

Entrega2 - Sistemas Operativos II

Delfina Martín - Ignacio Litmanovich - Agustín Díaz

29 de marzo de 2021

Ejercicio 1

Los dispositivos móviles a diferencia de las PC, carecen de memoria virtual, por ende el sistema operativo se ve obligado a limitar al número de procesos interactivos en ejecución.

Por otro lado, los sistemas operativos móviles deben estar pensados para entorno cambiantes. A diferencia de los servidores o computadoras de escritorio que tienen un evento constante (ajuste del procesador), y también a diferencia de las computadoras portátiles que deben adoptar diferentes perfiles dependiendo de si están conectados a la red eléctrica o de si su tapa está baja, los dispositivos móviles funcionan bajo una filosofía de siempre encendido: el dispositivo siempre tiene que estar encendido y pendiente del entorno: posible llamada telefónica entrante, constante reconexión a antenas de telefonía, cambio de luminosidad en pantalla debido a cambios de luminosidad circundante, activación del modo ahorro de energía, cambios en la orientación del dispositivo, etc.

- a) Falso. La mayoría de los sistemas operativos están basados en Unix y DOS que son sistemas operativos monolíticos.
- b) Verdadero. Un sistema híbrido combina características de sistemas monolíticos y de microkernel. Son mayormente monolíticos pero manejan procesos centrales a nivel usuario como lo hacen los microkernel. Un ejemplo de este tipo de proceso central es el sistema de archivos.
- c) Falso. Un SO con organización monolítica permite que un solo proceso privilegiado realice todas las tareas a nivel núcleo (y no a nivel usuario).
- d) Falso. Un sistema operativo organizado como un microkernel mantiene la mayor parte del SO en el espacio de usuario.
- e) Falso. El uso de la memoria virtual se hizo necesario en los sistemas multiprogramados dado que se necesita garantizar un tiempo de respuesta razonable.

La principal ventaja de un esquema que utiliza memoria virtual es que es posible correr programas que ocupan más espacio en memoria que lo que dispone la memoria principal. Y en el caso particular de sistemas multiprogramados una extensión virtual de la memoria es conveniente dado que se cuenta con más memoria lógica para los distintos programas en curso.

- f) Verdadero. En los sistemas multiprogramados cada proceso requiere distintas prioridades. Algunos criterios que permiten distinguir los niveles de prioridad incluyen: qué tan relevantes son para el funcionamiento de la computadora misma o qué tan alta es la carga de interactividad, entre otros.

Ejercicio 2

	Arquitectura predominante	Sistema operativo predominante
Equipos servidores	Intel	Derivados de Unix
Propósito general en empresas, oficinas y PC	Intel	Windows
Dispositivos móviles	ARM	Derivados de Unix

Tabla 1: Arquitecturas y sistemas operativos predominantes

Ejercicio 3

- a) Verdadero. Algunos ejemplos de esto son:
 - Varias instancias de un mismo programa pueden estar asociadas a diferentes procesos.
 - El mismo programa puede crear varios procesos al ser ejecutado.
- b) Falso. Cada proceso es generado por un único programa.
- c) Falso. En los años 50 se denominaba “tarea” a la ejecución de un determinado programa. O lo que es lo mismo decir, se denominaba tarea a la instancia de un programa y no al programa en sí.

Ejercicio 4

Los sistemas con un solo procesador logran manejar varios procesos al mismo tiempo a través de la concurrencia. Dan la ilusión del paralelismo a través de un tipo de señal llamada interrupción.

Los segmentos de múltiples programas se ejecutan de forma posiblemente intercalada “compartiendo” el tiempo de cómputo de la computadora. En conjunto con la señalización, también se deben implementar mecanismos que guarden los estados intermedios de cada proceso en el momento de su interrupción de forma de que éstos se reestablezcan cuando se reanude la ejecución.

Los cambios de contexto se pueden dar a través de un esquema colaborativo en donde cada proceso cede el paso a voluntad o a través de uno apropiativo en el que el reloj del sistema interrumpe forzosamente la ejecución del proceso y traslada el control al SO.