

Carrera: Analista Programador

Arquitectura de Sistemas Operativos

Módulo I

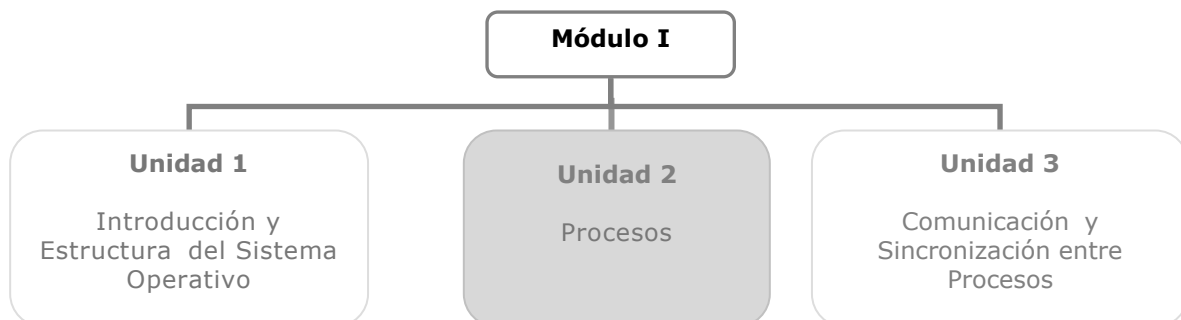
Aprender el objetivo y las funciones del sistema operativo, la planificación del procesador, la comunicación y sincronización entre procesos.

Unidad 2

Procesos

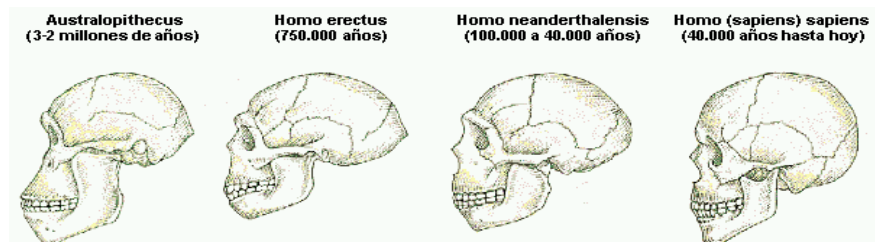
Profesor Titular: Martín Mario.

Autor de contenidos: lic. Juan Carlos Romero.



Presentación

Para presentar esta unidad me gustaría que recordemos la siguiente imagen de la evolución de la de estructura craneana de los habitantes de la tierra.



Y en función de ella, invítalos a pensar en el hombre moderno: un individuo que habita la tierra y que vive en una sociedad que él mismo creó. Reflexionar sobre su capacidad de construir, destruir, crear; relacionarse con otros y formar organizaciones con individuos que perdiguen el mismo fin.

El ser humano nace, se reproduce y muere. En ese ciclo de vida, pasa por diferentes estados, entre ellos, uno muy conocido por todos nosotros: el estado de trabajar, para el cual demanda infinidad de recursos que utiliza para transformar otros.

En este momento: usted *¿qué recursos está utilizando para realizar su tarea? ¿Qué cambios o transformaciones haría para que, estos recursos, se adapten a sus necesidades?*

Lo invito a imaginar un Sistema de Computación como un mundo físico conformado por un conjunto de recursos que fueron puestos ahí para que el microsistema cumpla una misión. Podemos conjeturar que tenemos un planeta con todas sus bellezas naturales pero falta algo para que la existencia de este planeta particular tenga sentido -sí lo esta pensando bien- para lograr la representación que buscamos falta aquello que signifique a los individuos y la relación entre ellos.

Entonces, *¿quiénes son los habitantes del planeta que llamamos "Sistema de Computación"?*

Los vamos a llamar **Procesos**, ellos también nacen, se reproducen o no, mueren y trabajan. Algunos SO relacionan a los procesos como padres e hijos y todos ellos pertenecen a una gran familia dentro del Sistema, hasta se habla de procesos huérfanos y zombies; cuando nace o se crea un proceso, el SO entrega el número del DNI (del proceso creado) al proceso que lo creo.



Es decir, podemos pensar en una sociedad de procesos que interactúan entre sí, que trabajan incansablemente para los usuarios del microsistema; y que para realizar su trabajo en forma cooperativa necesitan comunicarse, sincronizarse y hacer uso de los recursos del sistema.

Por todo lo expresado hasta aquí es que esperamos que usted, a través del estudio de esta unidad, adquiera capacidad para:

- Comprender al trabajador incondicional del sistema, al cual llamaremos proceso.
- Comprender el ciclo de vida, los estados del proceso y la ventaja de la multitarea.
- Apreciar los beneficios que un SO aporta con la planificación de procesos o planificación del procesador.
- Crear procesos padres e hijos y definir procesos huérfanos y zombies.
- Realizar una sincronización básica entre proceso padre y proceso hijo.
- Aprender el uso de las señales, las excepciones y los temporizadores.
- Conocer la diferencia entre procesos pesados y livianos y las ventajas y desventajas de los mismos.

A continuación, le presentamos un detalle de los contenidos y actividades que integran esta unidad. Usted deberá ir avanzando en el estudio y profundización de los diferentes temas, realizando las lecturas requeridas y elaborando las actividades propuestas, algunas de desarrollo individual y otras para resolver en colaboración con otros estudiantes y con su profesor tutor.

