

**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA**  
**COORDINACIÓN DE FORMACIÓN BÁSICA**  
**COORDINACIÓN DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA**  
**PROGRAMA DE UNIDAD DE APRENDIZAJE**

**I. DATOS DE IDENTIFICACIÓN**

1. Unidad Académica: FACULTAD DE CIENCIAS.
2. Programa (s) de estudio: Licenciatura en Ciencias Computacionales      3. Vigencia del plan: \_\_\_\_\_
4. Nombre de la Asignatura: Sistemas Operativos      5. Clave: \_\_\_\_\_
6. HC: 2    HL 3    HT \_\_\_\_\_    HPC \_\_\_\_\_    HCL \_\_\_\_\_    HE 2    CR 7
7. Etapa de formación a la que pertenece: Disciplinaria
8. Carácter de la Asignatura:      Obligatoria   X        Optativa \_\_\_\_\_
9. Requisitos para cursar la asignatura:

**Formuló:** L.C.C. Roberto Carranco Polo,  
Dra. María Victoria Meza Kubo

**Vo. Bo.** Dr. Alberto Leopoldo Morán y Solares

**Fecha:** Agosto de 2016

**Cargo:** Subdirector

## **II. PROPÓSITO GENERAL DE LA UNIDAD DE APRENDIZAJE**

La finalidad de la asignatura es mostrar al alumno los fundamentos de diseño de un sistema operativo. Principalmente la estructura básica y los mecanismos que contiene un sistema operativo. Dar a conocer al alumno el funcionamiento interno de un sistema operativo.

Es una asignatura de la etapa disciplinaria y de carácter obligatoria.

## **III. COMPETENCIA (S) DE LA UNIDAD DE APRENDIZAJE**

Implementar estrategias de las estructuras de los sistemas operativos, a partir del análisis de sus mecanismos y estrategias para obtener un sistema óptimo, con responsabilidad e iniciativa.

## **IV. EVIDENCIA (S) DE DESEMPEÑO**

Elabora un proyecto final consistente en el desarrollo de una o varias de las estructuras de los sistemas operativos. Los algoritmos deberán simular el comportamiento óptimo de un sistema operativo.

## V. DESARROLLO POR UNIDADES

### Competencia

Emplear los conceptos básicos, definiciones e historia de los sistemas operativos, mediante la lectura de artículos y bibliografía de sistemas operativos, para familiarizarse con sus propiedades, con actitud crítica.

### Contenido temático

#### 1. Conceptos básicos

1.1 Qué es un Sistema Operativo (definición inicial)

1.2 Estructura de un sistema operativo

1.2.1 Intérprete de mandos

1.2.2 Programa supervisor

1.2.3 Bibliotecas de aplicación

1.3 Evolución y tendencias de los Sistemas Operativos

1.4 Hardware, Software y Firmware.

1.5 Tipos de sistemas (tiempo real, tiempo compartido, teleproceso, etc).

1.6 Concurrencia y multiprogramación (definición y ejemplos)

1.7 Modelos de Sistemas (monolítico, en capas, máquinas virtuales, cliente-servidor, etc)

### Duración

4 horas

### Competencia

Emplear el concepto de proceso dentro de un sistema operativo analizando las técnicas y mecanismos del entorno donde se desarrolla para comprender conceptos de multiprogramación y calendarización de cpu, trabajando con respeto y responsabilidad.

### Contenido temático

#### 2. Procesos

2.1 Definición de un proceso

2.2 Estados y transiciones

2.3 Contexto de un proceso

2.4 Operaciones sobre procesos y suspensión/reinicialización

2.5 Procesamiento de interrupciones y cambio de contexto

2.6 Calendarizado del procesador

2.7 Niveles, objetivos y criterios de calendarizado.

2.8 Políticas de calendarización (FIFO, prioridades, SJF, SRF, Round-Robin, ...)

2.9 Hilos

### Duración

4 horas

**Competencia**

Analizar distintas estrategias utilizando herramientas de sincronización, para resolver problemas de competencia entre procesos por un mismo recurso, trabajando con responsabilidad.

