AMPLIACIÓN DE SISTEMAS OPERATIVOS

Prácticas y Talleres

Prof. Luis Alberto Aranda

Universidad Rey Juan Carlos

CRITERIOS DE EVALUACIÓN

- Hay dos prácticas y dos talleres
  - Práctica 1 (≥ 5): 15%
  - Práctica 2 (≥ 5): 15%
  - Taller 1: 10%
  - Taller 2: 10%

  TOTAL **50%**

- Convocatoria extraordinaria
  - Se guarda la nota de aquellas actividades aprobadas
  - No es posible presentarse a “subir nota”

Universidad Rey Juan Carlos

CRITERIOS DE EVALUACIÓN

- Grupos de 2 o 3 personas máximo
- Se realiza una entrega (PDF) por grupo
- Códigos anexados en el propio PDF
- Copia implica el suspenso inmediato
- **NO** habrá examen de prácticas

Universidad Rey Juan Carlos

AMPLIACIÓN DE SISTEMAS OPERATIVOS

Práctica 1 - Minikernel

Prof. Luis Alberto Aranda

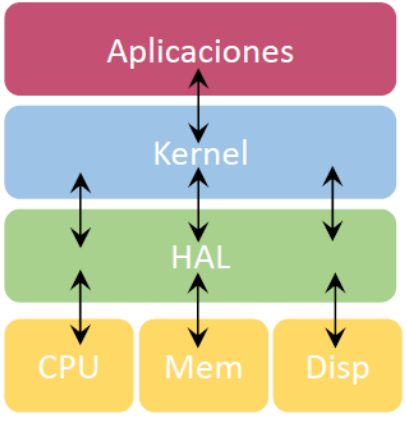
Universidad Rey Juan Carlos

FUNCIONAMIENTO DEL MINIKERNEL

`minikernel/usuario`

`minikernel/minikernel/kernel`

`minikernel/minikernel/HAL`



Carga de Sistema Operativo:

1. Programa de arranque (`minikernel/boot/boot`)
2. Inicializar Sistema Operativo (`minikernel/minikernel/kernel`)
3. Punto de entrada inicial (`minikernel/usuario/init`)

ENTORNO DE PRÁCTICAS

- **HAL** (`minikernel/HAL`): implementa la capa de hardware
- **Programa cargador** (`boot/boot`): carga el sistema operativo en memoria
- **Kernel** (`minikernel/kernel`): contiene la funcionalidad del SO
- **Programas de usuario** (`usuario`): conjunto de programas de ejemplo (init)

- `Make` y `GCC` deben estar instalados para poder compilar el kernel
- Una vez compilado, se puede ejecutar el Sistema Operativo usando:

```
boot/boot minikernel/kernel
```

O bien de la siguiente forma para escribir el output en un fichero de texto:

```
boot/boot minikernel/kernel > salida
```

VERSIÓN INICIAL DEL MINIKERNEL

- La versión inicial tiene algunas **partes ya completas**:
  - Iniciación
  - Tratamiento de excepciones
  - Infraestructura de llamadas al sistema

- Y otras **partes que hay que modificar/completar**:
  - Tratamiento de interrupciones externas y software
  - Llamadas al sistema (sólo hay 3 implementadas: crear_proceso, terminar_proceso y escribir)
  - Estructuras de datos (sólo hay 4 implementadas: BCP, tabla de procesos, cola de procesos, cola de listos)

INCLUIR NUEVAS LLAMADAS AL SISTEMA

1. Incluir en `minikernel/kernel.c` una rutina con la nueva llamada al sistema
2. Incluir en `minikernel/include/kernel.h` la nueva llamada al final de `tabla_servicios`
3. Modificar `minikernel/include/llamsis.h` para incrementar el número de llamadas disponibles y asignar el código más alto a la nueva llamada
4. Una vez realizados los pasos anteriores, el sistema operativo ya incluiría el nuevo servicio, pero sólo sería accesible desde los programas usando código ensamblador. Por lo tanto, es necesario modificar `usuario/lib/serv.c` para que proporcione la nueva interfaz para el nuevo servicio
5. También hay que modificar `usuario/include/servicios.h` para que se disponga del prototipo de la función de interfaz

EJEMPLO: FUNCIÓN OBTENER ID

- Objetivo: proporcionar al usuario una función obtener_id_pr
  - Sistema Operativo
    - `Minikernel/kernel.c y include/kernel.h` → Incluir función y cabecera
    - Manejador
      - Tabla de servicios (`include/kernel.h`) → Incluir rutina
      - Servicios (`include/llamsis.h`)→ Incluir servicio
  - Usuario
    - `Usuario/lib/serv.c` → Incluir invocación
    - `Usuario/servicios.h` → Incluir cabecera
  - Implementar un programa en usuario
    - `Usuario/prueba.c` →Donde se haga uso de la función obtener_id_ip
  - Makefile

TAREAS A REALIZAR

1. (2 puntos) Llamada que bloquea un proceso durante un determinado tiempo
2. (5 puntos) Ofrecer un servicio de sincronización basado en mutex
3. (3 puntos) Sustituir el algoritmo de planificación FIFO por un Round-Robin

El fichero init.c contiene diversas pruebas para que el alumno pueda verificar el correcto funcionamiento de sus implementaciones

Sólo hay que comentar/descomentar las partes correspondientes en el fichero init.c

Realizar un documento PDF explicando los cambios realizados y los resultados obtenidos. Anexar únicamente los códigos modificados en el PDF.