Contenidos y Actividades

1. Proceso: multitarea, estado e información del proceso.



Lectura requerida

- Silberschatz A. y Galvin P.; Capítulo 4. Procesos. Página 89. **En su: Sistemas Operativos**; 5ta Edición; México Addison Wesley; 1999.



Lectura Sugerida

- Stallings W.; Capítulo 3. Descripción y control de procesos. Página 108. **En su: Sistemas Operativos –Aspectos Internos y principios de diseño–**; 5ta Edición; Prentice Hall 2007.
- Silberschatz A. Galvin P. Gagne G.; Capítulo 3. Procesos. Página 81. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.
- Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada–**; Mc Graw Hill.



Trabajo Práctico Sugerido

- Trabajo Práctico Nº 2: **Introducción a Procesos**

2. Ppanificación de procesos o planificación del procesador.



Lectura requerida

- Silberschatz A. y Galvin P.; Capitulo 5. Planificación de la CPU. Página 123. **En su: Sistemas Operativos**; 5ta Edición; México Addison Wesley; 1999.



Lectura Sugerida

- Stallings W.; Capítulo 3. Descripción y control de procesos. Página 108. **En su: Sistemas Operativos –Aspectos Internos y principios de diseño-**; 5ta Edición; Prentice Hall 2007.
- Silberschatz A. Galvin P. Gagne G.; Capítulo 5. Planificación de CPU. Página 153. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.
- Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada-**; Mc Graw Hill.



Trabajo Práctico Sugerido

- Trabajo Práctico Nº3 : **Planificación del procesador**

3. Señales, excepciones y temporizadores.



Lectura requerida

- Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada-**; Mc Graw Hill.

4. Servicios POSIX para el manejo de procesos.



Lectura requerida

- Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada-**; Mc Graw Hill.



Lectura Sugerida

- Silberschatz A. y Galvin P.; Capitulo 4. Procesos. Página 89. **En su: Sistemas Operativos**; 5ta Edición; México Addison Wesley; 1999.
- Silberschatz A. Galvin P. Gagne G.; Capítulo 3. Procesos. Página 81. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.



Trabajo Práctico Sugerido

- Trabajo Práctico Nº 4: **Programación de procesos**

5. Procesos livianos o hilos.



Lectura requerida

- Silberschatz A. y Galvin P.; Capitulo 4. Procesos. Página 89. **En su: Sistemas Operativos**; 5ta Edición; México Addison Wesley; 1999.



Lectura Sugerida

- Stallings W.; Capítulo 4. Hilos, SMP y micronúcleos. Página 158. **En su: Sistemas Operativos –Aspectos Internos y principios de diseño-**; 5ta Edición; Prentice Hall 2007.
- Silberschatz A. Galvin P. Gagne G.; Capítulo 4. Threads. Página 127. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.
- Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada-**; Mc Graw Hill.

Cierre de la unidad

ANEXO

Para el estudio de estos contenidos usted deberá consultar la bibliografía que aquí se menciona:

BIBLIOGRAFÍA OBLIGATORIA

- **Silberschatz A. y Galvin P.; Sistemas Operativos; 5ta Edición; Addison Wesley 1999.**
- **Stallings W.; Sistemas Operativos –Aspectos Internos y principios de diseño-; 5ta Edición; Prentice Hall 2007.**

Bibliografía Ampliatoria

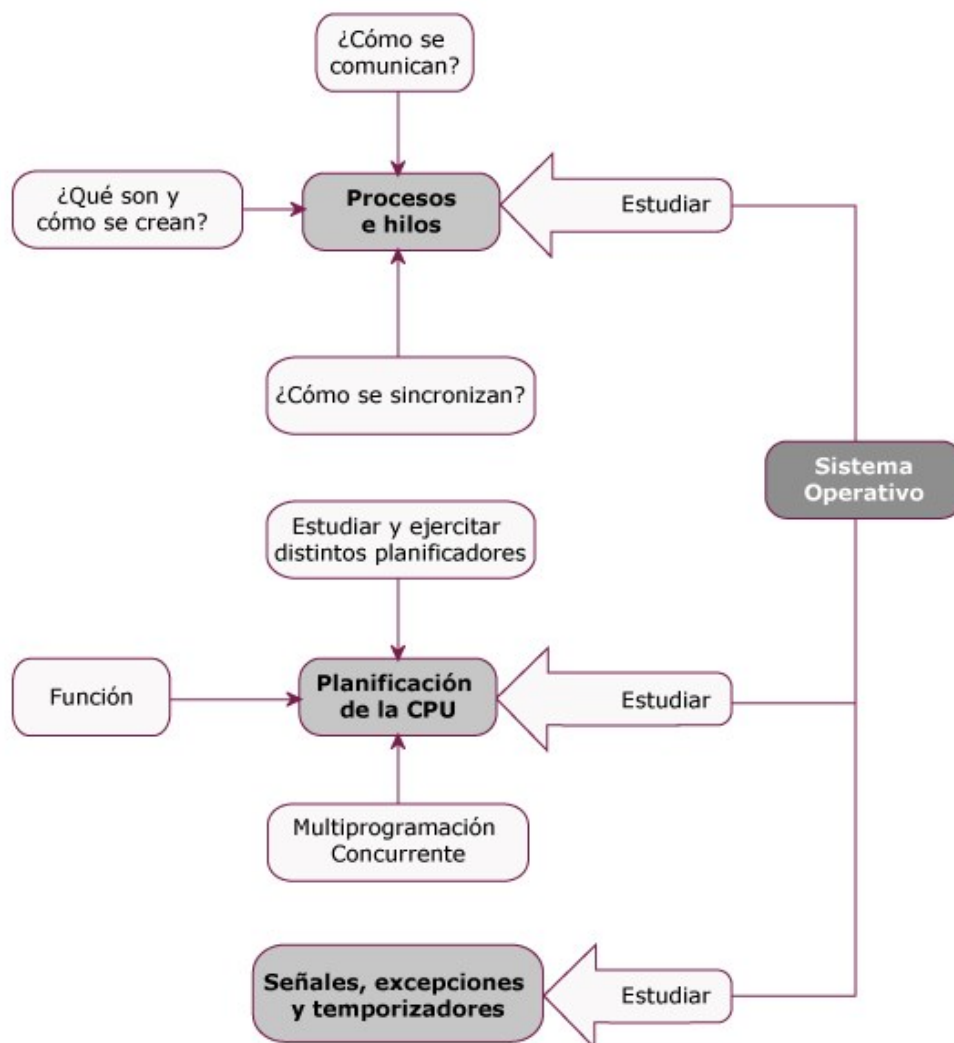
- **Silberschatz A. Galvin P. Gagne G.; Fundamentos de Sistemas Operativos; 7ma Edición; Mc Garw Hill 2007.**



