**SIMPLE ATM SIMULATION PROJECT**

**INTRODUCTION:**

**This project presents a basic ATM simulation program that allows users to check their balance, withdraw funds, and deposit money into their account. The program uses PIN authentication for secure access and file handling to persistently store the balance in a file. The menu options, including checking balance, withdrawing, and depositing, are managed with a switch-case structure. Upon entering the correct PIN, the user is presented with a menu to select actions, and the balance is updated and saved after each operation.**

**ALGORITHM:**

**STEP-1: Start the Program.**

**STEP-2: Load Balance:**

**Open the file balance.txt.**

**If the file exists, read the balance from it.**

**If the file does not exist, create a new one with a default balance (e.g., 1000).**

**STEP-3: PIN Authentication:**

**Prompt the user to enter their PIN.**

**Compare the entered PIN with the correct PIN (e.g., 1234).**

**If the PIN is incorrect:**

**Display "Invalid PIN" message.**

**End the program.**

**If the PIN is correct:**

**Proceed to the menu.**

**STEP-4: Display Menu (inside a loop):**

**Show options:**

**1. Check Balance**

**2. Withdraw**

**3. Deposit**

**4. Exit**

**STEP-5: User Chooses an Option:**

**Switch-Case Control Structure is used:**

**Case 1 (Check Balance):**

**Display the current balance.**

**Case 2 (Withdraw):**

**Prompt the user to enter the withdrawal amount.**

**Check if the withdrawal amount is less than or equal to the balance.**

**If yes:**

**Deduct the amount from the balance.**

**Save the new balance to balance.txt.**

**Display "Withdrawal successful!" message.**

**If no:**

**Display "Insufficient balance!" message.**

**Case 3 (Deposit):**

**Prompt the user to enter the deposit amount.**

**Add the deposit amount to the balance.**

**Save the new balance to balance.txt.**

**Display "Deposit successful!" message.**

**Case 4 (Exit):**

**Display "Exiting..." message.**

**Terminate the program.**

**Default Case:**

**If an invalid option is selected, display "Invalid choice!" message.**

**STEP-6: Repeat Step 4 until the user selects Exit (option 4).**

**STEP-7: End the Program.**

**SOURCE CODE:**

**#include <stdio.h>**

**// Predefined PIN and initial balance**

**int pin = 1234;**

**int balance = 1000;**

**// Function to save the current balance to a file**

**void saveBalance() {**

**// Open the file "balance.txt" in write mode**

**FILE \*fp = fopen("balance.txt", "w");**

**// Check if the file opened successfully**

**if (fp == NULL) {**

**printf("Error saving balance!\n");**

**return; // Exit the function if there was an error opening the file**

**}**

**// Write the current balance to the file**

**fprintf(fp, "%d", balance);**

**// Close the file after writing**

**fclose(fp);**

**}**

**// Function to load the balance from the file**

**void loadBalance() {**

**// Try to open the file "balance.txt" in read mode**

**FILE \*fp = fopen("balance.txt", "r");**

**// If the file exists, read the balance**

**if (fp != NULL) {**

**fscanf(fp, "%d", &balance);**

**fclose(fp); // Close the file after reading**

**} else {**

**// If the file doesn't exist, save the initial balance to create the file**

**saveBalance();**

**}**

**}**

**// Function to display the menu of options to the user**

**void displayMenu() {**

**printf("\nATM Menu:\n");**

**printf("1. Check balance\n");**

**printf("2. Withdraw\n");**

**printf("3. Deposit\n");**

**printf("4. Exit\n");**

**printf("Choose an option: ");**

**}**

**int main() {**

**// Load the balance from the file at the start of the program**

**loadBalance();**

**// Variable to store the entered PIN**

**int enteredPin;**

**printf("Enter PIN: ");**

**scanf("%d", &enteredPin);**

**// Check if the entered PIN is correct**

**if (enteredPin == pin) {**

**int choice;**

**// Loop to keep displaying the menu until the user chooses to exit**

**while (1) {**

**// Display the menu options**

**displayMenu();**

**scanf("%d", &choice);**

**// Perform actions based on the user's choice**

**switch (choice) {**

**case 1: {**

**// Case 1: Show current balance**

**printf("Balance: %d\n", balance);**

**break;**

**}**

**case 2: {**

**// Case 2: Withdraw money**

**int withdrawAmount;**

**printf("Enter amount to withdraw: ");**

**scanf("%d", &withdrawAmount);**

**// Check if the user has sufficient balance for the withdrawal**

**if (withdrawAmount <= balance) {**

**balance -= withdrawAmount; // Deduct the amount from balance**

**saveBalance(); // Save the updated balance**

**printf("Withdrawal successful!\n");**

**} else {**

**printf("Insufficient balance!\n");**

**}**

**break;**

**}**

**case 3: {**

**// Case 3: Deposit money**

**int depositAmount;**

**printf("Enter amount to deposit: ");**

**scanf("%d", &depositAmount);**

**balance += depositAmount; // Add the deposited amount to the balance**

**saveBalance(); // Save the updated balance**

**printf("Deposit successful!\n");**

**break;**

**}**

**case 4:**

**// Case 4: Exit the program**

**printf("Exiting...\n");**

**return 0;**

**default:**

**// If the user chooses an invalid option**

**printf("Invalid choice! Please try again.\n");**

**}**

**}**

**} else {**

**// If the entered PIN is incorrect**

**printf("Invalid PIN! Access Denied.\n");**

**}**

**return 0; // Return 0 to indicate successful program execution**

**}**

**OUTPUT:**

**Enter PIN: 1234**

**ATM Menu:**

**1. Check balance**

**2. Withdraw**

**3. Deposit**

**4. Exit**

**Choose an option: 1**

**Balance: 1000**

**ATM Menu:**

**1. Check balance**

**2. Withdraw**

**3. Deposit**

**4. Exit**

**Choose an option: 2**

**Enter amount to withdraw: 500**

**Withdrawal successful!**

**ATM Menu:**

**1. Check balance**

**2. Withdraw**

**3. Deposit**

**4. Exit**

**Choose an option: 1**

**Balance: 500**

**ATM Menu:**

**1. Check balance**

**2. Withdraw**

**3. Deposit**

**4. Exit**

**Choose an option: 3**

**Enter amount to deposit: 200**

**Deposit successful!**

**ATM Menu:**

**1. Check balance**

**2. Withdraw**

**3. Deposit**

**4. Exit**

**Choose an option: 1**

**Balance: 700**

**ATM Menu:**

**1. Check balance**

**2. Withdraw**

**3. Deposit**

**4. Exit**

**Choose an option: 4**

**Exiting…**

**CONCLUSION:**

**The ATM simulation program demonstrates key concepts like file handling, user input validation, and basic banking operations. It ensures secure access through PIN authentication and maintains balance persistence using file I/O. The switch-case structure provides an efficient and user-friendly interface, making it easy to navigate through different options.**