**OPERATING SYSTEMS LAB**

**Experiment No. :** 2

**Date :** 04/03/2025  
**Name :** NILAMBAR ELANGBAM  
**Roll No. : 2201CS0005**  
**Section :** CSE

**Experiment Title :** Dual Booting Linux (with Existing Windows 10)

1. **Aim:**

To configure a dual-boot environment on a desktop computer, allowing the user to choose between booting into the existing Windows 10 operating system and a newly installed Linux distribution.

1. **Objectives:**

Upon successful completion of this experiment, students will be able to:

* Prepare a desktop computer with an existing Windows 10 installation for dual booting.
* Create a bootable USB drive for a chosen Linux distribution (e.g., Ubuntu, Fedora, etc.).
* Partition the hard drive to allocate space for the Linux installation without damaging the existing Windows 10 installation.
* Install a Linux distribution in a dual-boot configuration, ensuring the bootloader (GRUB) is correctly configured to allow selection of either Windows 10 or Linux at startup.
* Verify the successful dual-boot setup by booting into both Windows 10 and the installed Linux distribution.
* Understand the concepts of dual-booting, partitioning, and bootloader management in a multi-OS environment.

1. **Materials and Software Required:**

* **Hardware:**
  + Desktop Computer (with pre-installed Windows 10)
  + USB Drive (Minimum 8GB, preferably USB 3.0)
  + Internet Access (For downloading Linux ISO and potentially bootable USB creation tools)
* **Software:**
  + Linux Distribution ISO Image File (e.g., Ubuntu, Fedora, Linux Mint - Download from the official website of your chosen distribution)
  + Rufus or similar Bootable USB creation tool (if needed, or use distribution-specific tools)
  + Disk Partitioning Tool (Built-in Windows Disk Management or GParted from Linux live environment can be used)

1. **Theory (Brief):**

Dual-booting allows a user to have two or more operating systems installed on a single computer and choose which OS to boot into upon startup. In this experiment, we will install Linux alongside an existing Windows 10 installation.

**Partitioning:** To install Linux, we need to create new partitions on the hard drive to accommodate the Linux system files, user data, and swap space. It's crucial to resize existing partitions (typically the Windows partition) or utilize unallocated space to create these new partitions without affecting the existing Windows installation.

**Bootloader (GRUB):** After installing Linux, a bootloader is required to present a menu at startup allowing the user to select which operating system to boot. For Linux, GRUB (GRand Unified Bootloader) is commonly used. The Linux installer will typically configure GRUB to detect both the newly installed Linux system and the existing Windows 10 installation, creating a boot menu with both options.

**Dual-boot process involves:**

1. Creating bootable media for Linux.
2. Booting from the Linux media.
3. Partitioning the hard drive to create space for Linux.
4. Installing Linux onto the newly created partitions.
5. Configuring the bootloader (GRUB) to recognize both OSes.
6. **Procedure:**

**Part 1: Preparing for Linux Installation**

1. **Backup Important Data (CRITICAL STEP):** **Before proceeding, back up all important data from your Windows 10 system.** Partitioning and OS installation carry a risk of data loss if done incorrectly. Back up personal files, documents, and any other critical information.
2. **Check Disk Space in Windows 10:**
   * In Windows 10, open **Disk Management** (Right-click on the Start button and select "Disk Management").
   * Identify the drive where Windows 10 is installed (usually Disk 0, often C: drive).
   * Check the available free space on this drive or any other drive you intend to use for Linux. Ensure you have sufficient free space for the Linux installation (at least 20-30 GB recommended, more for a comfortable experience). If necessary, free up space by deleting unnecessary files or uninstalling programs within Windows.
3. **Disable Fast Startup in Windows 10 (Recommended):**
   * Open **Control Panel** (View by: Large icons or Small icons).
   * Go to **Power Options**.
   * Click on "Choose what the power buttons do" (on the left sidebar).
   * Click on "Change settings that are currently unavailable" (you may need administrator privileges).
   * Uncheck the box next to "Turn on fast startup (recommended)".
   * Click "Save changes". *Disabling Fast Startup can prevent potential issues with dual-booting.*
4. **Download Linux Distribution ISO:**
   * Download the ISO image file of your chosen Linux distribution (e.g., Ubuntu, Fedora, Linux Mint) from its official website. Choose the appropriate architecture (usually 64-bit).
5. **Create Bootable USB Drive:**
   * Use Rufus (or a distribution-recommended tool like dd, Etcher, or distribution-specific USB creators) to create a bootable USB drive from the downloaded Linux ISO image.
   * **Using Rufus (Example):**
     + Run Rufus.
     + Select your USB drive in the "Device" dropdown.
     + In "Boot selection", choose "Disk or ISO image" and select the downloaded Linux ISO file.
     + Leave other settings as default or adjust based on the Linux distribution's recommendations (usually "Standard Linux Installation" or similar).
     + Click "START" and confirm any warnings about data loss on the USB drive.
     + Wait for Rufus to finish creating the bootable USB.

**Part 2: Installing Linux in Dual-Boot Configuration**

1. **Boot from Linux USB:**
   * Plug the bootable Linux USB drive into your desktop computer.
   * Restart the computer.
   * Enter BIOS/UEFI settings (as in Experiment 1, using Del, F2, F12, etc. key during startup).
   * Change the boot order to prioritize booting from the USB drive.
   * Save changes and exit BIOS/UEFI. The computer should now boot from the Linux USB.
2. **Start Linux Installation:**
   * You will likely boot into a "Live" Linux environment from the USB drive.
   * Look for an "Install" or "Install [Distribution Name]" icon on the desktop or in the application menu. Double-click it to start the Linux installer.
3. **Follow the Linux Installer Wizard:**
   * **Language, Keyboard, etc.:** Select your preferred language, keyboard layout, and other initial settings as prompted by the installer.
   * **Installation Type/Partitioning:** This is the crucial step for dual-booting. **Carefully read the options.** You will likely be presented with options like:
     + **"Install alongside Windows"** (or similar, Easy Option): If this option is available and you have enough free space, it's often the simplest choice. The installer will automatically resize the Windows partition and create partitions for Linux in the free space. **Choose this option if you want an automated and easier approach.**
     + **"Something else"** (Manual Partitioning, Advanced Option): This option gives you more control over partitioning. **Choose this option if you want to manually configure partitions or if the "Install alongside" option is not suitable or not available.** If you choose "Something else":
       - You will see a list of your hard drive partitions. **Identify your Windows partitions carefully (usually NTFS file system).**
       - **Find the free space** you created earlier (or unallocated space). If no free space, you may need to resize an existing partition (be very cautious when resizing partitions; ensure you are resizing the correct partition and have backups).
       - **Create partitions for Linux in the free space:**
         * **Root Partition (/):** This is where the main Linux system files will be installed. Create a partition, choose ext4 file system, and set the mount point to /. (Recommended size: at least 20-30GB or more depending on your needs).
         * **Swap Partition (Optional but Recommended):** Used for virtual memory. Create a partition and choose "swap area" as the file system. (Recommended size: Equal to or slightly larger than your RAM, or you can use a swap file instead, depending on the distribution’s recommendation).
         * **Home Partition (/home) (Optional but Recommended):** For user data (documents, downloads, etc.). Create a partition, choose ext4 file system, and set the mount point to /home. (Allocate remaining space or as desired).
       - **Important:** **Do NOT format your Windows partitions.** Only format the newly created partitions for Linux.
   * **Bootloader Installation:** The installer should automatically detect your existing Windows 10 installation and configure GRUB to manage dual-booting. Usually, the installer will suggest installing GRUB to the main hard drive (e.g., /dev/sda or /dev/nvme0n1). Accept the default unless you have specific reasons to change it.
   * **User Account Setup:** Create a user account for your Linux system (username, password, etc.) as prompted by the installer.
   * **Continue and Complete Installation:** Follow the remaining steps of the installer. The installation process will copy files to the new partitions, configure the system, and install GRUB. The computer may restart several times during this process.
4. **Restart and Boot into Linux (First Boot):**
   * Once the installation is complete, the installer will usually prompt you to restart the computer.
   * Remove the USB drive when prompted.
   * Upon restart, you should see the GRUB boot menu. This menu should list both your newly installed Linux distribution and Windows 10.
   * Use the arrow keys to select your Linux distribution and press Enter to boot into Linux.
5. **Verify Linux Installation:**
   * Log in to your newly installed Linux system using the username and password you created.
   * Explore the Linux environment to ensure it's working correctly.
   * Check if you can access the internet, open applications, etc.
6. **Reboot and Boot into Windows 10:**
   * Restart your computer again.
   * At the GRUB boot menu, use the arrow keys to select "Windows 10" (or similar, the label may vary).
   * Press Enter to boot into Windows 10.
   * Verify that Windows 10 boots and works as expected, and that your files are still intact (as they should be if you followed the instructions carefully and backed up your data).

**6. Conclusion:**

In this experiment, we successfully configured a dual-boot environment on a desktop computer, allowing us to boot into either Windows 10 or a chosen Linux distribution. We learned how to prepare for dual-booting by backing up data and creating bootable Linux media, how to partition the hard drive to accommodate Linux alongside an existing Windows installation, and how the Linux installer configures the GRUB bootloader to manage the dual-boot menu.

This experiment provided practical experience in setting up a dual-boot system, which is a valuable skill for users who need to work with both Windows and Linux operating systems. We now understand the fundamental concepts of partitioning and bootloader management in a dual-boot scenario and can apply this knowledge to set up dual-boot systems on other computers or troubleshoot dual-boot related issues.