ATM WITHDRAWAL

AIM:

To create a ATM withdrawal program using some of its main function.

ALGORITHM:

- Enter the number of 500, 200, 100, and 50 denomination notes for deposit.
- Compute the total balance using the deposited denominations.
- Display the total deposited amount along with the denomination details.
- Present a menu with options for Withdrawal, Balance Enquiry, and Exit.
- Accept the user's menu selection.
- If Withdrawal is chosen, request the amount to be withdrawn from the user.
- Validate whether the entered amount is positive and a multiple of 50.
- Verify that sufficient balance is available in the account.
- Determine the required number of notes starting from higher denominations based on availability.
- If the exact amount cannot be formed using available denominations, display an appropriate message.
- If withdrawal is successful, update the available denominations accordingly.
- Recalculate and display the remaining balance after withdrawal.
- If Balance Enquiry is selected, display the current balance and available denominations.
- If Exit is selected, display a thank-you message and terminate the program.
- Repeat the menu process until the user chooses to exit.

PROGRAM:

```c
/*
* Program to demonstrate ATM Withdrawal
* Author   : MUTHUGANESH S
* Date     : 20/01/2026
* Filename : ATM.c
* retval   : void
*/

#include <stdio.h>
#include <stdlib.h>

#define FIFTY 50
#define HUNDRED 100
#define TWO_HUNDRED 200
```

```c
#define FIVE_HUNDRED 500
#define MINUS -1

int CalculateTotal(int FiveHundredNotes, int TwoHundredNotes, int HundredNotes,
int FiftyNotes) {

    int TotalAmount = (FiveHundredNotes * FIVE_HUNDRED) + (TwoHundredNotes *
TWO_HUNDRED) +(HundredNotes * HUNDRED) + (FiftyNotes * FIFTY);

    return TotalAmount;
}

void CalculateBalance(int FiveHundredNotes, int TwoHundredNotes, int
HundredNotes, int FiftyNotes) {

    int TotalAmount = CalculateTotal(FiveHundredNotes, TwoHundredNotes,
HundredNotes, FiftyNotes);

    printf("Current Balance: %d\n", TotalAmount);
    printf("\nDenominations:\n");
    printf("500 notes: %d\n", FiveHundredNotes);
    printf("200 notes: %d\n", TwoHundredNotes);
    printf("100 notes: %d\n", HundredNotes);
    printf("50 notes: %d\n", FiftyNotes);
}

int CalculateWithdrawal(int Amount, int *FiveHundredNotes, int *TwoHundredNotes,
int *HundredNotes, int *FiftyNotes) {

    int RemainingAmount = Amount;
    int Fifty_Notes, Hundred_Notes, TwoHundred_Notes, FiveHundred_Notes;


    if(Amount%FIFTY != 0|| Amount <= 0){
        return MINUS; // Cannot dispense amount not multiple of 50
    }

    FiveHundred_Notes = RemainingAmount / FIVE_HUNDRED;

    if(FiveHundred_Notes > *FiveHundredNotes){
    FiveHundred_Notes = *FiveHundredNotes;
    }

    RemainingAmount -= FiveHundred_Notes * FIVE_HUNDRED;   // ✓ MISSING


    TwoHundred_Notes = RemainingAmount / TWO_HUNDRED;

    if(TwoHundred_Notes > *TwoHundredNotes){
        TwoHundred_Notes = *TwoHundredNotes;
    }

    RemainingAmount -= TwoHundred_Notes * TWO_HUNDRED;

    Hundred_Notes = RemainingAmount / HUNDRED;
```

```c
    if(Hundred_Notes > *HundredNotes){
        Hundred_Notes = *HundredNotes;
    }

    RemainingAmount -= Hundred_Notes * HUNDRED;

    Fifty_Notes = RemainingAmount / FIFTY;

    if(Fifty_Notes > *FiftyNotes){
        Fifty_Notes = *FiftyNotes;
    }

    RemainingAmount -= Fifty_Notes * FIFTY;

    if(RemainingAmount != 0){
        return MINUS; // Cannot dispense exact amount with available
denominations
    }

    *FiveHundredNotes -= FiveHundred_Notes;
    *TwoHundredNotes -= TwoHundred_Notes;
    *HundredNotes -= Hundred_Notes;
    *FiftyNotes -= Fifty_Notes;

    printf("Dispensed Notes:\n");
    printf("%d notes of 500\n", FiveHundred_Notes);
    printf("%d notes of 200\n", TwoHundred_Notes);
    printf("%d notes of 100\n", Hundred_Notes);
    printf("%d notes of 50\n", Fifty_Notes);

    return RemainingAmount; // Should be zero if exact amount can be dispensed
}
int main(void){

    printf("Welcome to the ATM Service\n\n");
    printf("Enter the amount to deposit in denomination\n");

    int FiveHundredNotes, TwoHundredNotes, HundredNotes, FiftyNotes;
    int TotalAmount;

    printf("Enter number of 500 notes: ");
    scanf("%d", &FiveHundredNotes);
    printf("Enter number of 200 notes: ");
    scanf("%d", &TwoHundredNotes);
    printf("Enter number of 100 notes: ");
    scanf("%d", &HundredNotes);
    printf("Enter number of 50 notes: ");
    scanf("%d", &FiftyNotes);

    TotalAmount=CalculateTotal(FiveHundredNotes, TwoHundredNotes, HundredNotes,
FiftyNotes);

    printf("\nTotal Amount Deposited: %d\n", TotalAmount);
    printf("Deposit Successful!\n\n");
```

```
    printf("Menu:\n");
    printf("1. Withdraw Amount\n");
    printf("2. Check Balance\n");
    printf("3. Exit\n");
    int Choice;
    printf("Enter your choice: ");
    scanf("%d", &Choice);

    while(1){

        switch (Choice){
        case 1:{

            int WithdrawAmount;
            printf("Enter amount to withdraw: ");
            scanf("%d", &WithdrawAmount);

            int Status=CalculateWithdrawal(WithdrawAmount, &FiveHundredNotes,
&TwoHundredNotes, &HundredNotes, &FiftyNotes);

            if(Status == MINUS){
                printf("Cannot dispense the requested amount with available
denominations\n");
                break;
            }


            printf("\nPlease collect your cash\n");

            TotalAmount=CalculateTotal(FiveHundredNotes, TwoHundredNotes,
            HundredNotes, FiftyNotes);

            printf("Remaining Balance: %d\n", TotalAmount);
            break;
        }
        case 2:

            CalculateBalance(FiveHundredNotes, TwoHundredNotes, HundredNotes,
FiftyNotes);
            break;

        case 3:
            printf("Thank you for using the ATM service!\n");
            exit(0);
            break;

        default:
            break;
        }

        printf("\nMenu:\n");
        printf("1. Withdraw Amount\n");
        printf("2. Check Balance\n");
        printf("3. Exit\n");
```

```
        printf("Enter your choice: ");
        scanf("%d", &Choice);
    }
    return 0;
}
```

OUTPUT: