This assignment is to familiarize you with OS past and present.

You are required to do your own research through the Internet or books. The marks will be given on your analyzing or research effort. Copy/pa5te or plagiarism is not allowed.

Part One (26 marks)

a) Make a timeline of OS from 1960 to the present according to the OS major family or category below (12 marks):

1. IBM

(i) system360(1964)

- Before the Syste360 computers were very different from each other. Each machine needed its own operating system and software and programs written for one computer could not run on anothe This made upgrading very expensive because companies had to rewrite all their software when they bought new hardware. IBM changed this with the System360 It introduced a family of computers that all shared the same architecture and could run the same software. This meant programs were portable across different models. Businesses could upgrade hardware without starting from scratch This idea of compatibility became a foundation of modern computin

(ii)system370(1972)

The System370 was developed as an improvement of the System360. Its most important feature was **virtual memory** which allowed programs to use more memory than what was physically installed by using disk space as an extension of RAM Another major change was the move from magnetic core memory to semiconductor memory, which increased performance and reliability. IBM also kept backward compatibilityso older System/360 programs continued to work. In short System/360 focused on compatibilitywhile Syste370 expanded system capabilities through virtualization.

(iii) PC DOS(1981)

PCDOS was released with the IBM Personal Computer in 1981 It played a major role in standardizing personal computing For the first time many different machines could run the same operating system and software.PC-DOS introduced the FAT file system which organized data on disks and is still used today in devices like USB drives IBM allowed Microsoft to license the same operating system as MS DOS to other manufacturers This decision helped create the IBm compatible PC market and allowed Microsoft to become the dominant software company.

(iv)system390(1990)

With System390 IBM focused on reliability and availability Large organizations such as banks and airlines needed systems that could run continuously without downtime

System390 supported clustering allowing multiple mainframes to work together as one system. If one machine failed, others could continue handling the workload It also introduced ESCON which used fiber optic connections to increase data transfer speed. These features made mainframes extremely dependable for critical business operations

(v)The zSeries and z/OS (2000)

In 2000 IBM renamed its mainframes to the zSeries where z refers to zero downtime. The system moved from 31bit to 64bit addressing allowing access to much larger amounts of memory.IBM also made a surprising decision to support linux on mainframes. This allowed companies to run modern applications alongside older legacy systems on the same hardwareThis flexibility helped keep mainframes relevant in the modern computing world.

(vi)z13 to z16(2010s – 2023)

During this period, IBM focused heavily on security and data processing. One major feature was **pervasive encryption**, which allows all data to be encrypted without affecting performance. This became especially important for financial institutions.The z16 introduced the Telum processor which includes builtin AI acceleration. This allows rea time analysis of large numbers of transactions such as detecting credit card fraud while transactions are happening.

(vii) z/OS 3.1 2024 – present day)

- We are now in the era of Hybrid Cloud The current version zOS 3.1 is designed to work seamlessly with the cloud (like Amazon or Google) while keeping the core data on the mainframe.
- Quantum Safe Cryptography IBM is now preparing for the future when Quantum Computers might be able to break today's passwords The newest systems use post quantum cryptographic algorithms designed to resist known quantum attacks

2. UNIX

(i) The origin(1969–1971)

Before UNIX, Bell Lab ,MIT and General Electric were working on a large project called **Multics**. The goal of Multics was ambitious but over time it became too complex expensive and difficult to manage In 1969 Bell Labs decided to withdraw from the project.After this **Ken Thompson** and **Dennis Ritchie** at Bell Labs wanted a simpler system they could experiment with Using the experience they gained from Multics they created a new operating system called **UNIX**.

UNIX was designed to be **small simple and modular**. Instead of one massive progra, it consisted of many small tools each designed to perform a single task well. Initially UNIX was written in **assembly language** which made it hardware dependent. In 1973 it was rewritten in the **C programming language** allowing it to be easily moved to different types of hardware This portability became one of UNIX's most important features.

(ii) (1970s–1980s)

- In the late 1970s and early 1980 The ecosystem bifurcated into two major competing camps, historically referred to as the Unix Wars and the innovations created during this rivalry are the reason our modern internet and servers work today.