**Contenido temático**

- 3. Comunicación entre procesos
  - 3.1. Condiciones de competencia.
  - 3.2. Sección crítica.
  - 3.3. Exclusión mutua con espera ocupada.
  - 3.4. Dormir y despertar.
  - 3.5. Semáforos.

**Duración**

4 horas

**Competencia**

Analizar el problema del bloqueo entre procesos a través del uso de las técnicas que resuelven este problema, para el diseño de los sistemas operativos, trabajando con responsabilidad.

**Contenido temático**

- 4. Abrazo mortal o interbloqueo
  - 4.1 Conceptos introductorios y ejemplos
  - 4.2 Condiciones necesarias para la ocurrencia de un abrazo mortal
  - 4.3 Manejo de abrazo mortal
    - 4.3.1 Prevención
    - 4.3.2 Evasión
    - 4.3.3 Detección y Recuperación

**Duración**

4 hrs.

|   |                 |
|---|-----------------|
| <b>Competencia</b>  |                 |
| Analizar la evolución del manejo y administración de la memoria primaria, haciendo análisis en cada uno de los esquemas, para el diseño de sistemas operativos, trabajando con responsabilidad. |                 |
| <b>Contenido temático</b>   | <b>Duración</b> |
| 5. Organización y manejo de memoria primaria  | 4 horas.        |
| 5.1 Organización y jerarquía del almacenamiento   |                 |
| 5.2 Estrategias de manejo de memoria (de traído, de colocación y de reemplazo)  |                 |
| 5.3 Asignación contigua vs. no contigua y asignación contigua para usuario único  |                 |
| 5.4 Multiprogramación con particiones fijas   |                 |
| 5.5 Multiprogramación con particiones variables   |                 |
| 5.6 Multiprogramación con intercambio (swaping)   |                 |
| <b>Competencia</b>  |                 |
| Administrar la memoria virtual, analizando sus diferentes esquemas, como parte fundamental en un sistema operativo moderno, trabajando con responsabilidad.                                     |                 |
| <b>Contenido temático</b>   | <b>Duración</b> |
| 6. Organización y manejo de memoria virtual   | 4 horas.        |
| 6.1 Almacenamiento virtual o conceptos básicos  |                 |
| 6.2 Organización de almacenamiento multiniveles   |                 |
| 6.3 Mapeo de bloques  |                 |
| 6.4 Paginación  |                 |
| 6.4.1 Estrategias de reemplazo de página  |                 |
| 6.4.2 Localidad y conjuntos de trabajo  |                 |
| 6.4.3 Paginación por demanda  |                 |
| 6.5 Segmentación  |                 |
| 6.6 Organización combinada paginación / segmentación  |                 |

|  |                 |
|--|-----------------|
| <b>Competencia</b>   |                 |
| Analizar las estrategias de manejo de disco, a través de la ejecución de ejemplo, para optimizar el desempeño en cuanto a las solicitudes de acceso, trabajando con responsabilidad. |                 |
| <b>Contenido temático</b>  | <b>Duración</b> |
| 7. Manejo de disco   | 4 horas.        |
| 7.1 Calendarizado de disco   |                 |
| 7.2 Políticas de calendarizado   |                 |
| 7.3 Optimización en búsqueda   |                 |
| 7.4 Optimización rotacional  |                 |
| 7.5 Caché de disco   |                 |
| <b>Competencia</b>   |                 |
| Manejar el sistema de archivos, analizando las estrategias del manejo de archivos, para organizar los archivos en almacenamiento secundario, trabajando con responsabilidad.         |                 |
| <b>Contenido temático</b>  | <b>Duración</b> |
| 8. El sistema de archivos  | 4 horas.        |
| 8.1 Introducción   |                 |
| 8.2 Funciones del sistema de archivos  |                 |
| 8.3 Jerarquía de datos   |                 |
| 8.4 Organización de archivos   |                 |
| 8.5 Manejo de espacio en disco   |                 |
| 8.6 Servidores de archivos   |                 |

