

Operating System Course Report - First Half of the Semester

A class

October 10, 2024

Contents

1	Introduction	3
2	Course Overview	3
2.1	Objectives	3
2.2	Course Structure	3
3	Topics Covered	4
3.1	Basic Concepts and Components of Computer Systems	4
3.1.1	Jenis-Jenis Komputer	4
3.1.2	Hardware	4
3.1.3	Software	4
3.1.4	Interaksi Antara <i>Brainware</i> , <i>Hardware</i> , dan <i>Software</i>	8
3.2	System Performance and Metrics	8
3.3	System Architecture of Computer Systems	9
3.4	Process Description and Control	9
3.5	Scheduling Algorithms	9
3.6	Process Creation and Termination	9
3.7	Introduction to Threads	10
3.8	File Systems	10
3.9	Input and Output Management	11
3.10	Deadlock Introduction and Prevention	11
3.11	User Interface Management	11
3.12	Virtualization in Operating Systems	11
4	Assignments and Practical Work	12
4.1	Assignment 1: Process Scheduling	12
4.1.1	Group 1	12
4.2	Assignment 2: Deadlock Handling	12
4.3	Assignment 3: Multithreading and Amdahl's Law	12
4.4	Assignment 4: Simple Command-Line Interface (CLI) for User Interface Management	13
4.5	Assignment 5: File System Access	13
5	Conclusion	13

1 Introduction

This report summarizes the topics covered during the first half of the Operating System course. It includes theoretical concepts, practical implementations, and assignments. The course focuses on the fundamentals of operating systems, including system architecture, process management, CPU scheduling, and deadlock handling.

2 Course Overview

2.1 Objectives

The main objectives of this course are:

- To understand the basic components and architecture of a computer system.
- To learn process management, scheduling, and inter-process communication.
- To explore file systems, input/output management, and virtualization.
- To study the prevention and handling of deadlocks in operating systems.

2.2 Course Structure

The course is divided into two halves. This report focuses on the first half, which covers:

- Basic Concepts and Components of Computer Systems
- System Performance and Metrics
- System Architecture of Computer Systems
- Process Description and Control
- Scheduling Algorithms
- Process Creation and Termination

- Introduction to Threads
- File Systems
- Input and Output Management
- Deadlock Introduction and Prevention
- User Interface Management
- Virtualization in Operating Systems

3 Topics Covered

3.1 Basic Concepts and Components of Computer Systems

This section explains the fundamental components that make up a computer system, including the CPU, memory, storage, and input/output devices.

3.1.1 Jenis-Jenis Komputer

3.1.2 Hardware

3.1.3 Software

Software adalah sekumpulan instruksi kode dan data-data tertentu yang berfungsi untuk menjalankan sebuah program atau perintah yang ada didalam komputer. Program yang akan dijalankan tentunya akan memerlukan sebuah *hardware* atau perangkat keras yang dimiliki agar perangkat lunak berjalan optimal.

Dalam pembuatannya, *software* dibuat oleh seorang programmer atau pengembang perangkat lunak. Perangkat lunak ini dibuat dengan menggunakan bahasa pemrograman tertentu yang kemudian akan diubah menjadi kode mesin agar dapat dijalankan oleh komputer.

Software dirancang untuk memfasilitaskan pekerjaan manusia. Contohnya seperti menghitung, membuat dokumen, mengedit gambar, dan sebagainya. Selain itu, dengan *software* kamu juga bisa melakukan pengeditan video, pembuatan desain, permainan *game*, dan masih banyak lagi.

- **Ciri-ciri *Software***

Namun meski secara jelas *software* memiliki kegunaan dalam menjalankan sebuah perintah atau fitur tertentu dalam komputer, *software* juga memiliki ciri spesifik. Berikut beberapa ciri-ciri jika program tersebut adalah sebuah *software* :

- Penunjang *hardware*

Software bertanggung jawab untuk mengelola perangkat keras pada komputer. Perangkat keras juga terkadang membutuhkan pemrograman yang dihasilkan dari perangkat lunak untuk bekerja secara sempurna. Komputer atau gadget tentunya akan berfungsi dengan baik jika pemrograman dilakukan sesuai dengan fungsi dan perintah.

- Memiliki sumber terbuka atau berdifat komersil

Dalam menginstal sebuah perangkat lunak terkadang developer dapat dengan mudah mengunduhnya secara gratis. Hal ini disebabkan memang ada beberapa perusahaan yang dengan sengaja menerapkan *Open source* pada *software* mereka. Namun juga ada beberapa pengembang yang menerapkan sistem berbayar pada *software* yang dikembangkannya.

- Membutuhkan File *Installer*

Dalam melakukan pemasangan sebuah *software* kedalam sebuah komputer tentunya akan membutuhkan yang dinamakan dengan File *Installer*.

- Rentan Terhadap Virus

Baik sebelum mengunduh maupun setelah mengunduh *software*, komputer Anda akan jauh lebih rentan terkena virus saat menjalankannya. Hal tersebut juga tidak dapat dipungkiri perangkat lunak yang akan digunakan mungkin sudah terdapat virus didalamnya.