(a) BSD (Berkeley Software Distribution)

The Academic Innovator: Developed at the University of California, Berkeley, this version was driven by students and researchers who wanted to add new features to the original Bell Labs code.

- TCP/IP Networking BSD was the first OS to integrate the protocols that run the internet. If you use a socket to connect to a website today, you are using technology born in BSD.
- Virtual Memory Leap BSD introduced the first robust implementation of virtual memory for UNIX, allowing it to run much larger programs on the hardware of the time
- BSD became the foundation for FreeBSD and a major component of Apple's macOS

(b) System V (AT&T)

The Corporate Standard: After a legal change allowed AT&T to sell software they released System V as

the official commercial version of Unix.
- Inter-Process Communication (IPC). System V introduced Shared Memory and Semaphores, which allowed different programs to talk to each other and share data instantly. This is essential for modern databases.
- Standardization. AT& T focused on making Unix stable and predictable for big business. They created the System V Interface Definition (SVID) to ensure software would work the same way every time.
- System V influenced almost all high-end commercial Unix versions including IBM's AIX, HP UX, and Sun Solaris.

(iii) The Workstation Era (1980s–1990s): The Rise of AIX and Solaris

- This was a time when engineers, scientists, and animators required high-performance computing capabilities. IBM and Sun Microsystems dominated this era by creating Proprietary Unix versions that were tied to their own high-performance hardware.

1. IBM AIX (Advanced Interactive eXecutive)

IBM introduced **AIX** in 1986 to run on its RISC based systems
One major contribution was the **Logical Volume Manager (LVM)**, which allowed disk storage to be resized or expanded without shutting down the system.
AIX also introduced the **Journaled File System (JFS)** which improved reliability by tracking changes before writing them to disk.
AIX is still used today in environments where stability and uptime are critical

2. Sun Solaris

- If AIX was the market leader of the corporate office, Solaris was the market leader of the **Internet**. Sun Microsystems used the slogan The Network is the computer, and Solaris proved it.
- The main achievement of this version was The Network File System (NFS). Solaris invented the standard way for computers to share files over a network as if they were on a local drive. This is why we can have Shared Drives in offices today.
- Later in its life, Solaris introduced DTrace (for seeing exactly what the OS is doing in real time) and ZFS (a file system that can handle massive amounts of data with self-healing capabilities). Solaris powered the early servers of companies like eBay and Yahoo and laid out the technical groundwork for modern **Cloud Computing.**

(iv) **Modern era (2001–Present): macOS and Beyond**

UNIX continues to influence modern operating systems One of the most important examples is **macOS**. In the 1990s, Steve Jobs founded **NeXT** which developed an operating system called **NeXTSTEP** based on BSD UNIX When Apple acquired NeXT in 1996 this technology became the foundation for Apple's future operating systems.
In 2001, Apple released **Mac OS X** combining a UNIX based core (Darwin) with a user friendly graphical interface This brought UNIX reliability and networking capabilities to everyday users.
Today UNIX concepts remain central to **macOS** Linux systems and the servers that support much of the global internet.

3. WINDOWS

(i) **From DOS to Desktop (1985-1994)**

- In this first stage, Windows was not a full operating system. It was a GUI Shell that ran on top of MS-DOS.
- This meant if DOS crashed, Windows crashed with it. It was Microsoft's response to the Apple Macintosh attempting to bring ease of use to the IBM PC market.

(a). Windows 1.0 (1985): Introduced the mouse and tiled windows (they couldn't overlap) A key limitation was that windows could not overlap they had to sit side by side. It was heavily talked about for being slow and relying too much on the mouse which few people owned at the time.

