Satyndieva Adinai COMCEH-24

This Java program implements the Caesar Cipher, a basic encryption technique that shifts letters forward (encryption) or backward (decryption) by a fixed amount within the alphabet. Users can input a string and a shift value, and the program will output both the encrypted and decrypted versions of the text.

I chose to implement this cipher because it introduces fundamental concepts of string manipulation, ASCII arithmetic, and modular arithmetic in encryption. While the algorithm itself is simple, working on it helped reinforce my understanding of loops, conditionals, and character handling in Java.

During development, I faced some challenges, mainly with handling edge cases such as non-alphabetic characters and ensuring that the shift wraps around correctly within the alphabet. Additionally, managing user input required handling issues like the newline character left in the buffer after reading an integer. Overcoming these challenges helped me become more comfortable with Java's Scanner class and string operations.

Program Structure:

caesarCipher(String text, int shift, boolean encrypt): Encrypts or decrypts a given string using the specified shift value.

Adjusts shift direction based on encryption or decryption.

Ensures that only letters are modified while preserving non-alphabetic characters.

Handles both uppercase and lowercase letters correctly.

main(String[] args): Handles user interaction.

Takes user input for text and shift value.

Encrypts the text and displays the result.

Decrypts the text back to its original form and displays it.

Future Improvements:

Implement a more complex encryption technique, such as Vigenère Cipher, to explore polyalphabetic encryption.

Add an option to automatically detect the shift value for decryption, simulating basic cryptanalysis.

Improve the user interface by providing a menu-based interaction, making it more intuitive.

This project was a great exercise in strengthening my Java programming skills, particularly in handling text-based processing and user input. I look forward to improving it further and exploring more advanced cryptographic techniques.