- **Jenis-Jenis *Software***

Secara umum, perangkat lunak terbagi menjadi 2, yaitu :

1. *Software System*

Software System Merupakan *software* yang paling dasar dalam sebuah komputer, bertugas untuk mengelola semua *hardware* komputer dan menyediakan layanan umum bagi *software* lain. Contoh *Software System* yang paling umum, yaitu :

- Sistem Operasi (*Operating System/OS*)
Software yang bertugas untuk mengelola semua sumber daya komputer, seperti *hardware*, *software*, dan data. Sistem Operasi juga bertanggung jawab untuk menjalankan program-program aplikasi dan memberikan antarmuka pengguna agar pengguna dapat berinteraksi dengan komputer. Contohnya, *Windows*, *macOS*, *Linux*, dll.
- *Driver*
Software yang bertugas untuk menghubungkan perangkat keras dengan sistem operasi. *Driver* memungkinkan sistem operasi untuk mengenali dan berkomunikasi dengan perangkat keras, seperti *printer*, *scanner*, dan kartu grafis.
- *Utility Software*
Software yang dirancang untuk membantu pengguna dalam mengelola, mengoptimalkan, dan memperbaiki sistem komputer. *Utility Software* dapat digunakan untuk membersihkan file sampah, mempercepat kinerja komputer, dan memperbaiki kesalahan sistem. Contohnya, *antivirus*, *disk cleaner*, dan *system optimizer*.

2. *Software Aplikasi*

Software Aplikasi merupakan *software* yang sudah banyak digunakan di suatu bidang tertentu. Jenis perangkat lunak ini dibuat untuk mengerjakan suatu fungsi atau perintah khusus yang terdapat didalam komputer. Contoh *Software Aplikasi* yang paling umum, yaitu :

- *Software Productivity*
Software yang dirancang untuk membantu pengguna dalam menyelesaikan tugas sehari-hari, seperti membuat dokumen, mengelola data, dan membuat presentasi. Contohnya, *Microsoft Office*, *Google Docs*, dan *LibreOffice*.
- *Software Multimedia*
Software yang dirancang untuk mengakses file multimedia,

seperti gambar, video, dan audio. *Software* Multimedia dapat digunakan untuk melihat foto, menonton video, dan mendengarkan musik. Contohnya, *Spotify*, *Photos* dan *Media Player*.

- *Software* Kreativitas

Software yang dirancang untuk membantu pengguna dalam membuat konten kreatif, seperti desain grafis, animasi, dan pengembangan web. Contohnya, *Adobe Photoshop*, *Blender*, dan *WordPress*.

- *Software* Game

Software yang dirancang untuk memberikan hiburan kepada pengguna dalam bentuk permainan. *Software* Game dapat digunakan untuk bermain *game* tunggal atau bermain *game* online dengan pemain lain. Contohnya, *Minecraft*, *Fortnite*, dan *League of Legends*.

- *Software* Web Browser

Software yang dirancang untuk mengakses dan menjelajahi di internet. *Software* Web Browser dapat digunakan untuk mencari informasi, berbelanja *online*, dan berkomunikasi dengan orang lain. Contohnya, *Google Chrome*, *Mozilla Firefox*, dan *Microsoft Edge*.

- **Jenis Software Berdasarkan Distribusinya**

Software atau perangkat lunak juga memiliki beberapa pendistribusiannya. Diketahui ada 7 jenis *software* yang memiliki klasifikasi fungsi dan pekerjaan yang berbeda. Berikut penjelasan mengenai 7 tipenya:

1. ***Adware***

Adware secara khusus diciptakan untuk membagikan beberapa iklan secara luas kedalam komputer terutama ketika sedang berada didalam internet. Jenis perangkat lunak ini juga bisa menghasilkan pendapatan melalui metode *pay per click* (PPC).

2. ***Firmware***

Perangkat lunak ini berguna sebagai sistem operasi yang terdapat pada komputer khususnya pada bagian perangkat keras. Penyimpanan perangkat lunak ini bersifat hanya baca sehingga tidak perlu lagi melakukan modifikasi maupun pengembangan lebih lanjut meskipun terjadi masalah pada fungsinya.

3. ***Freeware***

Merupakan salah satu jenis *software* yang tidak memiliki batas waktu tertentu. Akan tetapi, kebanyakan *software* jenis ini memiliki fitur yang tidak begitu lengkap sehingga penggunaannya pun kurang maksimal.

4. ***Malware***

Merupakan salah satu jenis dari *software* yang dianggap berbahaya dan bisa merusak apabila disalahgunakan penggunaannya. Program perangkat lunak ini secara spesifik digunakan untuk merusak komputer atau mencuri data-data yang ada didalam komputer.

5. ***Spyware***

Perangkat lunak ini juga sama berbahaya seperti pada sebelumnya. Akan tetapi perangkat lunak jenis ini lebih diciptakan untuk memata-matai terhadap kegiatan yang ada di dalam komputer.

