



# SISTEM OPERASI

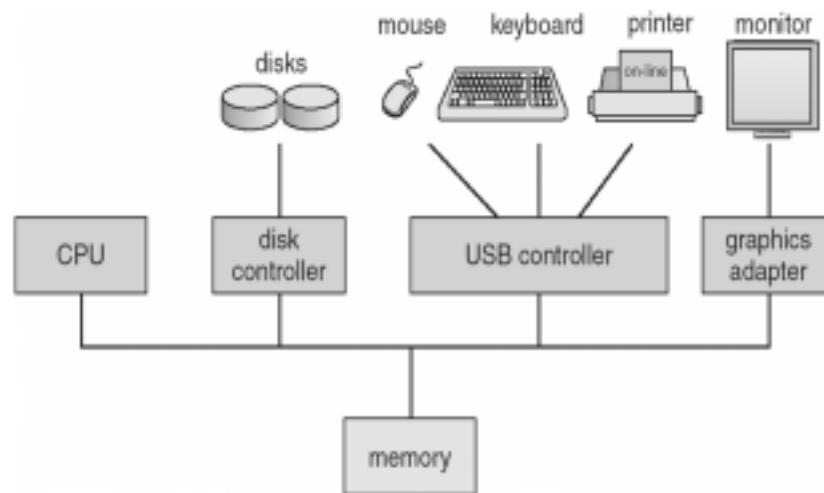
PAIK6302

**Departemen Ilmu Komputer/ Informatika  
Universitas Diponegoro  
Semester Gasal 2025/ 2026**

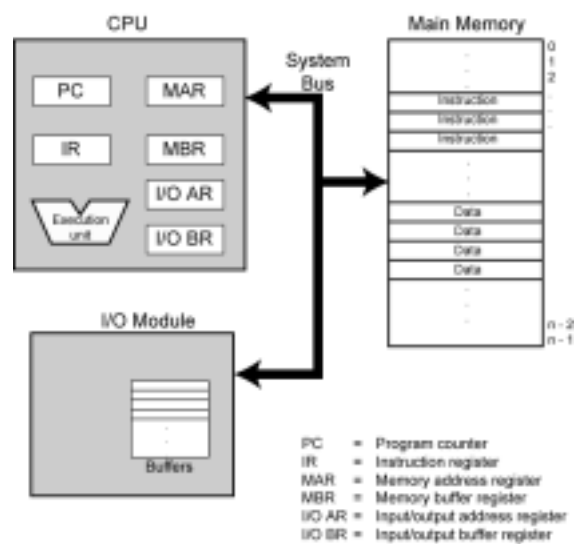
## Pertemuan ke 2

- Computer System
  1. Computer Hardware Organization
  2. Computer System Components
  3. What Operating Systems Do
  4. Views of an Operating System
  5. Definition of Operating System

