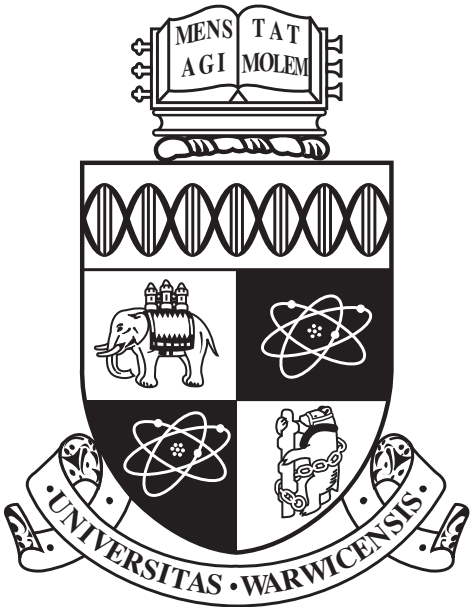
University of Warwick
Department of Computer Science

CS241

Operating Systems and Computer Networks



Cem Yilmaz
October 4, 2022

Contents

1 Introduction

1.1 What is an operating system?

There is no universally accepted definition, however, a broad definition would be as follows

Definition 1.1. Operating system

A software acting as an intermediary between the user of a device and the hardware of a device. Examples include, but are not limited to, PCs, laptops, phones, cars, airplanes, etc.

A computer system hardware resources include CPU, Memory, I/O devices and storage. The OS is responsible for allocating these resources to user processes and control their execution. Th e goal is to avoid failures and errors e.g., two programs writing in the same memory.

1.1.1 Program Execution

The system must be able to load a program into memory, execute that program and stop that program either normally or abnormally.

1.1.2 I/O Operations

While running, a program may require to perform I/O. For example, display something to the terminal, take input from keyboard.
For efficiency and protection, users cannot control devices directly. Instead, the OS controls I/O devices through device drivers and interrupts. More efficient because users do not need to write codes to perform I/O and more secure because multiple programs cannot access the same I/O simultaneously.

1.1.3 File System Management

A program needs to read and write files and directories, perform additional operations such as copy and move, and finally, these functions may be subject to permission management. The OS system provides system calls to achieve these.

1.1.4 Error handling

The operating system needs to be detecting and correcting errors constantly. Errors may occur in hardware e.g., failure of cpu, memory and I/O devices or user programs e.g., arithmetic overflow, illegal memory access.

1.1.5 Resource Allocation

When there are multiple users or multiple jobs running at the same time, resources must be allocated to each of them. In scheduling CPU jobs, the OS must consider the speed of the processor, number of processors available, jobs to be executed etc.

1.1.6 Accounting

It is necessary to keep track of which processes are running and how much resources they consume. Such information may be used for system administration or billing purposes e.g., clouds. The operating system gathers this information through control blocks.

1.1.7 Protection and Security

When separate processes execute concurrently they should not be able to interfere with other processes or with the operating system itself. Security of the system from outside attacks is just as important.

1.2 What is the Kernel?

Definition 1.2. Kernel

Kernel is the core of an operating system. It is loaded into the main memory at system startup and is a process that runs at all times in the computer. There are certain functions which only a kernel can perform, e.g., memory management, process scheduling and file handling.

1.2.1 Kernel Space vs User Space

Kernel space is part of the memory where kernel executes. User space is the section of memory where user processes run. Kernel space is kept protected from user space. Kernel space can be accessed via user processes through system calls. These calls perform services like I/O operations or process creation.

1.2.2 System Calls

When a user process requires a service from the kernel e.g., reading from a file or reading to a file, it invokes a system call. System calls are required since user processes cannot perform certain privileged operations. System calls are low-level functions provided by the operating system. They provide a consistent interface for common operations. E.g., read(), write()

1.2.3 Dual mode operation

A mechanism to distinguish between OS operations and user operations. Hardware operates in two modes: user mode and kernel mode. Mode bit (0 or 1) indicates a kernel or a user mode. Some instructions are designated as privileged, only executable in kernel mode. System calls by a user asking the OS to perform some function changes from user mode to kernel mode. Return from a system call recalls.