6. ***Open Source***

Merupakan salah satu jenis *software* yang memiliki kode sumber yang terbuka. Dengan kata lain, pengguna dapat mengakses kode sumber dari *software* tersebut dan melakukan modifikasi sesuai dengan kebutuhan.

7. ***Shareware***

Shareware adalah perangkat lunak yang dapat di gunakan secara gratis. *Software* ini pada umumnya biasa digunakan untuk melakukan pengujian atau demonstrasi dengan fungsionalitas dan pengujian rentang waktu tertentu.

3.1.4 Interaksi Antara *Brainware*, *Hardware*, dan *Software*

3.2 System Performance and Metrics

This section introduces various system performance metrics used to measure the efficiency of a computer system, including throughput, response time, and utilization.

3.3 System Architecture of Computer Systems

Describes the architecture of modern computer systems, focusing on the interaction between *hardware* and the operating system.

3.4 Process Description and Control

Processes are a central concept in operating systems. This section covers:

- Process states and state transitions
- Process control block (PCB)
- Context switching

3.5 Scheduling Algorithms

This section covers:

- First-Come, First-Served (FCFS)
- Shortest Job Next (SJN)
- Round Robin (RR)

It explains how these algorithms are used to allocate CPU time to processes.

3.6 Process Creation and Termination

Details how processes are created and terminated by the operating system, including:

- Process spawning
- Process termination conditions

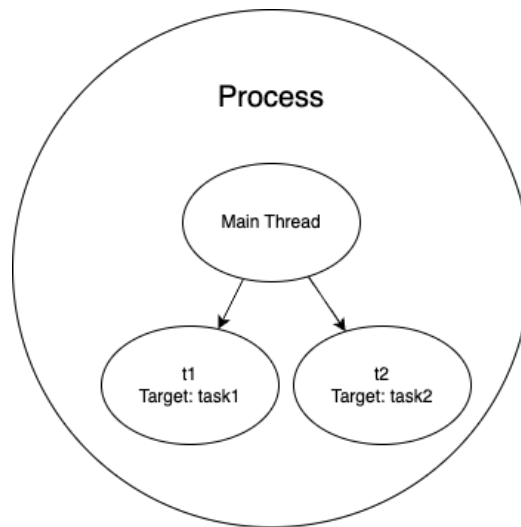


Figure 1: Ini adalah gambar contoh dari multithreading.

3.7 Introduction to Threads

This section introduces the concept of threads and their relation to processes, covering:

- Single-threaded vs. multi-threaded processes
- Benefits of multithreading

Seperti yang terlihat pada Gambar 1, inilah cara menambahkan gambar dengan keterangan.

3.8 File Systems

File systems provide a way for the operating system to store, retrieve, and manage data. This section explains:

- File system structure
- File access methods
- Directory management

3.9 Input and Output Management

Input and output management is key for handling the interaction between the system and external devices. This section includes:

- Device drivers
- I/O scheduling

3.10 Deadlock Introduction and Prevention

Explores the concept of deadlocks and methods for preventing them:

- Deadlock conditions
- Deadlock prevention techniques

3.11 User Interface Management

This section discusses the role of the operating system in managing the user interface. Topics covered include:

- Graphical User Interface (GUI)
- Command-Line Interface (CLI)
- Interaction between the user and the operating system

3.12 Virtualization in Operating Systems

Virtualization allows multiple operating systems to run concurrently on a single physical machine. This section explores:

- Concept of virtualization
- Hypervisors and their types
- Benefits of virtualization in modern computing

4 Assignments and Practical Work

4.1 Assignment 1: Process Scheduling

Students were tasked with implementing various process scheduling algorithms (e.g., FCFS, SJN, and RR) and comparing their performance under different conditions.

4.1.1 Group 1

```
class Process:
def __init__(self, pid, arrival_time, burst_time):
    self.pid = pid
    self.arrival_time = arrival_time
    self.burst_time = burst_time
    self.completion_time = 0
    self.turnaround_time = 0
    self.waiting_time = 0
```

Header 1	Header 2	Header 3
Row 1, Column 1	Row 1, Column 2	Row 1, Column 3
Row 2, Column 1	Row 2, Column 2	Row 2, Column 3

Table 1: Your table caption

4.2 Assignment 2: Deadlock Handling

In this assignment, students were asked to simulate different deadlock scenarios and explore various prevention methods.

4.3 Assignment 3: Multithreading and Amdahl's Law

This assignment involved designing a multithreading scenario to solve a computationally intensive problem. Students then applied **Amdahl's Law** to calculate the theoretical speedup of the program as the number of threads increased.

4.4 Assignment 4: Simple Command-Line Interface (CLI) for User Interface Management

Students were tasked with creating a simple **CLI** for user interface management. The CLI should support basic commands such as file manipulation (creating, listing, and deleting files), process management, and system status reporting.

4.5 Assignment 5: File System Access

In this assignment, students implemented file system access routines, including:

- File creation and deletion
- Reading from and writing to files
- Navigating directories and managing file permissions

5 Conclusion

The first half of the course introduced core operating system concepts, including process management, scheduling, multithreading, and file system access. These topics provided a foundation for more advanced topics to be covered in the second half of the course.