

1. **Investiga** como es el proceso de arranque de una computadora. Todo empieza cuando se oprime el botón de encendido de la computadora, el cual provoca que empiece el suministro de energía por parte de la fuente de poder hacia la placa madre. En la placa madre se encuentra un software de bajo nivel, llamado BIOS (Basic Input Output System) o actualmente reemplazado por UEFI (Unified Extensible Firmware Interface).

Este BIOS/UEFI es el que se carga inicialmente y será el encargado de llevar a cabo una revisión POST (Power-On Self-Test), la cual consiste en realizar una inspección de los diversos elementos de hardware conectados a la placa madre, como lo son los módulos de memoria RAM, CPU, monitor, discos duros, tarjeta gráfica, etc. Si todos los elementos de hardware funcionan se pasa al siguiente escenario, en el cual se detecta al disco primario de almacenamiento y se leen los primeros 512 bytes, que contienen un segmento llamado MBR (Master Boot Record), al cual empezará a leer un programa llamado boot loader (cargador de arranque).

El boot loader o también llamado boot manager, el cual examina más a detalle la computadora y determina diversas configuraciones como el sistema de archivos.

En máquinas con Linux, el boot loader se conoce como GRUB (GNU GRand Unified Boot Loader).

En caso de que existan varios SO, el boot loader ofrece un GUI para seleccionar el SO que se quiera inicializar.

Una vez determinado el SO a utilizar, el boot loader empezará a leer el kernel del SO que se encuentra en disco y, después de la lectura de la primera instrucción, el boot loader cede el control al SO.

Finalmente, el SO toma el control del hardware y realiza algunas configuraciones iniciales, como la creación de un stack, inicializar memoria virtual, organizar el planificador de tareas, organizar el sistema de archivos y otras tareas, dependiendo del SO. Ya que se finalizaron de realizar estas configuraciones iniciales, se puede empezar a visualizar contenido en algún dispositivo de salida como el monitor.

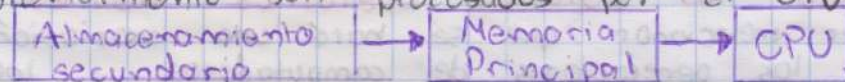


## 2.- Explica cómo funciona la memoria cache

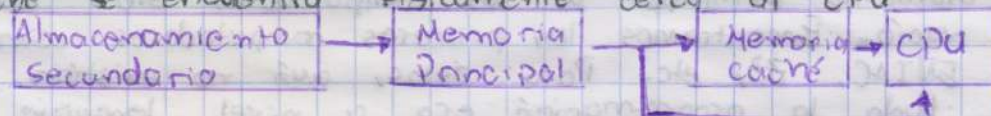
La memoria se divide en registros, memoria cache, memoria principal, discos magnéticos o discos de estado sólido, y cinta magnética o discos duros externos.

Por lo tanto, la memoria cache es una región de memoria que funciona como un intermediario entre la memoria principal y los registros de propósito especial.

Típicamente, los datos e instrucciones que llegan a realizar un procesador se traen desde almacenamiento secundario o terciario hacia la memoria principal y posteriormente son procesados por el CPU.



Lo que hace la memoria cache es almacenar datos e instrucciones de uso frecuente, para que el CPU pueda acceder con mayor rapidez a ellos, ya que la memoria cache se encuentra físicamente cerca al CPU.



Cabe resaltar que la memoria cache se encuentra dentro del CPU y cuenta con dos niveles: Nivel 1 (L1) y Nivel 2 (L2). La cache de nivel 1 usualmente es utilizada para guardar instrucciones, mientras que la de nivel 2 es utilizada para almacenar datos. La memoria L1 es de acceso inmediato (menor capacidad de memoria) y L2 tarda entre 1 a 2 ciclos de reloj en ser accesada (mayor capacidad de memoria).