(b). Windows 2.0 (1987): Allowed overlapping windows for the first time and introduced keyboard shortcuts that we still use today It also introduced the Control Panel, creating a central place for system settings.
(c). Windows 3.x (1990-1992): The first massive success It introduced the File Manager and supported TrueType fonts, making PCs viable for professional design and office work.
- The significance of this Era was that it provided a bridge It allowed users to keep using their old reliable DOS programs while slowly learning how to use a mouse and icons By the time this era ended with Windows 3.11 the world was fully prepared to leave the command line behind.
-A key technical breakthrough of this era was **virtual memory** Windows 3.1 could use a portion of the hard drive as temporary RAM, allowing users to run heavy applications (like Excel and Word) simultaneously without running out of memory. By the time this era ended with Windows 3.11 the world was fully prepared to leave the command line behind

(ii) **Mainstream Dominance (1995–2001)**

- This period marked Windows' achievement of market dominance
(a) Windows 95 (1995): The biggest turning point in Windows history. It introduced the Start Menu, the Taskbar, Plug and Play, and improved stability However it was a hybrid 16-bit/32-bit system that still relied on DOS code leading to frequent crashes known as the Blue Screen of Death (BSOD)
(b)Windows 98 (1998): Integrated the web browser directly into the OS a move that sparked a massive antitrust lawsuit but successfully brought the internet to the masses. It also added native support for USB killing off floppy disks and serial ports.
(c) Windows XP (2001): This was the Grand Unification. Before XP, Microsoft had two separate lines. Windows 9x for homes (unstable but compatible) and Windows NT for business (stable but strict). Combined the consumer look of Win95 with the pro stability of the Windows NT kernel. It became one of the most beloved and longest lasting OS versions in history.

(iii) **The Modern/Service Era (2009–Present)**

- marks the transition of Windows from a product you buy in a box to a service that lives in the cloud and updates automatically. In this era, Microsoft stopped trying to reinvent the interface every few years and focused instead on security mobility and cloud integration.
(a) Windows 7 (2009): Refined the OS after the poorly received Vista, Vista had introduced a new driver model (WDDM) that caused incompatibility, but Windows 7 reaped the benefits of that security work. It focused on speed and introduced Snap for organizing windows side-by-side.
(b)Windows 8 (2012): A radical, controversial change that tried to force a tablet style touch interface on desktop users by removing the Start Menu. While the interface failed, the under-the-hood engine was faster and more secure than 7
(c) Windows 10/11 (2015-Present): Shifted to Windows as a Service, meaning the OS receives constant online updates instead of big new releases every few year Windows 11 added a centered Start menu and native support for Android Apps.

4. OPEN SOURCE.

(i) **The GNU Project (1983)**

- before open source operating systems became common most software was **closed source** meaning users could not see or modify the code. **Richard Stallman** a programmer at MIT became frustrated when he was unable to fix a printer because the manufacturer refused to share the source code
- In responseStallman started the **GNU Project** in 1983. His goal was to create a **Unix-like operating system** that was completely free. Here "free" did not mean free of cost but free in terms of user freedom the freedom to study modify and share software.
- Stallman also created the **GNU General Public License (GPL)** This license introduced the idea of

copyleft which allowed anyone to modify the software but required that any changes must also be shared openly. This ensured that the software would always remain free.
By the early 1990s the GNU Project had produced most of the tools needed for an operating system such as compilers text editors and shells. However their own kernel called **HURD** proved difficult to complete and was delayed for many years.

(ii) the Linux Kernel (1991)

- In 1991 a Finnish university student named **Linus Torvalds** began working on his own kernel as a personal project He shared the code online and invited others to contribute.
- Unlike traditional commercial software development, Linux was developed openly on the internet. Programmers from around the world were able to review the code suggest improvements,] and fix bugs.
- Developers soon realized that by combining the **GNU tools** with the **Linux kernel** they had a complete operating system. This system is technically called **GNU/Linux** although it is more commonly referred to simply as LinuX.

