S.No: 1	Exp. Name: ***Project Module***	**Date: 2024-06-13**

Aim:

Project Module

Source Code:

hello.c

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

typedef struct {
    char name[100];
    int accountNumber;
    float balance;
    int pin;
} Account;

typedef struct Node {
    Account account;
    struct Node* next;
} Node;

typedef struct {
    Node* front;
    Node* rear;
} Queue;

Queue* createQueue() {
    Queue* q = (Queue*)malloc(sizeof(Queue));
    q->front = q->rear = NULL;
    return q;
}

void enqueue(Queue* q, Account account) {
    Node* temp = (Node*)malloc(sizeof(Node));
    temp->account = account;
    temp->next = NULL;
    if (q->rear == NULL) {
        q->front = q->rear = temp;
        return;
    }
    q->rear->next = temp;
    q->rear = temp;
}

Account dequeue(Queue* q) {
    if (q->front == NULL) {
        printf("Queue is empty.\n");
        exit(1);
    }
    Node* temp = q->front;
    Account account = temp->account;
    q->front = q->front->next;
    if (q->front == NULL) {
        q->rear = NULL;
```

```
void checkBalance(Account* account) {
    printf("Your current balance is: $%.2f\n", account->balance);
}

void withdrawCash(Account* account) {
    float amount;
    printf("Enter amount to withdraw: ");
    scanf("%f", &amount);
    if (amount > 0 && amount <= account->balance) {
        account->balance -= amount;
        printf("Withdrawal successful. New balance: $%.2f\n", account->balance);
    } else {
        printf("Invalid amount or insufficient funds.\n");
    }
}

void depositCash(Account* account) {
    float amount;
    printf("Enter amount to deposit: ");
    scanf("%f", &amount);
    if (amount > 0) {
        account->balance += amount;
        printf("Deposit successful. New balance: $%.2f\n", account->balance);
    } else {
        printf("Invalid amount.\n");
    }
}

void changePin(Account* account) {
    int newPin;
    printf("Enter your new PIN: ");
    scanf("%d", &newPin);
    account->pin = newPin;
    printf("PIN changed successfully.\n");
}

void menu(Account* account) {
    int choice;
    do {
        printf("\n=== ATM Menu ===\n");
        printf("1. Check Balance\n");
        printf("2. Withdraw Cash\n");
        printf("3. Deposit Cash\n");
        printf("4. Change PIN\n");
        printf("5. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);
        switch (choice) {
```

```c
}

int main() {
    Queue* atmQueue = createQueue();
    int n;
    printf("Welcome to the ATM Simulator\n");
    printf("Enter the number of users in the queue: ");
    scanf("%d", &n);

    for (int i = 0; i < n; i++) {
        Account account;
        printf("\nEnter details for user %d:\n", i + 1);
        printf("Enter your name: ");
        scanf("%s", account.name);
        printf("Enter your account number: ");
        scanf("%d", &account.accountNumber);
        printf("Enter your initial balance: ");
        scanf("%f", &account.balance);
        printf("Set your PIN: ");
        scanf("%d", &account.pin);
        enqueue(atmQueue, account);
    }

    while (!isEmpty(atmQueue)) {
        Account currentAccount = dequeue(atmQueue);
        int pin;
        printf("\nProcessing user: %s\n", currentAccount.name);
        printf("Enter your PIN: ");
        scanf("%d", &pin);
        if (currentAccount.pin == pin) {
            printf("Authentication successful.\n");
            menu(&currentAccount);
        } else {
            printf("Invalid PIN. Skipping to next user.\n");
        }
    }

    printf("All users processed. Exiting ATM Simulator.\n");
    return 0;
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Hello World