COSC 439: Operating Systems Project
Title: USB Flash Drive as Physical Key

Objective: The objective of this project is to develop a device driver that uses a USB flash drive as a physical key to control access to specific files and applications. When the designated USB drive is inserted into the system, access to sensitive files and applications will be unlocked. Once the USB drive is removed, those files and applications will be automatically locked or hidden, ensuring that access is restricted when the physical key is not present.

Key Features to Implement:

1. **USB Detection**: Implement functionality to detect the insertion and removal of the designated USB flash drive. The system should recognize the drive using unique identifiers (e.g., Vendor ID, Product ID).
2. **Access Control**: Upon detecting the USB drive, the system will unlock access to predefined files or directories. Once the drive is removed, access should be restricted by locking or hiding those files and applications.
3. **File/Application Locking Mechanism**: Develop a method to automatically lock or hide files and applications when the USB drive is removed. Ensure that these files are only accessible while the designated USB drive is mounted.
4. **User Configuration**: Allow users to configure which files and applications will be controlled by the USB flash drive. This can be done through a configuration file or a simple interface where users can select specific files or directories to be locked/unlocked.

Instructions:

1. **Setup the Driver**: Develop a device driver that monitors the system for the insertion and removal of the specified USB flash drive. The driver will use the unique identifiers of the USB drive to recognize when it is connected or disconnected.
2. **Unlock and Lock Files**: Implement functionality to unlock files or applications when the USB drive is inserted. These files should be automatically locked or hidden when the drive is removed.
3. **User Configuration**: Create a way for users to configure which files or applications will be controlled by the USB drive. This can involve a configuration file or a user interface that lists available files and directories for selection.
4. **Security Measures**: Ensure that only the designated USB drive can unlock the protected files and applications. The system should not allow access using any other USB drives.
5. **Testing and Validation**: Test the device driver to ensure that it accurately detects the insertion and removal of the USB drive, and that files are properly locked and unlocked based on the presence of the drive.

Requirements:

1. **Progress Report:** Submit a progress report outlining encountered challenges, how you have solved them, the current status, and forthcoming steps. Upon submission, feedback will be given for project adjustment based on the provided feedback. **(1 pt)**

2. **Code Implementation**: Implement a fully functional device driver that detects the USB flash drive and controls access to files or applications accordingly. The driver should correctly handle file locking and unlocking based on the presence of the USB. **(6 pts)**
3. **Technical Report**: **(5 pts)**
    - **Introduction**: Define the project objectives and describe the importance of using USB-based access control for securing sensitive data.
    - **Feature Description**: Provide details on each implemented feature, including USB detection, file locking mechanisms, and optional encryption.
    - **Implementation Details**: Describe the technical challenges faced, design decisions made, and the methods used to develop the device driver.
    - **Results Analysis**: Display examples of files or applications being locked and unlocked when the USB drive is inserted or removed.
    - **Conclusion**: Summarize key findings and discuss the effectiveness of USB flash drive-based access control for enhancing data security.
4. **Presentation:** In person presentation that focuses on the technical aspects of the project. Utilize PowerPoint slides to highlight project goals, algorithms employed, implementation details, evaluations, challenges encountered, and insights gained. Additionally, ensure the presentation includes a live demonstration of the project to provide a practical illustration of its functionality. **(5 pts)**
5. **Retrospective and Contribution Report:** Reflect on the Operating Systems (OS) course, summarizing significant lessons learned, their practical relevance, and their impact on understanding OS principles. Additionally, list your own contributions as well as those of your teammates towards the project. **(1 pt)**

Deadlines:

- **Progress Report:** November 17, 2025
- **Presentation and Demo:** December 3–8, 2025 (In person)
    - Presentations will take place in the professor's office (YR 456) or in the library (YR 454).
    - Each group will present together, and all group members must be present.
    - The professor may ask questions or request modifications to the project or source code to verify that the work was done by the students and not generated by GenAI or copied from online sources.
    - Time slots will be provided via Calendly, and students should book a slot according to their convenience.
- **Source Code Submission:** December 8, 2025
- **Technical Report:** December 11, 2025
- **Retrospective and Contribution Report:** December 11, 2025