- **Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Sistemas Operativos –Una visión aplicada-; Mc Graw Hill.**
- **Tanenbaum A. Woodhull A.; Sistemas Operativos –Diseño e implementación-; 2da Edición; Prentice Hall 1997.**

Organizador Gráfico

El siguiente esquema le permitirá visualizar la interrelación entre los conceptos que a continuación abordaremos. Le sugerimos que vuelva a este organizador luego de completar el estudio de la unidad, le ayudará a ordenar sus ideas.



Lo/a invitamos ahora a comenzar con el estudio de los contenidos que conforman esta unidad.

1. Proceso: multitarea, estados e información del proceso.

Los procesos son los elementos de software que desarrollan las tareas dentro de un sistema de computación. Seguramente usted ya ha lanzado a ejecución algún proceso, por ejemplo ha usado el procesador de textos Word o alguno similar que le ha permitido presentar algún lindo informe a su jefe o una carta de amor a su querida esposa.

Los procesos se pueden ejecutar de a uno o de a varios en forma simultánea o concurrente. Desde que comienzan su ejecución hasta que terminan su tarea pasan por distintos estados y como estos procesos son administrados por el SO, él necesita tener información del ciclo de vida de éstos.

Para tratar este tema podemos presentar la siguiente situación:

En una sociedad la mayoría de los individuos trabajan, entre ellos, existen quienes realizan más de una tarea en **forma concurrente no simultánea**. Usted se preguntará qué significa realizar este tipo de tareas.

Para comprender este tipo de acción será necesario, en primer lugar, identificar la diferencia entre **tarea concurrente y simultánea**. Es por ello que las describiré con una metáfora para facilitar y promover una representación mental de ellas.

- **Realizar más de una tarea en forma concurrente:** se realiza parte de cada una de las tareas en distintos instantes de tiempo, por ejemplo: un cocinero podría hacer el repulgue de "x" cantidad de empanadas al mismo tiempo que hornea un pastel. Y decorar el pastel mientras hornea las empanadas.
- **Realizar más de una tarea en forma simultánea:** las diferentes tareas se realizan en el mismo instante de tiempo, en el caso de nuestro cocinero, si tuviera dos hornos y un cocinero ayudante, podría realizar el pastel y las empanadas en el mismo instante de tiempo.

Luego de comprender la diferencia entre las dos tareas, lo invitamos realizar la siguiente actividad.



Actividades para la facilitación de los aprendizajes

A partir de las definiciones de las tareas de forma concurrente y simultánea le proponemos que plantee una metáfora para explicar cómo se imagina usted que se realiza una tarea concurrente no simultánea.



Estas formas de llevar adelante las tareas son las que comúnmente utiliza un SO, dependiendo de la cantidad de procesadores que posea el sistema de computación.

El SO es quien administra los procesos entonces el deberá mantener información actualizada del estado del proceso de su identificador único, de su prioridad, etc. Igual que en la sociedad hay organizaciones gubernamentales que poseen información actualizada de los individuos que pertenecen a esta sociedad, por ejemplo: cuando un individuo nace, sus tutores lo deben anotar con un nombre permitido en el registro civil, y el Estado le asigna un número; cuando se casa, nuevamente pasa por el registro civil y se produce un cambio de estado, del estado de soltero a pasa al estado casado; y cuando muere se debe registrar su defunción, de lo contrario el Estado no tiene conocimiento que el individuo ya no forma parte de la sociedad. Recuerdo que yo tenía un vecino que falleció, y durante muchos años le seguían llegando impuestos para pagar, su hall de entrada se llenó de boletas e impuestos.

El caso de mi vecino que falleció es un buen ejemplo de lo que sucede cuando algo falla en la administración, producto de una mala administración de recursos (no se optimiza el uso del papel y los recursos humanos entre otras cosas) Este ejemplo lo podemos relacionar con el estado de proceso zombie en SO.

A continuación, lo/a invitamos a profundizar estos temas en la bibliografía requerida.