#### IV. ESTRUCTURA DE LAS PRÁCTICAS

| No. de Práctica | Competencia(s)   | Descripción   | Material de Apoyo                                   | Duración |
|-----------------|--|---|---|----------|
| 1               | Introducir en el ambiente de cómputo que se manejara en el curso, mediante el acceso a computadoras con sistema operativo unix o Linux, para identificar sus diferencias con otros sistemas operativos, con actitud crítica. | Aprender mandos básicos del entorno unix, haciendo ejercicios de programación en shells.  | Acceso a computadora con unix                       | 6        |
| 2               | Utilizar el ambiente de un proceso, analizando sus estados, transiciones y el calendarizador de cpu, para entender el funcionamiento de estrategias de calendarización de procesos, con actitud crítica y compromiso.        | Simular en un programa de computadora el calendarizador FIFO, mostrando claramente el cambio de estado de cada proceso, utilizando los conceptos de contexto, ráfagas de utilización de cpu, de e/s, etc. | Acceso a computadora con software de programación C | 6        |
| 3               | Utilizar el calendarizador round robin, utilizando la práctica anterior, para entender el funcionamiento de estrategias de calendarización de procesos, con actitud crítica y compromiso.                                    | Modificar el calendarizador de la practica anterior para que funcione como un round robin.  | Acceso a computadora con software de programación C | 6        |
| 4               | Utilizar los mecanismos para solucionar problemas de competencia, utilizando hilos, para entender los problemas de sincronización de procesos, con responsabilidad ey creatividad.   | Implementar un algoritmo que resuelva un problema de competencia visto en clase, utilizando las herramientas de sincronización que ofrecen los hilos.   | Acceso a computadora con software de programación C | 6        |

|   |   |  |   |   |
|---|---|--|---|---|
| 5 | Implementar algoritmos de desbloqueo utilizando los mecanismos conocidos para solucionar los problemas de bloqueo entre procesos, con actitud responsable y crítica.  | Implementar un algoritmo para detectar un abrazo mortal y otro algoritmo para liberar el abrazo mortal   | Acceso a computadora con software de programación C | 6 |
| 6 | Analizar el funcionamiento de las estrategias de manejo de memoria realizando ejercicios y llevando estadísticos para tener un marco comparativo de las estrategias de colocación, con actitud crítica y responsable. | Implementar los algoritmos de colocación, sacando estadística de las solicitudes atendidas y rechazadas, número de compactaciones, para tener un marco comparativo.  | Acceso a computadora con software de programación C | 6 |
| 7 | Implementar un algoritmo de reemplazo aplicando la memoria virtual para comprender su funcionamiento con actitud crítica y responsable.   | Implementar un algoritmo de reemplazo de pagina visto en clase, donde las solicitudes tengan la característica de localidad temporal y espacial, se harán 10 experimentos aumentando el número de marcos disponibles para cada experimento, se debe sacar estadística del numero de reemplazos vs el numero de marcos disponibles. | Acceso a computadora con software de programación C | 6 |
| 8 | Analizar los algoritmos de calendarizado de disco, haciendo énfasis en los algoritmos de optimización de búsqueda, para comparar las estrategias de calendarizado, con actitud crítica y responsable.                 | Implementar al menos 3 algoritmos de calendarizado de disco, sacando estadística como tiempo de respuesta, numero de solicitudes, etc, que nos permita comparar estas estrategias.   | Acceso a computadora con software de programación C | 6 |

## VII. METODOLOGÍA DE TRABAJO

### ■ Aprendizaje participativo

Durante la clase se aplicará esta metodología en la que el estudiante juega un papel activo al intervenir propositivamente en la planeación, realización y evolución del proceso de aprendizaje. Consiste básicamente en asignar problemas, el cual primeramente es analizado en forma individual, posteriormente en equipo, después se resuelven los problemas en clase. La participación del maestro en la aplicación de esta metodología es de mediador.