(iii) The Distro Era (1993–2004)

Since anyone could take Linux and change it different Distributions or Distros started to appear, like
(a) Debian (1993): Focused on strict adherence to free software principles and stability. It invented the APT package manager which solved the headache of installing software by automatically finding and downloading the necessary parts or a program to run
(b) Red Hat (1994): The first company to prove you could make billions from free software. They didn't sell the Linux code (which was free) they sold Support and Services Big banks and governments paid Red Hat to guarantee that their Linux servers wouldn't crash. This legitimized Linux for the corporate world.
(c) Ubuntu (2004): Created by Mark Shuttleworth Ubuntu took the complex Debian code and optimized the user interface for non technical consumers It made Linux easy enough for an average person to install and use without being a command line expert, doing for Linux what Windows 95 did for the PC

(iv) The Modern Era (2010–Present)

Today opensource software forms the backbone of modern computing infrastructure. While many users run Windows or macOS on personal computers **Linux dominates servers and largeScale systems**.
Most internet services rely on opensource software often using the **LAMP stack** (Linux, Apache, MySQL, PHP/Python) Major platforms such as Google, Facebook, and Amazon Web Services operate primarily on Linux-based system
Linux is also used in nearly all of the world's top supercomputers and serves as the foundation for **Android smartphones** and **ChromeOS** Its popularity is largely due to its reliability, flexibility, and lack of licensing costs.

5. Network OS

- A Network Operating System (NOS) is a specialized OS designed to manage data, users, groups, and security across a network. In the history of computing this was a critical shift where computers stopped being islands and started working together.
- here is a brief history of NOS;

(i) Novell NetWare (1983–1995)

- Before Windows had networking, **Novell NetWare** was the premier of the office. It was a dedicated system, meaning the server did nothing but handle network traffic.
-NDS (NetWare Directory Services) This was the first giant phonebook for networks. It allowed an administrator to manage thousands of users and printers across different cities from one single login.

- Because it didn't have a heavy interface (it was text-based and specialized), it was incredibly fast at moving files, much faster than early Windows. It also established the standard Client Server model that every office in the world uses today.

(ii) Windows NT & Server (1993–2000s)

Microsoft changed the game by building networking directly into the Windows interface people already knew.

- Active Directory (AD) Launched with Windows 2000 AD replaced Novell's system. It became the global standard for how businesses control who can log into which computer and access which folder.
- The All in one)UI Unlike NetWare which was hard to learn, Windows Server looked just like the desktop. This made it cheap for companies to find staff who knew how to run it.

(iii) Cisco IOS (1980s–Present)

- While Novell and Microsoft managed users, Cisco IOS was created to manage the **hardware** of the internet itself.
- This OS doesn't run on a PC it runs on Routers and Switches It is responsible for making sure a packet of data from your house reaches a server in Russia or any other place in milliseconds.
- The key point is the reliability, it is also a Command Line system designed to run for years without a single reboot.

6. Mobile OS

- Te Mobile OS Era represents the most rapid shift in computing history It took the power of a desktop computer and shrunk it into a pocket sized device, moving from keyboard and mouse to touch and gestures.

Here is its brief history

(i) The Pre-Smartphone Era (Late 90s–2006)

- Before the iPhone, mobile devices were mostly for business. They were called PDAs (Personal Digital Assistants) or Feature Phones.

(a)Symbian OS (Nokia): the Market leader. It was the first OS to successfully bring apps and cameras to mobile phones.

(b) BlackBerry OS: Famous for its physical keyboard and Push Email. It was the head of the mobile world, essential for every businessman and government official.

(c)PalmOS: Popularized the use of the Stylus and touch sensitive screens for handheld organizers.

(ii) iOS (2007–Present)

- When Steve Jobs introduced the iPhone in 2007, he famously said it was three products in one. It changed the OS world by focusing on experience over raw technical specs.

-the main two achievement were

(a Multi-touch UI iOS replaced the stylus and tiny buttons with natural finger gestures (swipe to unlock, pinch to zoom).

(b) The App Store (2008) This was the real game changer. By allowing third-party developers to sell apps, Apple turned the phone into a platform that could be a bank map, a game console, or a library.

- iOS set the standard for Closed Ecosystems, where the hardware and software are perfectly tuned to work together.

(iii) Android (2008–Present)

- Seeing Apple's success, Google launched Android as an Open Source alternative.
- the main two achievements were;

(a) Hardware flexibility; Because Android is open source (based on the Linux kernel), any company could use it. This led to a massive variety of phones, from $100 budget devices to $2,000 folding phones.

