

Nombre de la asignatura: Sistemas Operativos I

Créditos: 3-2-5

Aportación al perfil

Esta asignatura apoyará al alumno en la consecución de las siguientes competencias:

- Aplicar conocimientos científicos y tecnológicos en la solución de problemas en el área informática con un enfoque interdisciplinario.
- Administrar las tecnologías de la información, para estructurar proyectos estratégicos.
- Aplicar normas, marcos de referencia y estándares de calidad y seguridad vigentes en el ámbito del desarrollo y gestión de tecnologías y sistemas de información.
- Integrar las soluciones de tecnologías de información a los procesos organizacionales para fortalecer objetivos estratégicos.
- Seleccionar y utilizar de manera óptima técnicas y herramientas computacionales actuales y emergentes.
- Identificar, diseñar, desarrollar los mecanismos de almacenamiento, distribución, visualización y manipulación de la información.

Objetivo de aprendizaje

- Seleccionar apropiadamente sistemas operativos que formen parte de plataformas de software aplicadas al incremento de la productividad de las organizaciones.

Temario

1. Introducción a los sistemas operativo

- 1.1. Definición de sistema operativo
 - 1.1.1. Principios y funciones del SO
 - 1.1.2. Administrador de recursos
 - 1.1.3. Proveedor de recursos
 - 1.1.4. Máquina virtual
- 1.2. Áreas de responsabilidad del sistema operativo
 - 1.2.1. Procesos
 - 1.2.2. Memoria
 - 1.2.3. Dispositivos de I/O
 - 1.2.4. Sistema de archivos
 - 1.2.5. Redes
 - 1.2.6. Interfaces de usuario
- 1.3. Evolución de los sistemas operativos
 - 1.3.1. Sistemas operativos por lotes
 - 1.3.2. Sistemas operativos de tiempo compartido
 - 1.3.3. Sistemas de escritorio
 - 1.3.4. Sistemas multiprocesadores
 - 1.3.5. Sistemas operativos distribuidos
 - 1.3.6. Sistemas agrupados
 - 1.3.7. Sistemas de tiempo real
 - 1.3.8. Sistemas portátiles
- 1.4. Técnicas de organización de los sistemas operativos
 - 1.4.1. Diseños monolíticos
 - 1.4.2. Diseños por capas
 - 1.4.3. Diseños de microkernels
 - 1.4.4. Módulos
 - 1.4.5. Máquina virtual

2. Administración de procesos (pp 38)

- 2.1. Concepto de proceso
- 2.2. Realización de proceso
 - 2.2.1. Operaciones de procesos
 - 2.2.2. Estados del proceso
 - 2.2.3. Tabla de procesos
- 2.3. Planificación de procesos
 - 2.3.1. Primero que llega, primero en servirse
 - 2.3.2. Primero el trabajo más corto
 - 2.3.3. Round Robin
 - 2.3.4. Programación por prioridad
 - 2.3.5. Ajuste de parámetros de programación
 - 2.3.6. Programación de dos niveles
 - 2.3.7. Programación para tiempo real
 - 2.3.8. Programación en sistemas empujados
- 2.4. Intercambio de contexto
- 2.5. Creación y terminación de procesos

- 2.6. Hilos
- 2.7. Secciones críticas
- 2.8. Interbloqueos

3. Administración de memoria (pp 48)

- 3.1. Introducción
- 3.2. Espacio de direcciones
 - 3.2.1. Vinculación
 - 3.2.2. Cargadores estático y dinámico
 - 3.2.3. Vinculaciones estática y dinámica
- 3.3. Asignación de memoria contigua
 - 3.3.1. Particiones fijas
 - 3.3.2. Particiones dinámicas
 - 3.3.3. Intercambio
- 3.4. Asignación de memoria no contigua
 - 3.4.1. Paginación
 - 3.4.2. Segmentación
- 3.5. Memoria virtual
 - 3.5.1. Conceptos básicos
 - 3.5.2. Localidad de los procesos
 - 3.5.3. Uso de segmentos
 - 3.5.4. Protección de memoria
 - 3.5.5. Memoria compartida
 - 3.5.6. Traducción de direcciones
 - 3.5.7. Consideraciones del tamaño de la página
- 3.6. Paginación con memoria virtual
 - 3.6.1. Políticas de paginado
 - 3.6.2. Asignación de marcos
 - 3.6.3. Fallas con páginas y asuntos de desempeño
- 3.7. Algoritmos de paginación
 - 3.7.1. Algoritmos de paginación estática
 - 3.7.2. Algoritmos de paginación dinámica