El movimiento de datos o instrucciones a la memoria cache desde la memoria principal se realiza en bloques del mismo tamaño denominados como ranuras, en las cuales se encuentra dividida la memoria cache. Además, en estas ranuras se tiene una etiqueta que almacena la dirección desde donde se obtuvo el registro en memoria principal. Si la CPU solicita información adicional desde una ubicación almacenada previamente en cache, entonces se obtiene la información sin acceder a memoria principal, ya que el CPU siempre revisa primero a la memoria cache. En caso de no encontrar la información en cache, entonces sí se debe proceder a revisar la memoria principal.

Una vez que la memoria cache se llena, es decir, que ya no existen ranuras disponibles para nueva información, y se debe llevar a cache un nuevo bloque, entonces se utiliza un mecanismo de hardware para especificar la ranura menos usada recientemente y que sea reemplazado por el nuevo bloque.



Para determinar que el algoritmo de reemplazo de bloques está funcionando eficazmente, se calcula la tasa de aciertos de la caché

$$\text{Tasa de aciertos} = \frac{\text{Número de solicitudes en caché}}{\text{Número total de solicitudes}} \times 100$$

Siempre se busca que esta tasa sea mayor del 60% para determinar que la memoria caché es eficiente.

### 3- Leer el PDF "Historia de los sistemas operativos" del libro de Tannenbaum y hacer un resumen, ampliando la investigación para los sistemas operativos móviles.

Se empieza por mencionar que se hará una asociación cronológica entre las generaciones de computadores y los sistemas operativo, debido a la estrecha relación que comporten el hardware de los computadores y los sistemas operativos.

En la primera generación de computadores (1949-1955), donde el elemento principal de los computadores eran los tubos de vacío. Encontramos las primeras computadores como la Mark I, ENIAC, Z3, etc. Para estas, aún no existían sistemas operativos, toda la programación era a nivel lenguaje máquina o inclusive a nivel hardware.

Para la segunda generación de computadores (1955-1965), donde el elemento principal eran los transistores. Encontramos computadores como la IB4 1401 y la IBM 7094. Para estas computadores se tiene al ancestro de los sistemas operativos, el cual era un programa especial encargado de leer una cinta magnética en donde se encontraban un conjunto de instrucciones a ejecutar, para procesarlas y escribir los resultados en otra cinta magnética.

Los sistemas operativos o programas especiales típicos en esta época fueron FMS (Fortran Monitor System) e IBSYS, ambos pertenecientes a la computadora IBM 7094.

La tercera generación de computadores (1965-1980), donde el elemento principal eran los circuitos integrados. Encontramos a las computadores IBM 360, los cuales pretendían que fueran utilizados tanto para fines comerciales, como para fines científicos y de investigación. Por esto, se creó el sistema operativo OS/360, el cual debía trabajar en todos los modelos de la línea 360 de IBM.

Esto provocó grandes dificultades y bugs, además de un sistema operativo bastante complejo y extenso. A pesar de ello, dio la pauta para la utilización de diversas técnicas, entre las que destaca la multiprogramación. Anteriormente, la CPU se quedaba en estado de espera cuando un programa en ejecución debía esperar una salida a otra cinta. Para aprovechar este tiempo desperdiciado, se partieron a la memoria, para que se pudieran cargar varios



programas y, si durante la ejecución de uno había una pausar el CPU pudiera ejecutar algún otro programa.

Otro concepto o técnica que surgió fue el timesharing (tiempo compartido), en la cual se permitía que varios usuarios estuvieran conectados a una misma computadora, principalmente programadores que requerían depurar código, ya que no requería de mucha capacidad computacional. A los sistemas desarrollados bajo esta técnica se les llamó CTSS (Compatible Time Sharing System). Los desarrolladores de este sistema, conocidos como MULTICS (MULTIplexed Information and Computing System), diseñaron máquinas capaces de soportar hasta cientos de usuarios, sin embargo, no tuvo tanto éxito, debido a problemas de financiación y de enorme ambición. No obstante, tuvo una gran influencia en sistemas operativos posteriores como UNIX, FreeBSD, Linux, etc.

En esta generación también empezó la creación de las microcomputadoras. Aquí encontramos a la serie PDP de computadoras.