Lectura requerida

Silberschatz A. y Galvin P.; Capitulo 4. Procesos. Página 89. **En su: Sistemas Operativos**; 5ta Edición; México Addison Wesley; 1999.



Lectura Sugerida

Stallings W.; Capítulo 3. Descripción y control de procesos. Página 108. **En su: Sistemas Operativos –Aspectos Internos y principios de diseño-**; 5ta Edición; Prentice Hall 2007.

Silberschatz A. Galvin P. Gagne G.; Capítulo 3. Procesos. Página 81. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.



Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada–**; Mc Graw Hill.

Guía para la Lectura

1. Dé su propia definición de proceso.
2. Defina cada uno de los estados de procesos.
3. Realice el diagrama de estados de los procesos en un entorno operativo que brinda monoprogramación.
4. ¿Qué es la multitarea o multiprogramación?
5. Realice el diagrama de estados de los procesos en un entorno operativo que brinda multiprogramación.
6. ¿Qué es el BCP o bloque de control de proceso?
7. ¿Qué es el vector de procesos?
8. ¿Qué información se almacena en el BCP?
9. ¿Cómo llama el SO Linux al BCP y al vector de procesos?
10. ¿Cuáles son los estados de procesos en el SO Linux?

A continuación le sugerimos realizar el siguiente trabajo. Su realización le permitirá consolidar sus conocimientos sobre lo estudiado.



Trabajo Práctico Sugerido

Trabajo Práctico Nº 2: Introducción a Procesos

Usted encontrará las consignas de este Trabajo Práctico en el Anexo que incluimos al final de este Orientador.

Comparta sus dudas e inquietudes con sus pares y con su tutor a través de los medios de comunicación disponibles en el Campus.



2. Planificación de procesos o planificación del procesador

Luego estudiar los conceptos más importante de los procesos, su definición, los diferentes estados por los cuales pasa en su ciclo de vida, la información, que el SO mantiene de ellos para su administración y la forma en que ellos se pueden ejecutar, en esta asignatura nos referimos a la multiprogramación concurrente monoprocesamiento.

Ahora estudiaremos las diferentes formas o planificadores de CPU que el SO puede llegar a utilizar para llevar adelante la ejecución concurrente de los procesos.

Retomemos la metáfora del cocinero "Realizar más de una tarea en forma concurrente: se realiza parte de cada una de las tareas en distintos instantes de tiempo, por ejemplo: un cocinero podría hacer el repulgue de "x" cantidad de empanadas al mismo tiempo que hornea un pastel. Y decorar el pastel mientras hornea las empanadas."

Si seguimos incrementando la cantidad de tareas con los mismos recursos (Cocinero, Cocina, Horno), todo se empieza a complicar y quizás las cosas no salgan tan bien como esperamos, se podrían llegar a quemar las empanadas o el pastel.

Pero si establecemos un procedimiento de orden de resolución de tareas basado en alguna variable por ejemplo asignar prioridades a las tareas y asignarle los recursos en función de dicha prioridad seguramente todo saldría mejor.

Para facilitar la comprensión de este tema lo invitamos a realizar la siguiente actividad.



Actividades para la facilitación de los aprendizajes

Imagine un cardumen de pirañas y una jugosa porción de carne dentro del acuario donde ellas habitan. Todas las pirañas, en una actitud egoísta, tratarían de apoderarse del exquisito bocado. Cambie las pirañas por procesos, la jugosa porción de carne por un microprocesador y el acuario por un sistema de computación. Tal como usted lo estudió, sabe que los procesos necesitan el microprocesador para llevar adelante su tarea, ¿qué pasaría si un proceso, en un momento determinado, por algún motivo se adueñara del microprocesador?

Los procesos durante su ejecución demandan recursos del sistema de computación y muchos de esos recursos son de uso exclusivo, esto quiere decir que mientras el recurso está asignado a un proceso no puede ser asignado a otro proceso.

Si el proceso solicita al SO realizar una lectura o escritura sobre un dispositivo de almacenamiento magnético, el proceso deberá esperar hasta que la solicitud haya sido resuelta para continuar con su tarea.

Las operaciones de lectura o escritura sobre un dispositivo de almacenamiento magnético no son resueltas por el procesador principal o CPU.

A partir de estos conceptos, lo/a invitamos a reflexionar sobre los siguientes interrogantes:



Actividades para la facilitación de los aprendizajes

- ¿Qué tarea se podría realizar para optimizar el uso del procesador principal mientras un proceso esta esperando que se resuelva su solicitud de lectura o escritura?
- ¿Qué pasaría si el proceso tiene un error de lógica en su código, como por ejemplo un ciclo infinito?
- ¿Qué pasaría si el proceso está esperando por algún recurso de uso exclusivo que esta asignado a otro proceso?

Todas las respuestas a estas preguntas tienen algo en común, el paso del tiempo, en este transcurrir del tiempo se puede hacer algo productivo, algo no productivo o simplemente no hacer nada.

El microprocesador es un recurso del sistema muy valioso, entonces sería interesante mantenerlo trabajando la mayor parte del tiempo posible. El tiempo en el cual el microprocesador esta sin hacer nada, es tiempo ocioso, y debería aprovecharse en algo productivo, por ejemplo la ejecución de un proceso.

Para que esto suceda existen los algoritmos de planificación del procesador que junto con la técnica de la multiprogramación intentan que el procesador trabaje la mayor parte del tiempo, tratan de que los procesos se ejecuten de la manera más eficaz, eficiente y justa posible. De este modo, logran que todos los procesos, que compiten por el procesador, en algún momento terminen su tarea.