(b)Customization; Unlike the strict walls of iOS, Android allowed users to change almost everything about their OS from the home screen layout to the file system access.

- Android became the Windows of Mobil eventually powering roughly 70% of the world's smartphones.

- Now here is a table which brings together all the OS timelines I discussed above;

Decade	IBM (The Foundation)	UNIX/Open Source (The Brains)	Windows (The Desktop)	Network & Mobile (The Connectors)
1960s	System360: Portable software is born.	UNIX Birth: Modular lean design at Bell Labs.	*Not yet introduced*	*Not yet introduced*
1970s	System370: Virtual Memory breakthrough.	The Split: BSD (Internet tech) vs System V (Corporate).	*Not yet introduced*	*Not yet introduced*
1980s	PC-DOS: The start of the PC era.	GNU Project: The Free Software movement begins.	Windows 1.0/2.0: The first mouse-driven shells.	NetWare: Offices start sharing files; Cisco IOS routes the web.
1990s	System390: Fiber optics & High Availability.	Linux Kernel: Torvalds completes the Open Source engine.	Win 95/98: Start Menu and USB go mainstream.	Windows NT: Networking is built directly into the OS.
2000s	zSeries: 64-bit power and Linux integration.	macOS: Apple moves to Unix; Ubuntu makes Linux easy.	Windows XP/7: Stability meets the consumer desktop.	iOS/Android Birth: The shift from desks to pockets begins.
2010s	z14/z15: Pervasive encryption for banks.	Cloud Dominance: Linux runs 90% of the internet.	Windows 10: Windows as a Service (Auto-updates).	App Economy: Smartphones become the primary computer.
2020s	z16: AI-on-chip to stop fraud in real-time.	Open Source Everywhere: Android Supercomputers.	Windows 11: AI (Copilot) & Android app support.	Hybrid Cloud: Everything is connected via 5G/Cloud.

b) List the OS under the following category (8 marks)

- CLI vs GUI

GUI	CLI
Windows	Unix
Mobile OS	IBM

- Proprietary vs Open Source

PROPRIETARY	OPEN SOURCE
Windows	Mobile OS (Android)
IBM	Mobile OS (iOS)

c) Answer the following regarding CLI OS (6 marks):

- What are the advantages of CLI over GUI?
  (i)Lightweight: A CLI does not need to load heavy graphics, icons, or animations. It uses very little RAM and CPU
  (ii)Direct Action: You can perform complex tasks in a single line. For example, you can find, copy, and rename 100 specific files in one step, which might take 100 separate clicks in a GUI.
- What are the advantages of GUI over CLI?
  (i)Multitasking - You can have a web browser, a music player, and a document open side-by-side. You can see everything happening at once and resize windows to fit your needs.
  (ii)Ease of Use - You can explore menus to find features you didn't know existed. In a CLI, if you don't know the command exists, you can't use it.

- Which one do you think is better – CLI or GUI? Explain in one sentence.
  - Generally, i think GUI is better for usability and learning due to its intuitive visual nature.

Part Two (10 Marks)

a) Pick what you think is the best OS (or one of the best) among them.
   - Open source.

b) For the OS that you have picked at (a), describe THREE points what you think to be the best features of the OS.
   Freedom When you use Windows or iOS, you are renting the experience. Microsoft or Apple can decide to stop supporting your computer or force an update you don't like.
   Longevity With Open Source, you truly own the system. You can install modern Linux on a 15 year

old laptop and it will run perfectly safe and fast, saving you money on new hardware.
Faster Fixes: When a security hole is found, the community often fixes it in hours, whereas proprietary companies might take weeks to release a patch.

c) Describe THREE features you want the OS to have in the future.
(i) Holography- I want my OS to advance from the flat screen entirely. Instead of staring at a 2D Zoom window or what's app, I want the system to project 3D life-size holograms of my family and friends right into my room.
(ii) Programmable Matter Control- I also want my OS to control the texture of the device itself. If I am reading a book, the screen should physically feel rough like paper. If I need to type, the smooth glass should physically rise to form tactile keyboard buttons and then disappear when I'm done. I want the software to be able to change the shape of the hardware.
(iii)Temporal Computing- I don't want to dig through backups or versions. I want to be able to scroll my entire system back to 'Last Tuesday at 2:00 PM.' The whole screen should shift to exactly how it looked then windows open, unsaved drafts present, cursor in the same spot. I grab the file I lost and then scroll forward to 'Now.' I want Undo for the entire computer, not just one document.