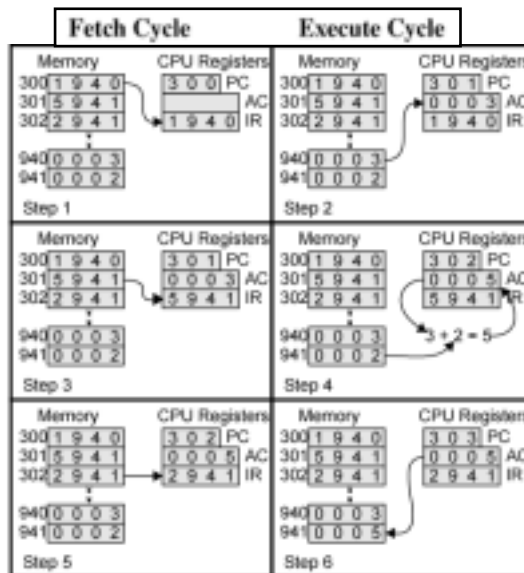
## 1. Computer Hardware Organization



## Top Level View



## Example of Program Execution



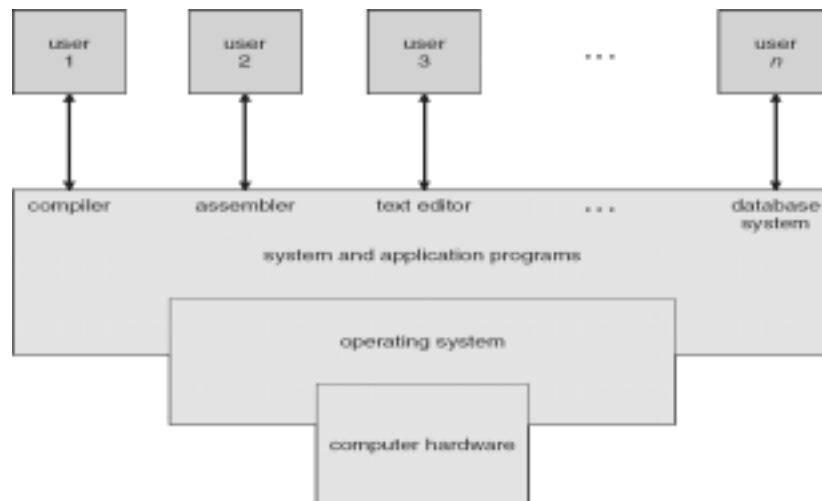
Informatika Undip

0001 = Muatkan AC dari memori  
 0010 = Simpan AC ke memori  
 0101 = Tambahkan AC dari memori

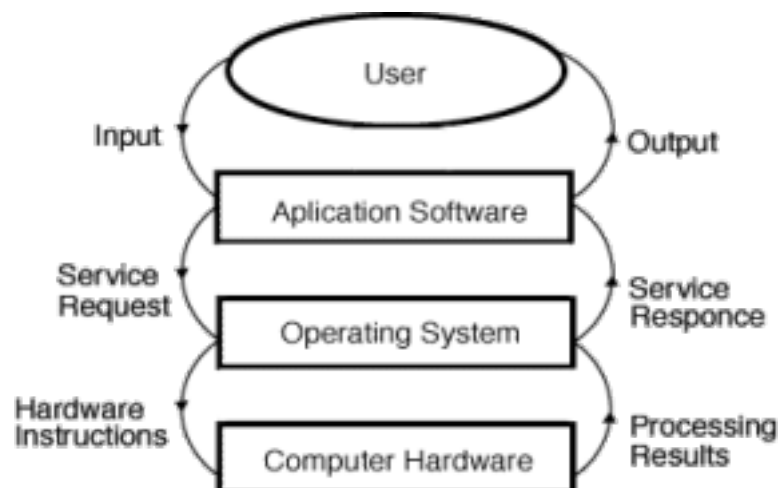
## 2. Computer System Components

- 1. Hardware** – provides basic computing resources (CPU, Memory, I/O devices, Communication).
- 2. Operating System** – controls and coordinates use of the hardware among various application programs for various users.
- 3. System & Application Programs** – ways in which the system resources are used to solve computing problems of the users (Word processors, Compilers, Web browsers, Database systems, Video games).
- 4. Users** – (People, Machines, other computers).

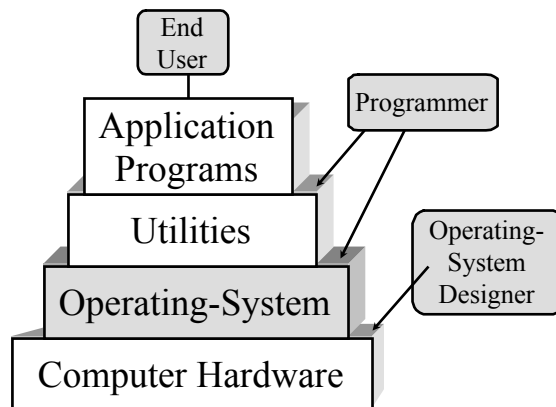
## Static View of System Components



## Dynamic View of System Components



## Layers of a Computer System



### 3. What Operating Systems Do?

- **Dari sudut pandang pengguna:** Mereka menginginkan kenyamanan, kemudahan penggunaan, dan kinerja yang baik. Mereka tidak terlalu peduli dengan pemanfaatan sumber daya.
- **Pada komputer yang digunakan bersama, seperti mainframe atau minicomputer:** Sistem operasi harus memastikan semua pengguna merasa puas dan dapat menggunakan komputer secara efektif.
- **Pengguna sistem khusus, seperti workstation:** Mereka memiliki sumber daya yang didedikasikan, tetapi sering menggunakan sumber daya bersama dari server.
- **Komputer genggam:** Biasanya memiliki sumber daya yang terbatas dan dioptimalkan untuk kegunaan dan daya tahan baterai.
- Beberapa komputer memiliki sedikit atau bahkan tidak memiliki antarmuka pengguna, seperti komputer tertanam dalam perangkat atau mobil.