4. Administración de dispositivos de I/O (pp 25)

- 4.1. Elementos del subsistema de I/O
- 4.2. Características de hardware de dispositivos de I/O
 - 4.2.1. Manejadores de disco
 - 4.2.2. Comunicaciones seriales
 - 4.2.3. Técnicas de interface del controlador
- 4.3. Tipos de dispositivos de I/O
 - 4.3.1. Comunicación vs dispositivos de almacenamiento
 - 4.3.2. Flujo vs dispositivos de block
- 4.4. Objetivos del diseño del subsistema de I/O
- 4.5. Servicios de dispositivos de I/O
- 4.6. Estructura del controlador de dispositivo
- 4.7. Técnicas de administración de dispositivos
 - 4.7.1. Uso de “buffers”
 - 4.7.2. Intercalado (“interleaving”)
 - 4.7.3. Algoritmo “Elevator”

- 4.7.4. RAID
- 4.7.5. Marcas de agua
- 4.7.6. Procesamiento de entrada humana
- 4.7.7. Pseudo-dispositivos

5. Sistema de archivos

- 5.1. Servicios del sistema de archivos
 - 5.1.1. Accesos compartido y exclusivo
 - 5.1.2. Patrones de acceso
 - 5.1.3. Estructura de archivo
 - 5.1.4. Metadatos
 - 5.1.5. Archivos mapeados en memoria
- 5.2. Diseño del sistema general de archivos
 - 5.2.1. Forma del sistema de archivos
 - 5.2.2. Estructuras de datos principales
- 5.3. Espacio de nombres
 - 5.3.1. Especificadores de controladores
 - 5.3.2. Especificadores de cuentas
 - 5.3.3. Asignación jerárquica de nombres
 - 5.3.4. Extensiones de archivos
 - 5.3.5. Versiones de archivo
 - 5.3.6. Archivos especiales y directorios
 - 5.3.7. Asignaciones de nombres relativa y absoluta
- 5.4. Administración del espacio de almacenamiento
 - 5.4.1. Metadatos del sistema de archivos
 - 5.4.2. Unidades de datos
 - 5.4.3. Administración del espacio libre
 - 5.4.4. Archivos regulares
 - 5.4.5. Archivos escasos
 - 5.4.6. Bifurcaciones
 - 5.4.7. Directorios
 - 5.4.8. Alias
- 5.5. Verificación de consistencias
- 5.6. Registro diario y el sistema estructurado de registro
- 5.7. Memoria caché para bloques

Definición de las competencias específicas

1. Identificar las funciones, evolución, organización, el proceso de arranque y las formas en que se vincula el sistema operativo con su ambiente, reconociendo estas características generales en el estudio de un sistema operativo en particular.
2. Aplicar los conceptos básicos de administración de procesos en el análisis de las características particulares de un sistema operativo.

Sugerencias didácticas transversales

1. Fomentar actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes.
2. Relacionar los contenidos de esta asignatura con las demás del plan de estudios para desarrollar una visión interdisciplinaria en el estudiante.
3. Llevar a cabo actividades prácticas que promuevan el desarrollo de habilidades para la experimentación, tales como: observación, análisis, la aplicación de lenguajes de programación en la solución de problemas elementales, la contrastación del comportamiento y aplicación de los conceptos entre lenguajes de programación diferentes aplicados a la solución de un mismo problema.
4. Propiciar, en el estudiante, el desarrollo de actividades intelectuales de inducción-deducción y análisis-síntesis, las cuales lo encaminan hacia la investigación, la aplicación de conocimientos y la solución de problemas.
5. Propiciar el uso adecuado de conceptos y de terminología científico-tecnológica.

Prácticas

1. Instalar y configurar dos o más lenguajes que soporten el paradigma de la programación concurrente, resolviendo grupalmente, catedrático y alumnos, uno o dos ejercicios de las actividades de aprendizaje.
2. Instalar y configurar dos o más lenguajes que soporten el paradigma de la programación con “scripts”, resolviendo grupalmente, catedrático y alumnos, uno o dos ejercicios de las actividades de aprendizaje.
3. Instalar y configurar uno o más lenguajes que soporten el paradigma de la programación funcional, resolviendo grupalmente, catedrático y alumnos, uno o dos ejercicios de las actividades de aprendizaje.
4. Seleccionar grupalmente, catedrático y alumnos, un lenguaje apropiado para un caso propuesto por el maestro, aplicando los criterios vistos en clase.

Criterios de evaluación

La evaluación puede hacerse considerando los siguientes criterios:

1. La aplicación de exámenes de conocimientos
2. Tareas tales como ensayos, resúmenes, informes, etcétera.
3. Solución a los casos propuestos por el catedrático.
4. Ejercicios realizados en los diferentes lenguajes de programación.

