivos específicos

2 Programa sintético

2.1 Unidad 1 - Terminología

Términos más frecuentes, definición de conceptos de concurrencia.

2.2 Unidad 2 - La vida real

Ejemplos sencillos en C, para mostrar las clases de concurrencia. Un ejemplo en assembler para microcontrolador de un microsistema concurrente.

2.3 Unidad 3 - Controladores PID

Introducción al control de procesos industriales. Algoritmo Proporcional/Integral/Diferencial.

2.4 Unidad 3 - Pascal FC

Alan Burns desarrolló Pascal-FC como herramienta para la enseñanza de las herramientas clásicas de comunicación inter-tarea. Instalación, uso y limitaciones.

2.5 Unidad 4 - Procesos

Conceptos de procesos - secuenciales y concurrentes. Representación matemática. Condiciones especiales y consecuencias.

2.6 Unidad 5 - Exclusión mutua y sincronización

Estudio detallado de las herramientas clásicas de exclusión mutua y sincronización. Implementación en Pascal-FC, discusión y resolución de problemas.

2.7 Unidad 6 - Otra vez, la vida real

Como se implementan las herramientas en la vida real. Algunos ejemplos de uso en sistemas operativos comunes, y algunos ejemplos de sistemas operativos específicamente desarrollados para concurrencia/tiempo real.

Ejemplo de uso de PLC en la implementación de control industrial.

3 Programa analítico

3.1 Unidad 1 - Terminología

Unidad 1 - Terminología - clasificación de programas por funcionamiento: cooperativos - preemptivos, por respuesta: sincrónicos - asincrónicos, por punto de implementación: hardware - firmware - sistema operativo - compilador - programa usuario.

3.2 Unidad 2 - La vida real

Ejemplos sencillos de diferentes formas de concurrencia en C. Un ejemplo de un sistema concurrente sencillo en assembler para microcontroladores. Ejemplo de un sistema operativo preemptivo implementado en C.

3.3 Unidad 3 - Controladores PID

Descripción y abstracción de un proceso industrial - realimentación proporcional - corrección de los errores mediante función integral - aumento de velocidad por función diferencial - motor simulado.

3.4 Unidad 3 - Pascal FC

Introducción de Pascal-FC como herramienta didáctica. Descripción del compilador, sintaxis, uso, limitaciones. Uso de la versión original DOS, y de la versión adaptada en la Universidad para uso bajo Windows.

3.5 Unidad 4 - Procesos

El concepto de 'proceso' - estructuras de procesos - representación en Pascal-FC - representación simbólica - consecuencias de concurrencia.

3.6 Unidad 5 - Exclusión mutua y sincronización

La necesidad de comunicación entre procesos - y los problemas. Mensajería sincrónica. Semáforos. Regiones críticas y monitores.

3.7 Unidad 6 - Otra vez, la vida real

Comparación de la implementación de exclusión mutua y sincronización en sistemas de la vida real: sistemas operativos e idiomas. Como se implementa control de procesos en la industria: introducción a los PLC.

4 Bibliografía obligatoria

- *Concurrent Programming* Allan Burns, Geoff Davies, Addison Wesley, ISBN 0-201-54417-2
- *Modern Operating Systems* Andrew S. Tanenbaum, Prentice Hall, ISBN 0-13-588187-0

5 Bibliografía de consulta

- *Algorithms + Data Structures = Programs*
Niklaus Wirth, Prentice Hall, Inc., ISBN 0-13-022418-9
- *Programming and Problem Solving with Pascal*
Michael Schneider, Steven Weingart, David Perlman, John Wiley, ISBN 0-471-08216-3
- *Advanced Programming and Problem Solving with Pascal*
Michael Schneider, Steven Bruell, John Wiley, ISBN 0-471-01128-2
- *Advanced Programming in the UNIX Environment*
Richard Stevens, Addison Wesley, ISBN 0-201-56317-7
- *Writing Windows Virtual Device Drivers*
David Thielen, Bryan Woodruff, Addison Wesley, ISBN 0-201-48921-X
- *Beginning Linux Programming*
Neil Matthew, Rick Stones, Wrox Press Ltd, ISBN 1-874416-68-0
- *Linux Device Drivers, 2nd edition*
Alessandro Rubini, Jonathan Corbet, O'Reilly, ISBN 0-596-00008-1

6 Metodología

El aspecto teórico será presentado con exposiciones didácticas, con énfasis sobre la implementación de ejemplos. La mayoría de los ejemplos se realizarán mediante Pascal-FC, pero casos donde sea más apropiado, se utilizarán otros medios.

En particular, la parte práctica del curso se dedicará a implementar trabajos bajo condiciones reales:

1. Estudio de un sistema operativo didáctico en C
2. Implementación de un protocolo de comunicación
3. Estudio e implementación de algoritmos PID
4. Control de módulos didácticos

7 Calendario

Fecha	Unidad	Tema/actividad
04/03		Introducción a la materia
12/03	1	Terminología
19/03	2	La vida real
26/03	2	La vida real
02/04	3	Controladores PID
09/04	4	Pascal FC
16/04	5	Procesos
23/04		<i>Primer parcial</i>
30/04	6	Exclusión mutua y sincronización
07/05	6	Exclusión mutua y sincronización
14/05	6	Exclusión mutua y sincronización
21/05	6	Exclusión mutua y sincronización
28/05	7	Otra vez, la vida real
04/06	7	Otra vez, la vida real
11/06		<i>Segundo Parcial</i>
18/06		Revisación de la materia

8 Criterios y formas de evaluación

8.1 Durante el semestre

Tanto en la parte teórica como en la parte práctica, se presentarán por lo mínimo 3 trabajos durante el semestre. Estos trabajos se evaluarán por su: originalidad, solidez de implementación, estructura, documentación.

8.2 Parciales

En la parte teórica se realizarán además dos parciales, evaluando el conocimiento sobre la materia en forma individual.

En las notas finales se agregará una evaluación de la participación del alumno en las actividades.

Final

El examen final teórico consistirá en dos partes: una evolución de los conocimientos teóricos de la materia y una confirmación individual de la participación en la realización de los trabajos presentados.

El responsable de la parte práctica, se encargará de la misma manera de evaluar los conocimientos absolutos e individuales de los alumnos.

8.3 Final

El examen final teórico consistirá en dos partes: una evolución de los conocimientos teóricos de la materia y una confirmación individual de la participación en la realización de los trabajos presentados.

El responsable de la parte práctica, se encargará de la misma manera de evaluar los conocimientos absolutos e individuales de los alumnos.

9 Condiciones para obtener la regularidad

Las condiciones formales son las que actualmente rigen para todas las cátedras, según el reglamento vigente.

10 Trabajos prácticos

- Para la parte teórica: resolución de problemas de concurrencia, principalmente utilizando el compilador Pascal-FC, para evaluar la comprensión de la materia y de los conceptos.
- Para la parte práctica en sí: estudio de un sistema operativo de tiempo real pre-emptivo (en su momento desarrollado por alumnas de la Universidad). Adición de tareas simples al sistema operativo. Desarrollo de un controlador para procesos industriales. Énfasis sobre la programación estructurada. Simulación de un motor. Implementación del algoritmo PID. Comunicación con los módulos didácticos. Graficación.