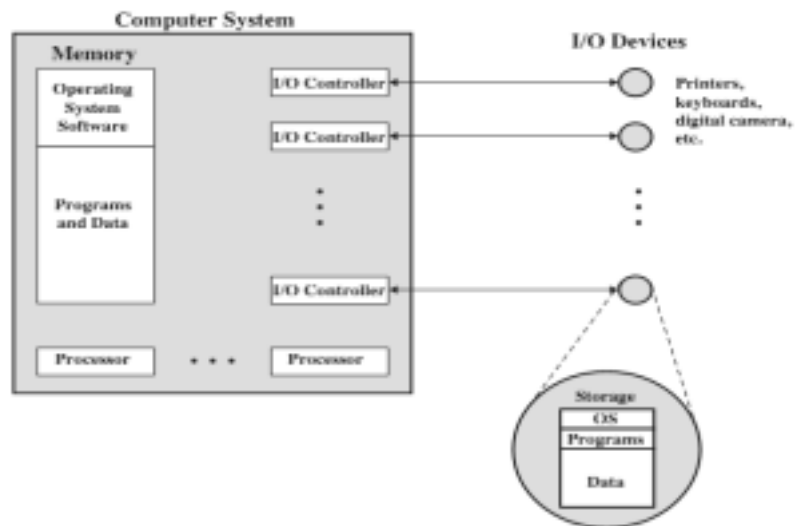
#### 4. Views of an Operating System

- Ada tiga pandangan klasik (dalam literatur):
  - **Resource Manager** – mengelola dan mengalokasikan sumber daya.
  - **Control Program** – mengontrol eksekusi program pengguna dan operasi perangkat I/O.
  - **Command Executer** – menyediakan lingkungan untuk menjalankan perintah pengguna.
- Namun, ada satu pandangan modern lainnya: Sistem Operasi sebagai Mesin Virtual.

#### 1. Resource Manager

- **Resource Manager:**
  - Manages and protects multiple computer resources: CPU, Processes, Internal/External memory, Tasks, Applications, Users, Communication channels, etc...
  - Handles and allocates resources to multiple users or multiple programs running at the same time and space (e.g., processor time, memory, I/O devices).
  - Decides between conflicting requests for efficient and fair resource use (e.g., maximize throughput, minimize response time).
- Sort of a bottom-up view.

## OS as a Resource Manager



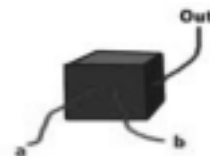
## Resource Manager oriented OS names

- DEC RSX – Resource Sharing eXecutive
- MIT Multics – MULTiplexed Information and Computing Services
- IBM MFT/MVT – Multiple Fixed/Variable Tasks
- IBM MVS – Multiple Virtual Storage
- DEC VMS – Virtual Memory System
- MVS TSO – Time Sharing Option
- CTSS – Compatible Time Sharing System
- IBM VM – Virtual machine

- **DEC RSX (Resource Sharing eXecutive)**: Sistem operasi yang dirancang oleh Digital Equipment Corporation (DEC) untuk mendukung pemakaian sumber daya secara bersama-sama, terutama dalam sistem real-time.
- **MIT Multics (MULTiplexed Information and Computing Services)**: Sistem operasi yang sangat berpengaruh, dikembangkan di MIT, yang memperkenalkan konsep-konsep seperti keamanan komputer dan pemakaian bersama informasi secara multi-plexing.
- **IBM MFT/MVT (Multiple Fixed/Variable Tasks)**: Sistem operasi dari IBM yang mendukung eksekusi beberapa tugas secara simultan dengan alokasi memori yang tetap (MFT) atau variabel (MVT).
- **IBM MVS (Multiple Virtual Storage)**: Sistem operasi IBM yang mendukung konsep penyimpanan virtual, memungkinkan banyak program untuk berjalan bersamaan dengan alokasi memori virtual yang terpisah.
- **DEC VMS (Virtual Memory System)**: Sistem operasi yang dikembangkan oleh Digital Equipment Corporation (DEC), dirancang untuk sistem komputer VAX, mendukung penggunaan memori virtual.
- **MVS TSO (Time Sharing Option)**: Ekstensi dari sistem operasi MVS dari IBM yang memungkinkan beberapa pengguna untuk berbagi waktu akses ke komputer secara bersamaan.
- **CTSS (Compatible Time Sharing System)**: Salah satu sistem operasi berbasis time-sharing pertama yang dikembangkan di MIT, memungkinkan banyak pengguna untuk berinteraksi dengan komputer secara bersamaan.
- **IBM VM (Virtual Machine)**: Sistem operasi dari IBM yang memungkinkan eksekusi beberapa mesin virtual di atas satu perangkat keras fisik, memberikan setiap mesin virtual lingkungan yang independen.

## 2. Control Program

- **Control Program:**
  - Manages all the components of a complex computer system in an integrated manner.
  - Controls the execution of user programs and I/O devices to prevent errors and improper use of computer resources.
  - Looks over and protects the computer: Monitor, Supervisor, Executive, Controller, Master, Coordinator ....
- Sort of a black box view.





### Control program oriented OS names

- Unisys MCP – Master Control Program
- DR CP/M – Control Program/Microcomputer
- IBM VM/CP – VM Control Program
- IBM AIX – Advanced Interactive eXecutive
- DEC RSX – Resource Sharing eXecutive

- **Unisys MCP (Master Control Program):** Sistem operasi yang digunakan pada komputer mainframe Unisys, salah satu sistem operasi tertua yang masih digunakan, dirancang untuk mendukung pemrosesan transaksi dan komputasi berskala besar dengan kinerja tinggi.
- **DR CP/M (Control Program/Microcomputer):** Sistem operasi yang dikembangkan oleh Digital Research untuk mikrokomputer, menjadi salah satu sistem operasi pertama yang populer di komputer pribadi pada akhir 1970-an dan awal 1980-an.
- **IBM VM/CP (VM Control Program):** Sistem operasi dari IBM yang mengelola mesin virtual pada sistem VM (Virtual Machine), memungkinkan beberapa lingkungan komputasi virtual berjalan secara simultan di atas perangkat keras fisik yang sama.
- **IBM AIX (Advanced Interactive eXecutive):** Sistem operasi Unix komersial yang dikembangkan oleh IBM, dirancang untuk workstation dan server, terkenal dengan stabilitas dan skalabilitasnya dalam lingkungan bisnis dan enterprise.
- **DEC RSX (Resource Sharing eXecutive):** Sistem operasi real-time yang dikembangkan oleh Digital Equipment Corporation (DEC), dirancang untuk mendukung pembagian sumber daya di antara beberapa pengguna atau proses, terutama dalam aplikasi industri dan sains.

### 3. Command Executer

- **Command Executer:**
  - Interfaces between the users and machine.
  - Supplies services/utilities to users.
  - Provides the users with a convenient CLI (Command Language Interface), also called a Shell (in UNIX), for entering the user commands.
- Sort of a top-down view.

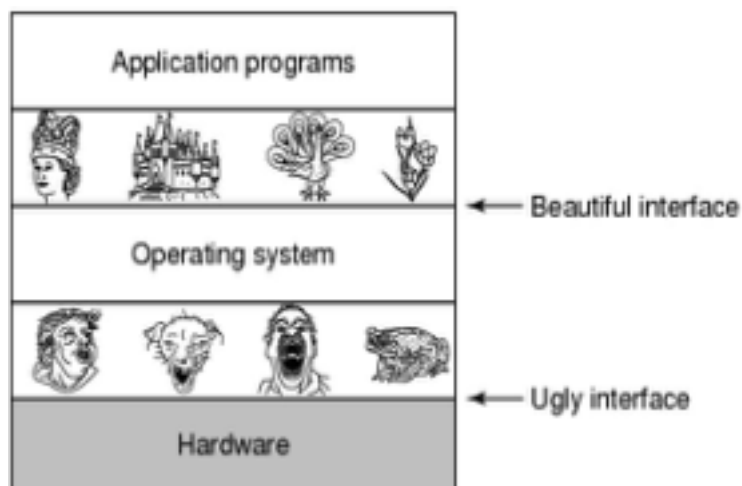
### Command Executer oriented OS names

- **IBM AIX (Advanced Interactive Executive):** AIX adalah sistem operasi berbasis Unix yang dikembangkan oleh IBM, terutama digunakan untuk workstation dan server. AIX dikenal karena keandalan, skalabilitas, dan kemampuannya untuk mendukung lingkungan komputasi yang kritis bagi bisnis dan enterprise.
- **IBM VM/CMS (Conversational Monitor System):** CMS adalah komponen dari sistem operasi VM (Virtual Machine) dari IBM. VM/CMS menyediakan antarmuka yang mudah digunakan bagi pengguna, memungkinkan mereka untuk menjalankan program dan berinteraksi dengan sistem dalam lingkungan mesin virtual. Ini sangat populer dalam lingkungan mainframe IBM untuk menjalankan aplikasi dan mengelola sumber daya komputasi secara efisien.

## Modern view: Virtual Machine (1)

- Operating System as a Virtual Machine:
  - An interface between the user and hardware that hides the details of the hardware (e.g., I/O).
  - Constructs higher-level (virtual) resources out of lower-level (physical) resources (e.g., files).
  - **Definition:** OS is a collection of software enhancements, executed on the bare hardware, culminating in a high-level virtual machine that serves as an advanced programming environment.
    - virtual machine = software enhancement = extended machine = abstract machine = layer = level = ring.

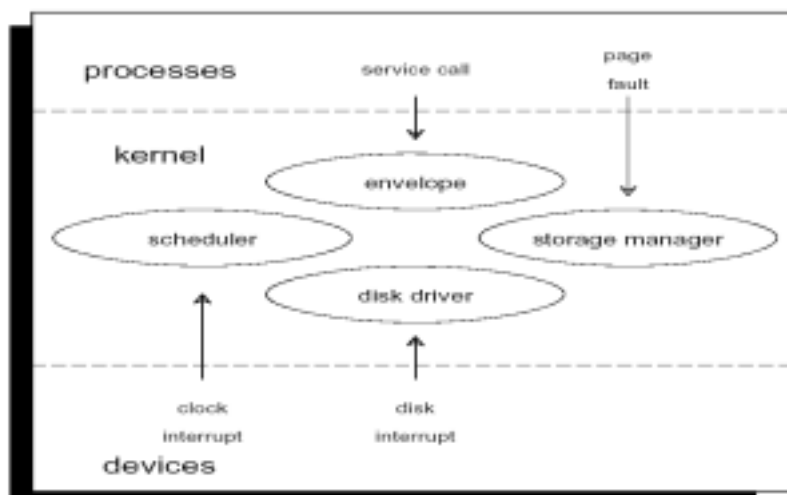
## Modern view: Virtual Machine (2)



## Definition of Operating System

- There is no universally accepted definition.
  - “Everything a vendor ships when you order an operating system” is good approximation but varies widely.
- “The one program running at all times on the computer” is the **Kernel**.
- Everything else is either a system program (ships with the operating system) or an application program.

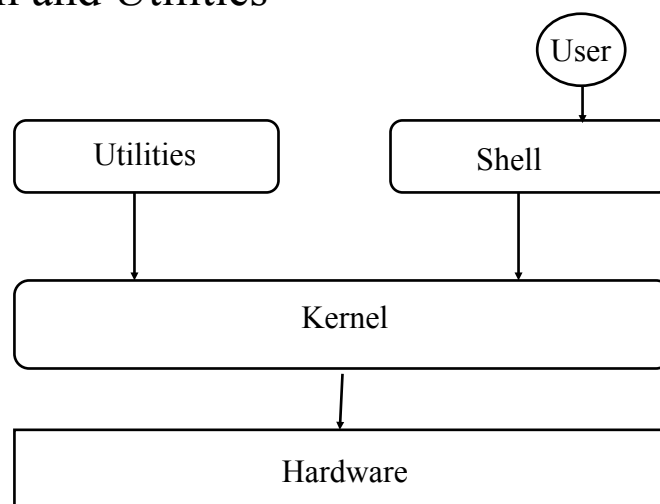
## One Kernel Point of View



### What is the OS/Kernel?

- Is the Operating System just the Kernel (not the utilities and application programs)?!
- The Command Line Interface (CLI) (or command layer/interpreter or shell) allows direct command entry by the user.
- The shell used to be in the kernel but now is a (first between equals) utility outside of it:
  - Easy to change/debug
  - Many of them (sh, bsh, csh, ksh, tcsh, wsh, bash)
  - Possible to switch between them (chsh)

### UNIX Shell and Utilities

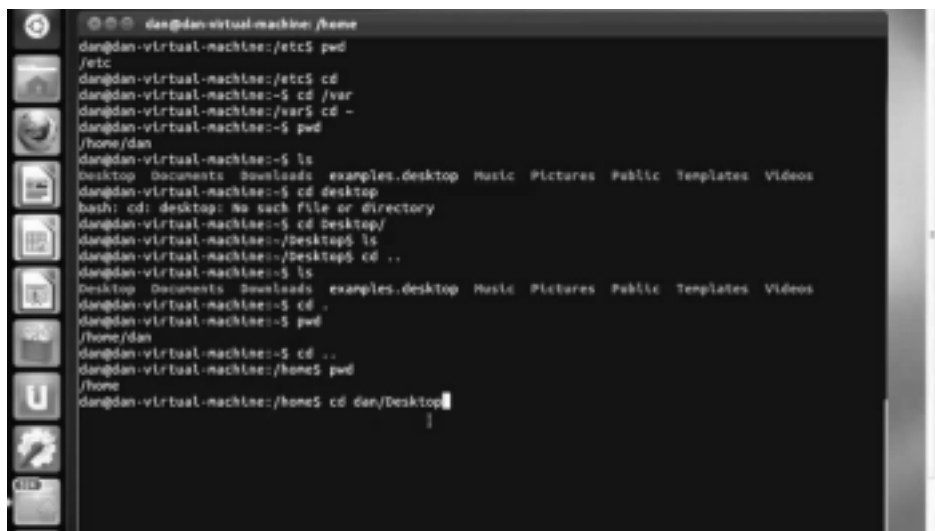


## CLI is the User OS Interface

CLI allows direct command entry:

- Sometimes implemented in kernel, sometimes by systems program.
- Sometimes multiple flavors implemented – shells.
- Primarily fetches a command from user and executes it.
- Sometimes commands built-in, sometimes just names of programs; if the latter, adding new features doesn't require shell modification.

## CLI, Terminal, Shell, Console



```
dan@dan-virtual-machine: /home
dan@dan-virtual-machine: /etc$ pwd
/etc
dan@dan-virtual-machine: /etc$ cd
dan@dan-virtual-machine: /var$ cd -
dan@dan-virtual-machine: /home$ pwd
/home/dan
dan@dan-virtual-machine: /home$ ls
Desktop Documents Downloads examples.desktop Music Pictures Public Templates Videos
dan@dan-virtual-machine: /home$ cd desktop
bash: cd: desktop: No such file or directory
dan@dan-virtual-machine: /home$ cd Desktop/
dan@dan-virtual-machine: /home/Desktop$ ls
Desktop Documents Downloads examples.desktop Music Pictures Public Templates Videos
dan@dan-virtual-machine: /home/Desktop$ cd ..
dan@dan-virtual-machine: /home$ ls
Desktop Documents Downloads examples.desktop Music Pictures Public Templates Videos
dan@dan-virtual-machine: /home$ cd .
dan@dan-virtual-machine: /home$ cd ..
dan@dan-virtual-machine: /home$ pwd
/home
dan@dan-virtual-machine: /home$ cd dan/Desktop
```

## A very simplified Shell

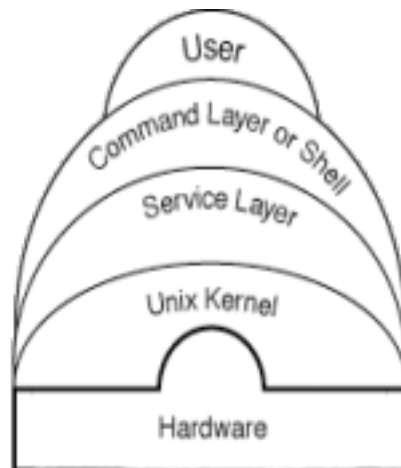
```
#define TRUE 1

while (TRUE) {                                /* repeat forever */
    type_prompt( );                          /* display prompt on the screen */
    read_command(command, parameters);       /* read input from terminal */

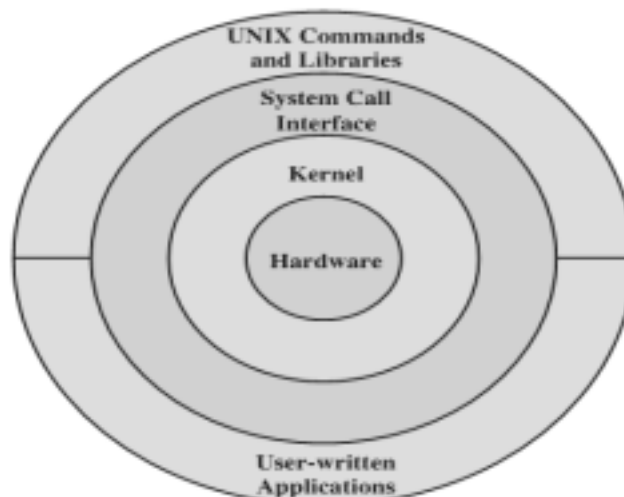
    if (fork() != 0) {                        /* fork off child process */
        /* Parent code. */
        waitpid(-1, &status, 0);           /* wait for child to exit */
    } else {
        /* Child code. */
        execve(command, parameters, 0);     /* execute command */
    }
}
```

- `#define TRUE 1`:
  - Ini mendefinisikan konstanta TRUE dengan nilai 1. Ini digunakan dalam loop untuk membuat loop berjalan terus menerus karena kondisinya selalu bernilai true.
- `while (TRUE) {`:
  - Ini adalah loop yang akan terus berjalan selamanya karena kondisinya (TRUE) selalu benar (selalu 1).
- `type_prompt();`:
  - Fungsi ini bertugas untuk menampilkan prompt di layar. Ini bisa jadi semacam tanda seperti \$ atau > yang muncul di terminal untuk memberi tahu pengguna bahwa sistem siap menerima perintah.
- `read_command(command, parameters);`:
  - Fungsi ini bertugas untuk membaca perintah dari terminal. Perintah dan parameter yang dimasukkan oleh pengguna akan disimpan dalam variabel `command` dan `parameters`.
- `if (fork() != 0) {`:
  - Fungsi `fork()` digunakan untuk membuat proses anak. Jika `fork()` mengembalikan nilai selain 0, itu berarti kode tersebut sedang dijalankan oleh proses induk.
  - `fork()` mengembalikan 0 di proses anak dan ID proses anak di proses induk.
  - Di dalam blok induk (if):
- `waitpid(-1, &status, 0);`:
  - Fungsi ini digunakan oleh proses induk untuk menunggu proses anak selesai. -1 berarti menunggu proses anak apa pun untuk selesai.
- Di dalam blok anak (else):
- `execve(command, parameters, 0);`:
  - Fungsi ini digunakan oleh proses anak untuk menjalankan perintah yang diberikan oleh pengguna. Fungsi ini menggantikan proses anak dengan program baru yang dijalankan perintah tersebut.

## UNIX System Layout

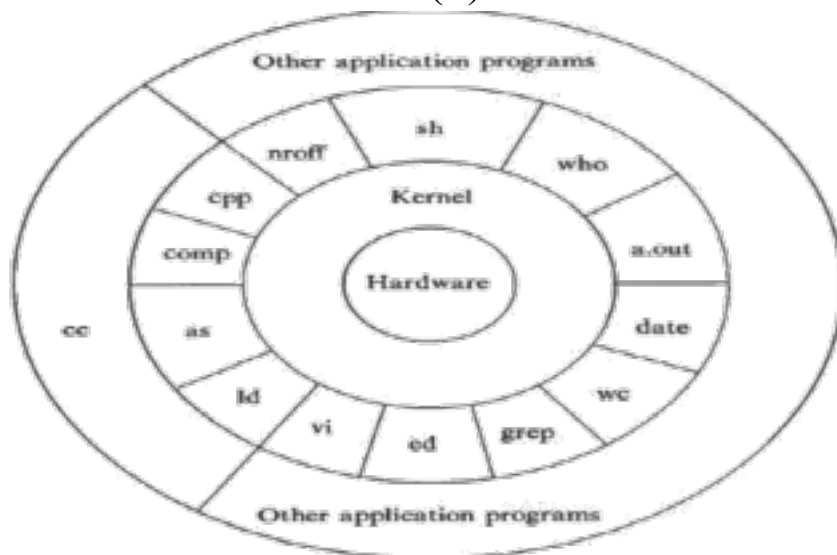


## General UNIX Architecture (1)





## General UNIX Architecture (2)



Soal :

1. Apa yang dimaksud sistem operasi?
2. Sebutkan tiga peran utama dari sistem operasi.
3. Apa fungsi kernel dalam sistem operasi?
4. Apa itu 'System Call Interface' dalam konteks sistem operasi?
5. Jelaskan peran 'Command Executer' dalam sistem operasi.