Fue la PDP-7 en donde Ken Thompson, desarrollador en MULTICS, creó UNIX, sistema operativo que adquirió gran popularidad, ya que era de código abierto. A partir de UNIX surge POSIX, que se definió según ciertos estándares de la IEEE, y también surge MINIX, para propósitos educativos.

De MINIX, surge Linux, desarrollado por Linus Torvalds y que también fue de código abierto.

Posteriormente llegó la cuarta generación de computadoras (1980 - Presente), donde el elemento principal eran los circuitos LSI (Large Scale Integration, traducido como larga escala de integración). Encontramos a las primeras computadoras personales, las cuales eran como microcomputadoras pero con costos más accesibles.

En esta generación se desarrolló el sistema operativo CP/M (Central Program for Microcomputers), desarrollado por Digital Research y que dominó el mercado por 5 años. (1977-1982)

En los 80's, IBM diseñó la computadora personal IBM PC y, necesitando de un sistema operativo, habla con Bill Gates para que les provea con uno. Bill Gates contrata a Tim Paterson, quien tenía un sistema operativo llamado DOS (Disk Operating System), el cual fue modificado y convertido en MS-DOS (Microsoft Disk Operating System). MS-DOS siguió mejorándose, incluyendo características más avanzadas, tomando varias del sistema operativo UNIX, que era de código libre.

Estos sistemas operativos no ofrecían interfaz gráfico, todo era a través de el ingreso de comandos desde el teclado. En 1960, Doug Engelbart inventa las interfaces gráficas, con ventanas, iconos, menús y mouse, sin embargo, fue Steve Jobs quien vio el potencial de esto y decidió



incorporarlo en sus computadores Apple Macintosh. Así es como surge Mac OS X, que también está basado en UNIX.

Con base en esta interfaz gráfica (GUI), MS-DOS se volvió Windows, no obstante, empezó a ser un sistema operativo con una interfaz gráfica más útil hasta 1995, con la llegada de Windows 95. Después de varias versiones de Windows, para Windows 2000, se le hizo una mejora que se llamó Windows XP, la cual estuvo vigente por 6 años.

A partir de este Windows, Microsoft desarrolló una línea de Windows para clientes y otro para servidores. Para la línea del cliente se desarrolló Windows Vista, Windows 7, Windows 8 y Windows 10 respectivamente.

Además de Windows y Mac, también se encuentran los derivados de UNIX, como Linux, FreeBSD, etc.

Para esta época, también se desarrollaron los sistemas operativos de red y distribuidos. Los primeros ofrecen la conexión con diversas computadoras en una misma red, pudiendo acceder unas a otras. Los sistemas operativos distribuidos se presentan al usuario como un sistema operativo tradicional, siendo que este se encuentra compuesto de varios procesadores. Estos sistemas operativos son más complejos, ya que requieren de algoritmos de sincronización y control de procesos más eficientes.

En la actualidad tenemos a la quinta generación de computadoras (1990 - Presente), en donde encontramos a los dispositivos móviles.

El primer dispositivo móvil apareció en 1970, y era conocido como "el ladrillo", por sus dimensiones. Fue a mediados de los 90's cuando apareció el primer smartphone, que combinaba un celular y un PDA (Personal Digital Assistant). Fue desarrollado por Nokia. Este contaba con el sistema operativo Symbian OS, al cual era ampliamente usado por empresas como Samsung, Sony Ericsson, Motorola, etc. Después aparecieron otros sistemas operativos como RIM's BlackBerry OS y Apple's iOS, los cuales le ganaron el mercado a Symbian OS, hasta la llegada de Android, el cual es un sistema operativo basado en UNIX lanzado en 2008 por Google, y que ofrece una adaptación bastante amigable con hardware, además de que es de código abierto y cuenta con una gran comunidad de desarrolladores de aplicaciones para Android.