### ■ Trabajo en equipo

A lo largo del semestre se estará trabajando en equipo, tanto para los trabajos en clase, así como para las prácticas y el proyecto final, y consiste básicamente en asignar problemas para la cual deberán organizarse y resolverlos. El maestro en esta metodología juega el papel de mediador al igual que en la anterior.

### ■ Investigación

Esta será empleada en las exposiciones y proyecto final, constará de dos partes, la presentación por escrito de la investigación y la presentación y defensa oral de la misma. AL investigación podrá ser a través de consulta de libros, revistas, sitios de prestigio en Internet y personas con conocimientos en el tema.

### ■ Clase expositiva

Ésta en el caso del alumno, será aplicada sobretodo en la exposición de la solución de los problemas asignados, en el caso del maestro se aplicará en la impartición de conceptos básicos, explicación de ejercicios, así como en la conclusión de temas.

Se requerirá que los estudiantes realicen horas extraclase para la resolución de los ejercicios que se dejen, así como que utilicen horas de cómputo extra para la realización de las prácticas, por lo menos requerirán 4 horas a la semana extra de uso de computadora cuando sean asignadas prácticas de laboratorio.

## VIII. CRITERIOS DE EVALUACIÓN

### Criterio de calificación

|                        |      |
|------------------------|------|
| ■ Exámenes parciales   | 40%  |
| ■ Prácticas            | 30%  |
| ■ Tareas, exposiciones | 20%  |
| ■ Proyecto final       | 10%  |
| TOTAL                  | 100% |

### Criterio de acreditación

- Resolver un examen parcial por cada unidad en tiempo y forma.
- Cumplir con las prácticas de laboratorio, tareas y exposiciones en tiempo y forma.
- Cumplir con la presentación y reporte de un proyecto en tiempo y forma.
- Para la acreditación del curso se atenderá al Estatuto Escolar Vigente, artículos 70-71, por lo que el estudiante deberá contar un mínimo de 80% de asistencias en el periodo. Tener un mínimo aprobatorio de 60 en su calificación final.

### Criterio de evaluación

- Los exámenes y tareas deberán ser resueltos en clase posterior a su entrega para realimentar el desarrollo de la asignatura.
- Las prácticas deberán ser evaluadas según los requerimientos solicitados, y la entrega del código fuente, así como del reporte con las conclusiones de las mismas.

Proyecto final se deberá calificar la originalidad, presentación y desarrollo del tema, independientemente de ser práctico o teórico.

## IX. BIBLIOGRAFÍA

### Básica

- Anderson, Thomas and Dahlin, Michael (2014). Operating Systems: Principles and Practice 2nd Edition. Recursive Books.
- Deitel, H. M., Deitel, P. J., & Choffnes, D. R. (2008). Operating systems. Pearson/Prentice Hall. [CLÁSICO]
- Silberschatz, A., GALVIN, G., Silberchatz, P. B., Galvin, P. B., & Gagne, G. (2012). Fundamentos de sistemas operativos. McGraw-Hill,.
- Tanenbaum, A. S. (2009). Sistemas operativos modernos. 3ra ed. Pearson Educación. [CLÁSICO]

### Complementaria

- Revistas: Dr. Dobb's Journal Comunicaciones de ACM/Sistemas Operativos IEEE Concurrency. <http://www.drdobbs.com/>
- Recursos para el profesor: <http://recursosmcc.pearsonenespanol.com/Tanenbaum/>

## X. PERFIL DOCENTE

El docente de esta asignatura deberá ser un profesionalista con formación en el área de computación o áreas afines; con experiencia en sistemas operativos, redes y seguridad en cómputo. Además deberá tener experiencia docente.