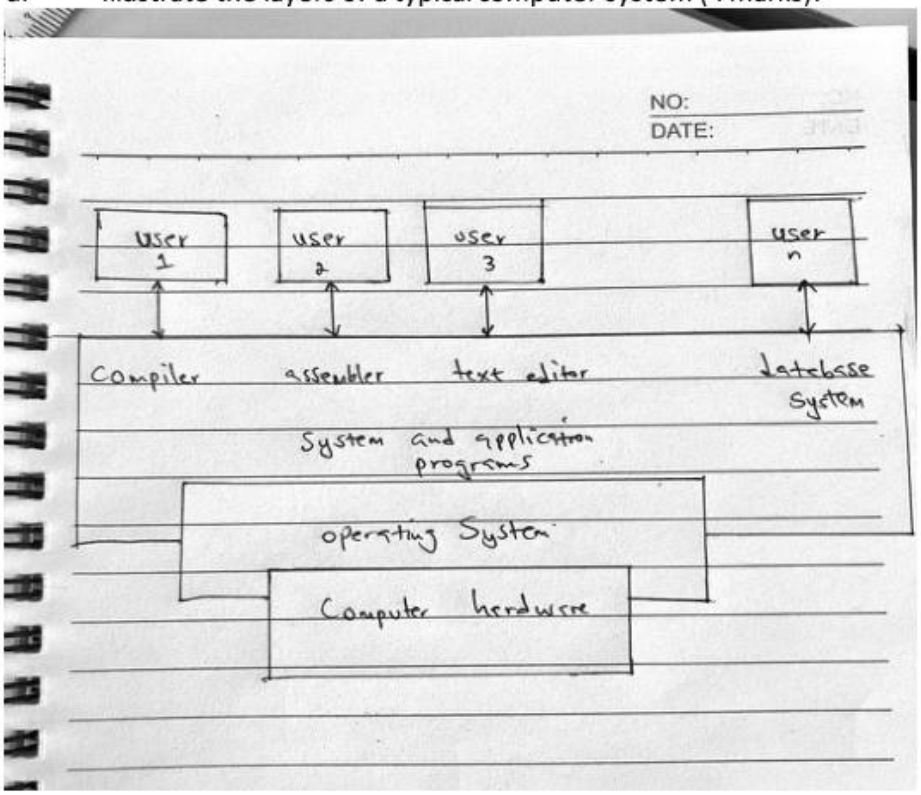
Part Three (36 marks)

a. What are the four basic resources of an OS? (2 marks)
(i) CPU
(ii) RAM
(iii) Storage
(iv) Input/Output devices

b. OS also known as **system software.**

c. OS development can be described as starting with (i) no OS; to (ii) RAM (iii) storage and I/O.

d. Illustrate the layers of a typical computer system (4 marks).



e. Explain the terms below (14 marks):

1) Batch processing

- The computer collects data into groups and processes them all at once later, usually during off peak hours, without user interaction.

Examples;

- Credit card bills
- Bank statements

2) Multiprocessing

- where a computer uses two or more CPUs to physically process multiple tasks at the exact same time.

Examples;

- Supercomputers
- Multi core Smartphones

3) Multiprogramming

- Running multiple programs on a single processor by switching between them, maximizing CPU utilization.

Examples;

- Typing while Downloading.
- Listening to Spotify in the background while working on a spreadsheet

4) Time-Sharing

- Multiple users share system resources simultaneously, with the CPU switching rapidly between tasks to give the illusion of dedicated access

Examples;

- Airline Systems
- University Servers

5) Virtual Machine

- This is a software-based computer that runs an operating system inside another OS.