Fuentes de información

1. Scott, Michael L.; ***Programming Language Pragmatics***; Third edition; ed. Morgan Kaufmann Publishers; Estados Unidos; 2009.
2. Watt, David A.; Findlay, William; ***PROGRAMMING LANGUAGE DESIGN CONCEPTS***; ed. John Wiley & Sons, Ltd; Estados Unidos; 2004.
3. Louden, Kenneth C.; ***Lenguajes de programación: principios y práctica***; segunda edición; México, 2004.
4. Watt, David A.; ***Programming Language Processors in Java: COMPILERS AND INTERPRETERS***; ed. Pearson Education Limited; Inglaterra; 2000.

Actividades de aprendizajes

En equipos de trabajo realizar lo siguiente:

Unidad I

1. Investigar las características generales de dos sistemas operativos señalados por el catedrático, incluyendo en ellas asuntos como los siguientes:
 - 1.1. Origen
 - 1.2. Organización
 - 1.3. Boot
 - 1.4. Estructura
 - 1.5. Llamadas al sistema
2. Resumir la especificación del lenguaje de programación, la interface de programación de aplicaciones (API), la máquina virtual y el ambiente de desarrollo de la tecnología de JAVA.
3. Identificar las principales actividades de un sistema operativo en lo relativo a cada uno de los siguientes asuntos:
 - 3.1. Administración de procesos.
 - 3.2. Administración de memoria.
 - 3.3. Administración del almacenamiento secundario.
 - 3.4. Administración de archivos.
4. Expresar cuáles son los propósitos del intérprete de comandos, las llamadas al sistema y los programas del sistema.
5. Explicar por qué el diseño de micro-kernel es apropiado para los sistemas operativos distribuidos.
6. Explicar la ventaja principal del diseño de sistemas operativos por capas o niveles y la de micro-kernel.

Unidad II

1. Juzgar si el hecho de que una aplicación de Java pueda tener varios hilos es consistente con la noción básica de hilos vista en este curso de sistemas operativos.
2. Describir el contenido real de un descriptor de procesos para dos sistemas operativos señalados por el catedrático.
3. Explicar las diferencias entre procesos, programas e hilos.
4. Explicar los estados básicos de un proceso, juzgando las posibilidades de las existencias de estados adicionales y estados intermedios.
5. Escoger cuáles de los siguientes conceptos pueden ser usados por los procesos de usuario y cuáles deberían ser reservados para uso del sistema operativo, justificando la decisión.
 - 5.1. Métodos de exclusión mutua.
 - 5.2. Des-habilitación de interrupciones.
 - 5.3. Instrucciones “test-and-set”.
 - 5.4. Semáforos.
6. Investigar cómo se implantan los siguientes conceptos en los sistemas operativos señalado por el catedrático.
 - 6.1. Procesos e hilos

- 6.2. Llamadas al sistema en lo relativo a procesos
- 6.3. Estados del proceso
- 6.4. Tabla de procesos
- 6.5. Creación de procesos
- 6.6. Programación de procesos

Unidad III

1. Indicar las diferencias entre los siguientes pares de conceptos:
 - 1.1. Dirección física y dirección lógica
 - 1.2. Fragmentación interna y fragmentación externa
2. Explicar lo siguiente:
 - 2.1. Por qué los tamaños de las páginas son siempre potencias de 2.
 - 2.2. Las razones por las cuales un proceso no puede acceder a memoria que no le pertenece en un sistema con paginación.
 - 2.3. El efecto de permitir que dos entradas en una tabla de páginas apunten al mismo marco en memoria.
 - 2.4. Por qué la segmentación y la paginación algunas veces son combinadas dentro de un esquema.
 - 2.5. Las circunstancias que ocasionan que ocurran fallas de página.
3. Describir los algoritmos “first fit”, “best fit”, “worst fit”.
4. Listar los costos y beneficios de la implementación de la memoria virtual en un sistema de computación, analizando la posibilidad de que los primeros excedan a los segundos.
- 5.
6. Investigar los siguientes asuntos sobre administración de memoria para el sistema operativo indicado por el catedrático.
 - 6.1. Esquemas de memoria (layouts)
 - 6.2. Llamadas al sistema en lo relativo a la memoria
 - 6.3. Mecanismos de asignación de memoria
 - 6.4. Administración de páginas
 - 6.5. Estructuras de datos usadas para la administración de memoria
 - 6.6. Implementación de la administración de memoria

Unidad IV

1. Definir, con sus propias palabras, el concepto de lenguaje “script”.

Unidad V

1. Investigar los siguientes conceptos:
 - 1.1. Polimorfismo paramétrico
 - 1.2. Abstracción de datos
 - 1.3. Evaluación “floja”
 - 1.4. Valores de primera clase en las funciones
 - 1.5. Constructores de objetos estructurados