Indagando un poco más en los tres principales sistemas operativos para dispositivos móviles tenemos que:



➤ **Android:** Su estructura consiste de 5 capas

|                         |                                |
|-------------------------|--------------------------------|
| Aplicaciones            |                                |
| Sistema de Aplicaciones |                                |
| Bibliotecas             | Tiempo de ejecución de Android |
| Linux Kernel            |                                |

Algunas características clave son:

- Almacena información mediante SQLite, en forma de tablas
- Tiene soporte de hardware para las aplicaciones, como: Sensor de proximidad, Cámara, GPS, Acelerómetro, etc.
- Soporta la ejecución de dos aplicaciones simultáneamente
- Soporte para MicroSD con Formatos FAT32, Ext3 o Ext4
- Permite la búsqueda por voz ofrecida por Google

➤ **iOS:** Su estructura consiste en 4 capas

|                        |
|------------------------|
| Aplicaciones           |
| Servicios principales  |
| Servicios de seguridad |
| Core OS                |

Algunas características clave son:

- Buen soporte y control para el almacenamiento en la nube
- Interfaz gráfica bastante llamativa y eficiente
- Sistema Operativo bastante estable y con pocos bugs, debido a los altos niveles de estandarización seguidos para el desarrollo de aplicaciones o actualizaciones
- Desarrollo fácil de aplicaciones a través de Swift.

➤ **Symbian OS:** Su estructura consiste en 8 capas

|                           |           |
|---------------------------|-----------|
| Aplicaciones              |           |
| UI Framework              | Java J2ME |
| Servicios de aplicaciones |           |
| Servicios del SO          |           |
| Servicios base            |           |
| Servicios de Kernel       |           |
| Interfaz de Hardware      |           |

Algunas características clave son:

- Permite la ejecución de procesos en tiempo real
- Presenta opciones de seguridad como encriptación total y el manejo de certificaciones.
- Sistema operativo completamente orientado a datos y basado en componentes.



- Permite la ejecución de varios tareas simultáneamente (multitarea)

En el 2012, el mercado se encontraba regido por Android en un 24.77%, iOS en un 8.05% y Symbian OS en 19.08%. Para 2019, Android controlaba el 84.02% del mercado, iOS el 13.33% y Symbian OS el 0.05%.

Actualmente también empiezan a surgir otros sistemas operativos para móviles como Sin OS, Kai OS, Ubuntu Touch, Tizen OS, Harmony OS, Lineage OS, etc. Sin embargo, el SO más utilizado actualmente es Android.

## Bibliografía

- Tanenbaum, A. & Bos, H. (2015). Modern operating systems: Global Edition. Pearson: [12.vireader.com/12eder?con=9781292061955#](https://vireader.com/12eder?con=9781292061955#)
- Ann, M. (2012). Sistemas Operativos. [VitalSource Bookshelf]. Obtenido de: <https://bookshelf.vitalsource.com/#/books/9786074817638>
- Goyathri, T., Madhusudhana, D., et al. (2020). A review of different handled devices operating systems. Journal of critical reviews. Obtenido de: [jcreview.com/fulltext/197-1591271436.pdf](http://jcreview.com/fulltext/197-1591271436.pdf)
- [beebom.com/android-alternative/](http://beebom.com/android-alternative/) consultado por última vez el 4/marzo/2021 a las 18:58.
- A DOS® Server in a Virtual Machine, The McGraw-Hill Companies. Consultado en: [vm.r454.com/redhat/OS-chOS.pdf](http://vm.r454.com/redhat/OS-chOS.pdf)
- Opar, P. (2018). The Process Behind a Computer Booting Up. Consultado en: [sires.psu.edu/pmo5086/Files/2018/11/Final-Draft-for-Technical-Definition-and-Description-1a8xtd7.pdf](http://sires.psu.edu/pmo5086/Files/2018/11/Final-Draft-for-Technical-Definition-and-Description-1a8xtd7.pdf)
- Montelius, J. (2018). These boots are made for walking. Consultado en: [people.kth.se/~johannor/ose/assignments/boot.pdf](http://people.kth.se/~johannor/ose/assignments/boot.pdf)