Examples;

- Running Windows on a Mac computer.
- Developers testing apps on different OS versions using virtual machines instead of separate hardware

6) Cloud Computing

- Storing data and running programs over the internet instead of on your computer's hard drive.

Example;

- Google drive
- Netfilx

7) Artificial Intelligence

- Simulation of human intelligence by machines, enabling them to learn, reason, and make decisions based on data.

Examples;

- Copilot
- Gemini

f. Explain the terms below relating to computer and processor technology (12 marks):

1) Vacuum tube

- A vacuum tube is an early electronic device used to control electric current in first-generation computers

2) Transistor

- Compared to vacuum tubes, transistors are much smaller, faster, cheaper, and more reliable. They allowed computers to shrink from the size of a room to the size of a large cabinet it also was useed in the second generation of computers.

3) Integrated circuit

- A method of combining thousands of transistors onto a single tiny slice of silicon, it was used in the third generation of computers.

4) Microchip

- A small wafer of semiconductor material (silicon) that holds electronic circuits, they are the foundation of all modern digital devices (phones, cars, washing machines).

5) Single processor chip

- This is a specific type of microchip that contains the entire Central Processing Unit (CPU) on a single component

6) Multicore processor chip

- A single physical chip that contains two or more CPUs (cores) working together. This is the **5th** Generation and current standard of computing.

References

Amdahl, G. M., Blaauw, G. A., & Brooks, F. P. (1964). Architecture of the IBM System360. *IBM Journal of Research and Development*, 8(2), 87-101.

Freiberger, P., & Swaine, M. (2000). *Fire in the Valley: The Making of the Personal Computer* (2nd ed.). McGraw-Hill.

Haase, C. (2021). *Androids: The Team That Built the Android Operating System*. Chet Haase.

IBM Corporation. (2000). *Linux on zSeries: Fiber Channel Protocol Implementation Guide*. IBM Redbooks. SG24-6344.

Lammle, T. (2020). *CCNA Certification Study Guide, Volume 2*. Sybex.

Lowe-Norris, A. (2000). *Windows 2000 Active Directory*. O'Reilly Media.

McKusick, M. K., Bostic, K., & Karels, M. J. (1996). *The Design and Implementation of the 4.4BSD Operating System*. Addison-Wesley.

Meaney, P., et al. (2022). IBM Telum Processor: The Next-Generation System for Transaction Processing and AI Inference. *IEEE Micro*, 42(2), 17-25.

Merchant, B. (2017). *The One Device: The Secret History of the iPhone*. Little, Brown and Company.

Petzold, C. (1992). *Programming Windows 3.1* (3rd ed.). Microsoft Press.

Pugh, E. W., Johnson, L. R., & Palmer, J. H. (1991). *IBM's 360 and Early 370 Systems*. MIT Press.

Ritchie, D. M., & Thompson, K. (1974). The UNIX Time-Sharing System. *Communications of the ACM*, 17(7), 365–375.

Salus, P. H. (1994). *A Quarter Century of UNIX*. Addison-Wesley.

Siyan, K. S. (1995). *Novell's Guide to NetWare 4.1 Networks*. Novell Press.

Solomon, D. A., & Russinovich, M. E. (2000). *Inside Microsoft Windows 2000* (3rd ed.). Microsoft Press.

Stallman, R. (1985). The GNU Manifesto. *Dr. Dobb's Journal of Software Tools*, 10(3), 30-35.

TOP500 Project. (2023). *Operating System Family Share for 06/2023*. https://www.top500.org

Torvalds, L., & Diamond, D. (2001). *Just for Fun: The Story of an Accidental Revolutionary*. HarperCollins.

United States v. Microsoft Corp., 87 F. Supp. 2d 30 (D.D.C. 2000).

West, J., & Mace, M. (2010). Browsing as the killer app: Explaining the rapid success of Apple's iPhone. *Telecommunications Policy*, 34(5), 270-286.

Young, R., & Rohm, W. (1999). *Under the Radar: How Red Hat Changed the Software Business*. Coriolis Group Books.