La planificación del procesador es una de las tareas más importantes e interesantes que realiza el SO, es por ello que entender bien este tema es



primordial. Para lograrlo le proponemos realizar la siguiente actividad de lectura.



Lectura requerida

Silberschatz A. y Galvin P.; Capítulo 5. Planificación de la CPU. Página 123. **En su: Sistemas Operativos**; 5ta Edición; Addison Wesley 1999.



Lectura Sugerida

Stallings W.; Capítulo 3. Descripción y control de procesos. Página 108. **En su: Sistemas Operativos –Aspectos Internos y principios de diseño-**; 5ta Edición; Prentice Hall 2007.

Silberschatz A. Galvin P. Gagne G.; Capítulo 5. Planificación de CPU. Página 153. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.

Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada-**; Mc Graw Hill.

Guía para la Lectura

1. ¿Qué tipos de instrucciones puede tener un proceso?
2. ¿Cuál es el ciclo de ráfagas de CPU y E/S?
3. ¿Qué es el planificador de CPU?
4. ¿Qué es la planificación expropiativa?
5. ¿Cuáles son los criterios de planificación?
6. Dé una explicación de cada uno de los algoritmos de planificación.
7. ¿Qué es la planificación de múltiples procesadores?
8. ¿Qué es la planificación en tiempo real?
9. ¿Cuáles son los modelos para evaluar los algoritmos de planificación?



Luego de realizar la lectura requerida, lo/a invitamos a continuar con la siguiente actividad. Le brindará la posibilidad de poner en práctica todo lo aprendido hasta el momento sobre la planificación de procesos.



Trabajo Práctico Sugerido

Trabajo Práctico N° 3: Planificación de procesos

Usted encontrará las consignas de este Trabajo Práctico en el Anexo que incluimos al final de este Orientador.

Comparta sus dudas e inquietudes con sus pares y con su tutor a través de los medios de comunicación disponibles en el Campus.

3. Señales, excepciones y temporizadores.

Las tareas que se realizan no siempre empiezan y terminan normalmente. En el transcurso del desarrollo de una tarea pueden ocurrir muchas cosas. Un proceso, como ya advertimos, puede tener un error: en la lógica del código, en una operación de división o un error en una operación de direccionamiento a memoria.

Antes de continuar con el desarrollo de este tema le sugerimos realizar la siguiente actividad.



Actividades para la facilitación de los aprendizajes

Reflexione sobre los siguientes interrogantes:

- ¿Qué acción se tomaría en los casos de error y más específicamente si se produce un error en una operación de direccionamiento a la memoria o un error en la operación de direccionamiento?
- ¿Quién tomaría la decisión de tomar una acción?

El proceso, durante la ejecución de una tarea, puede estar **programado sincrónicamente** para que ocurra algo determinado. Muchas veces, hay que prever y realizar éste tipo de acciones: cuando se tarda más del tiempo



establecido en resolver parte de una tarea, o cuando transcurre más tiempo en su realización.

Ahora, lo invito a pensar en las alarmas de los relojes de pulsera o en las alarmas programadas en nuestros teléfonos celulares que –si las programamos previamente- intentan decirnos algo transcurrido un tiempo.

Seguramente, estas preguntas tienen respuesta y esa respuesta es la que abordaremos en este punto de la unidad. Es por ello, que le proponemos la siguiente actividad de lectura.



Lectura requerida

Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada-; Mc Graw Hill.**

Guía para la Lectura

1. ¿Qué es una señal?
2. ¿Qué es una excepción?
3. ¿Qué diferencia hay entre una señal y una excepción?
4. ¿En qué casos utilizaría una señal?
5. ¿En qué casos utilizaría una excepción?
6. ¿Qué es un temporizador?
7. ¿En qué casos utilizaría un temporizador?

POSIX es una serie de normas, estándares y convenciones creadas por la comunidad de usuarios de Unix. Se utilizan con el fin de mejorar la compatibilidad entre diferentes sistemas operativos Unix y proporcionar una base para que las aplicaciones sean más portables entre distintos sistemas operativos.

4. Servicios POSIX para el manejo de procesos.

Los servicios POSIX para el manejo de procesos, son una ayuda para que los programadores puedan identificar, crear y terminar procesos. Si no existieran estos servicios, sería mucho más complicado programar todo lo relacionado con los procesos.

Utilizar los servicios POSIX, significa, pedirle al servicio que haga la tarea complicada por nosotros, llamamos al sistema de servicios para que haga lo que nosotros no podemos. Al igual que nosotros llamamos a un sistema de salud cuando necesitamos asistencia médica; cuando llamamos a un sistema de educación cuando necesitamos educarnos; cuando llamamos a un sistema de seguridad cuando nos sentimos amenazados por algún peligro.

Los procesos llaman a un sistema para que el sistema haga lo que el proceso no puede hacer porque no está autorizado o no está programado para ello.

Volvamos a hacernos algunas preguntas:



Actividades para la facilitación de los aprendizajes

- Un proceso que se está ejecutando quiere escribir en un dispositivo. ¿Es el proceso el que escribe en el dispositivo o llama al sistema para que escriba por él?
- Un proceso que se está ejecutando necesita transferir un dato por la red. ¿Es el proceso el que envía el dato por la red o llama al sistema para que lo envíe por él?
- Un proceso que se está ejecutando necesita memoria dinámica. ¿Es el proceso el que se asigna la memoria o llama al sistema para que le asigne la memoria?
- Un proceso que se está ejecutando necesita crear un proceso. ¿Es el proceso el que crea el proceso o llama al sistema para que lo asista en esta difícil tarea?

En la respuesta a todas estas preguntas aparece un concepto importante, que es **"llamada al sistema"**, y no es más que eso, llamar al sistema para que haga algo por nosotros, es como llamar a los bomberos para que apaguen un incendio porque ellos están preparados para hacerlo.

En este caso particular llamamos a los servicios **POSIX**, para que ayuden a los procesos de usuarios a realizar determinadas tareas sobre procesos, por ejemplo: Identificación de procesos, definir el entorno de los procesos, creación de procesos, terminación de procesos.

Le proponemos seguir investigando sobre el concepto de "llamada al sistema" a partir de la siguiente actividad de lectura.



Lectura requerida

Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada–**; Mc Graw Hill.

Guía para la Lectura

- ¿Qué significa POSIX?
- ¿Qué diferencia existe entre una llamada al sistema y una función de usuario?



- ¿Qué significa proceso huérfano y proceso zombie?
 - ¿Cómo podemos evitar la generación de procesos huérfanos y zombies?
-
-



Lectura Sugerida

Silberschatz A. Galvin P.; Capítulo 4. Procesos. Página 89. **En su: Sistemas Operativos**; 5ta Edición; Addison Wesley 1999.

Silberschatz A. Galvin P. Gagne G.; Capítulo 3. Procesos. Página 81. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.

Consideramos que usted a esta altura se encuentra en condiciones de realizar la siguiente actividad.



Trabajo Práctico Requerido

Trabajo Práctico N°4: Programación de procesos

5. Procesos livianos o hilos.

¡Qué difícil es compartir! cuanto esfuerzo le cuesta compartir sus recursos a las personas, parece que en este caso también los procesos se parecen al ser humano.

Recordemos la teoría del autor Silberschatz que dice "*Un proceso se define por los **recursos** que usa y por el **lugar** donde se ejecuta...*". En este concepto encontramos dos puntos importantes y críticos en el momento de compartir recursos y de comunicarnos y sincronizarnos con otros procesos, los puntos son **recursos** y **lugar**.

Quién administra la comunicación, la sincronización entre procesos y el hecho de compartir recursos entre procesos es nuestro amigo el SO, podemos decir, que es trabajo que el SO debe realizar para que los procesos hagan lo que tienen que hacer en el menor tiempo posible. Recuerde cuántas veces usted



estuvo frente a un computador esperando una respuesta que tardo varios segundos.

Los procesos livianos, también llamados hilos o threads, son una forma de unidades de ejecución que mejoran el rendimiento del sistema ya que la comunicación, la sincronización y poder compartir recursos entre ellos se hace de una manera más simple que demanda menos esfuerzo por parte del SO.



Lectura requerida

Silberschatz A. y Galvin P.; Capítulo 4. Procesos. Página 89. **En su: Sistemas Operativos**; 5ta Edición; Addison Wesley 1999.



Lectura Sugerida

Silberschatz A. Galvin P. Gagne G.; Capítulo 4. Threads. Página 127. **En su: Fundamentos de Sistemas Operativos**; 7ma Edición; Mc Graw Hill 2007.

Stallings W.; Capítulo 4. Hilos, SMP y micronúcleos. Página 158. **En su: Sistemas Operativos –Aspectos Internos y principios de diseño-**; 5ta Edición; Prentice Hall 2007. Capítulo 4.

Carretero Pérez J. De Miguel Anasagasti P. García Carballeira F. Pérez Costoya F.; Capítulo 3. Procesos. Página 77. **En su: Sistemas Operativos –Una visión aplicada-**; Mc Graw Hill.

Guía para la Lectura

1. ¿A qué se llama proceso pesado?
2. ¿A qué se llama proceso liviano o hilo?
3. Diferencias entre procesos pesado e hilo.
4. Describa como el SO Solaris gestiona los hilos.
5. ¿Qué son los hilos de nivel de usuario?
6. ¿Qué son los hilos de nivel de núcleo?



Cierre de la unidad

Esta unidad nos introduce en un mundo lleno de dinamismo, de transformación y movimiento, hay algo que tiene que trabajar, que tiene que hacer las cosas, que tiene que utilizar el mundo físico, conocido por los informáticos como hardware, parte de ese mundo físico son recursos que utilizan los procesos que pertenecen al mundo lógico conocido por los informáticos como software.

Los procesos para realizar su trabajo necesitan el procesador, la unidad trabajamos desde el concepto de proceso hasta las distintas planificaciones que el SO puede utilizar para que muchos procesos puedan ejecutarse en forma concurrente y compartir el procesador de la mejor manera posible. También comprendimos algunos mecanismos básicos que los procesos utilizan para activar alarmas en el tiempo y recibir señales para realizar alguna tarea.

Por último, nos introducimos en una forma de procesos llamados hilos, hebras, threads o procesos livianos que hacen que la comunicación y la sincronización entre ellos sea más simple que en los procesos pesados.



Anexo



Trabajo práctico sugerido

Trabajo práctico Nº 2: **Introducción a los procesos**

Presentación

Este trabajo tiene el propósito de orientarlo/a para la comprensión de los temas desarrollados en la segunda unidad de esta asignatura.

Los procesos son parte esencial de los sistemas operativos, ya que ellos son los que realizan las tareas que permiten llegar a los objetivos de los usuarios del sistema, como ya lo dijimos en otra oportunidad son como los individuos en una sociedad.

Por todo esto tenemos que conocerlos muy bien, para ello presentamos este trabajo práctico que nos permitirá ir familiarizándonos con ellos.

Este trabajo intenta favorecerle el acceso a las siguientes metas de aprendizaje:

- Internalizar el concepto de proceso como único recurso del sistema para llevar adelante las tareas dentro de un sistema de computación.
- Entender los métodos con los cuales el sistema operativo administra los procesos.
- Conocer las estructuras de información utilizadas para administrar procesos.

Le presentamos a continuación, las consignas de trabajo:

Consignas

1. Nombre las principales tareas que efectúa el SO en relación a la administración de Procesos.
2. ¿Qué es un proceso y qué es un thread (hilo)? ¿En qué se diferencian?
3. ¿En qué estructura de datos guarda el SO datos referidos a un proceso?
4. Dibuje el diagrama de estados de procesos en un entorno de multiprogramación y enumere los eventos que causan la transición entre los estados.

5. Nombre los distintos planificadores e indique cuáles son sus funciones y donde se ubicarían dentro del diagrama de estados.

6. Considere el siguiente conjunto de procesos planificados con un algoritmo round-robin con un quantum de 1, ¿Cuánto tardan en terminar todos ellos?

Proceso	Llegada	Duración CPU
P1	2	8
P2	0	5
P3	1	4
P4	3	3

Al finalizar, compare su producción con la grilla que incluimos a continuación.



Grilla de Autocorrección N° 2: Introducción a los procesos

Orientaciones para la corrección



*Recuerde que estas son solo orientaciones para que usted pueda comenzar a desarrollar el trabajo práctico.
Usted puede ampliar cualquiera de estos conceptos utilizando bibliografía adecuada, imaginación y creatividad.*

1. Nombre las principales tareas que efectúa el SO en relación a la administración de Procesos.

Las principales tareas que efectúa el sistema operativo en relación a la administración de procesos son: crear y eliminar procesos tanto de usuario como del sistema, suspender y reanudar procesos, proveer mecanismos para la sincronización y comunicación entre los procesos.

2. ¿Qué es un proceso y qué es un thread (hilo)? ¿En qué se diferencian?

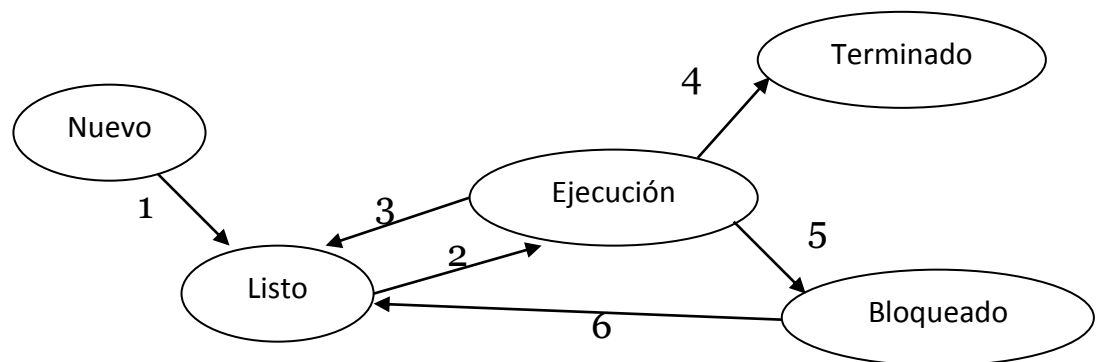
Un proceso es un programa que reside en memoria principal y al cual el sistema operativo le asigna un identificador y todos los recursos para que comience con su ejecución, a medida que se ejecuta se le pueden asignar más

recursos en función de sus necesidades, por lo tanto informalmente podemos decir que un proceso es un programa en ejecución. Los threads son los llamados procesos livianos que son hilos de ejecución de un proceso.

3. ¿En qué estructura de datos guarda el SO datos referidos a un proceso?

El sistema operativo guarda los datos referidos a un proceso en una estructura de datos que se llama Bloque de Control de Proceso (BCP). Los datos que guarda son: el Pid (identificador de proceso), el estado en que se encuentra el proceso, tiempo consumido del procesador, el contenido de los registros del procesador, el contador del programa, información relativo al uso de memoria, información relativa al uso de E/S e información relativa a los archivos utilizados por el proceso.

4. Dibuje el diagrama de estados de procesos en un entorno de multiprogramación y enumere los eventos que causan la transición entre los estados.



Cuando se crea un proceso, el sistema operativo le asigna su identificador de proceso y reside en área de almacenamiento de intercambio, encontrándose en estado Nuevo.

- 1) Una vez que el proceso está creado va a residir en memoria principal, listo y esperando para ejecutarse, formando parte de la cola de listos.
- 2) Cuando a un proceso se le asigna procesador pasa a estar en estado de Ejecución en cual puede ser interrumpido por distintas razones.
- 3) Puede ser que termine el tiempo asignado al proceso y el mismo no haya terminado o bien que llegue un proceso con mayor prioridad al que se está ejecutando, por lo que dicho proceso debe volver a la cola de listos esperando que nuevamente se le asigne procesador.
- 4) El proceso en ejecución finaliza y libera todos sus recursos asignados, así pasa al estado Terminado.
- 5) Mientras el proceso está en ejecución el mismo puede solicitar una E/S, lo cual hace que pase a la cola de dispositivo (ya que cada

dispositivo tiene su propia cola de espera) quedando en estado Bloqueado de entrada o salida.

- 6) Una vez que se atendió la solicitud de E/S nuevamente, el proceso, pasa a la cola de listos, esperando que se le asigne procesador.

5. Nombre los distintos planificadores e indique cuáles son sus funciones y donde se ubicarían dentro del diagrama de estados.

Los planificadores de proceso son:

- Planificador a largo plazo: Es el que determina qué procesos pasan del estado Nuevo a Listo.
- Planificador a corto plazo: Es el que determina qué procesos pasan del estado Listo a Ejecución.
- Planificador a medio plazo: Es el que determina que procesos del estado Listo a Bloqueado pasan al estado Swap.

6. Considere el siguiente conjunto de procesos planificados con un algoritmo round-robin con un quantum de 1, ¿Cuánto tardan en terminar todos ellos?

Proceso	Llegada	Duración CPU
P1	2	8
P2	0	5
P3	1	4
P4	3	3

Proceso/Tiempo	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
P2	E		E				E				E				E					
P3		E			E				E				E							
P1				E				E				E				E	E	E	E	E
P4						E				E				E						

En 20 unidades de tiempo todos los procesos habrán terminado.



Si tiene dudas con respecto a la adecuación de la tarea realizada proceda de la siguiente forma:

1. Compare lo realizado con los resultados propuestos en esta Grilla para la Autocorrección.
2. Identifique las diferencias y, si las hubiere, intente explicarlas. Verifique si es realmente un error de lógica, de forma de resolver o simplemente un problema de interpretación del ejercicio.
3. Si no encuentra explicación consulte a su tutor/a las dudas e inquietudes, expresándolas con claridad.



Trabajo práctico sugerido

Trabajo práctico Nº 3: **Planificación del procesador**

Presentación

Este trabajo tiene el propósito de fortalecer las definiciones, conceptos, procedimientos y funciones de los procesos que planifican y administran el microprocesador o CPU, para que los procesos de los usuarios puedan compartir el uso del mismo.

Estos ejercicios le ayudarán a entender cómo funcionan, qué hacen y cómo hacen estos algoritmos para llevar adelante la planificación del procesador.

Este trabajo intenta favorecerle el acceso a las siguientes metas de aprendizaje:

- Entender los mecanismos que permiten compartir el procesador entre los distintos procesos que se ejecutan en el sistema de computación.
- Reforzar los conceptos de la planificación del procesador.
- Conocer como el sistema operativo administra el recurso procesador principal.

Le presentamos a continuación, las consignas de trabajo:

Consignas

1. Nombre los distintos planificadores e indique cuáles son sus funciones y donde se ubicarían dentro del diagrama de estados.
2. Si estamos en un entorno operativo con un planificador de CPU a corto plazo basado en el algoritmo SJF (Próximo trabajo más corto primero). Existe la posibilidad de que el proceso que se está ejecutando (estado running), pueda ser desalojado de la CPU aunque no haya terminado su ráfaga de procesador.
3. ¿Qué algoritmo de planificación será más conveniente para optimizar el rendimiento de la CPU en un sistema que sólo tiene procesos en los cuales no hay entrada/salida?
4. ¿Cuál de las siguientes políticas de planificación es más adecuada para un sistema de tiempo compartido?



- a) SJF
 - b) Round-Robin
 - c) Prioridades
 - d) FIFO
5. En un sistema que usa un algoritmo de planificación de procesos round-robin con un quantum de tiempo de 100 ms. En este sistema ejecutan dos procesos. El primero no realiza operaciones de E/S y el segundo solicita una operación de E/S cada 50 ms. ¿Cuál será el porcentaje de uso de la CPU?
6. Considere el siguiente conjunto de procesos planificados con un algoritmo round-robin con un quantum de 1, ¿Cuánto tardan en terminar todos ellos?

Proceso	Llegada	Duración CPU
P1	2	8
P2	0	5
P3	1	4
P4	3	3

7. Se tienen los siguientes trabajos a ejecutar:

Proceso	Llegada	Duración CPU	Prioridad
P1	0	8	2
P2	1	5	4
P3	3	2	1
P4	4	7	3

La prioridad más alta es la de valor 1, se pide:
Escribir un diagrama que ilustre la ejecución de estos trabajos usando:

- 1) Planificación de prioridades no expulsiva.
 - 2) Planificación de prioridades expulsiva.
 - 3) FIFO.
8. Considere el siguiente conjunto de procesos con su correspondiente tiempo de llegada y su duración.

Proceso	Llegada	Duración CPU
P1	1	8
P2	5	1
P3	2	6
P4	4	9
P5	3	4



Considerando los siguientes algoritmos:

- 1) SJF sin desalojo
- 2) SJF con desalojo

Indicar cuál es el algoritmo de planificación con menor tiempo medio de Retorno.



Si surgen dudas u obstáculos que dificultan el aprendizaje o la comprensión de los contenidos durante la resolución de estas consignas, por favor, comuníquese con su tutor. Preséntele con claridad sus consultas para que él pueda brindarle las orientaciones que le permitirán resolverlas.