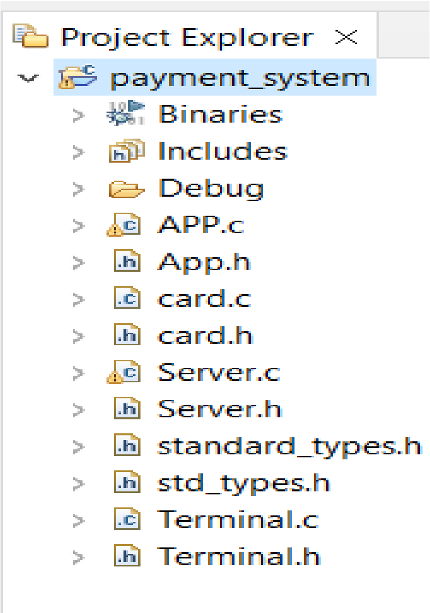
EGFWD PROFESSIONAL EMBEDDED SYSTEMS TRACK

PAYMENT APPLICATION PROJECT

الاسم: مؤمن أحمد فتحى عبد المؤمن

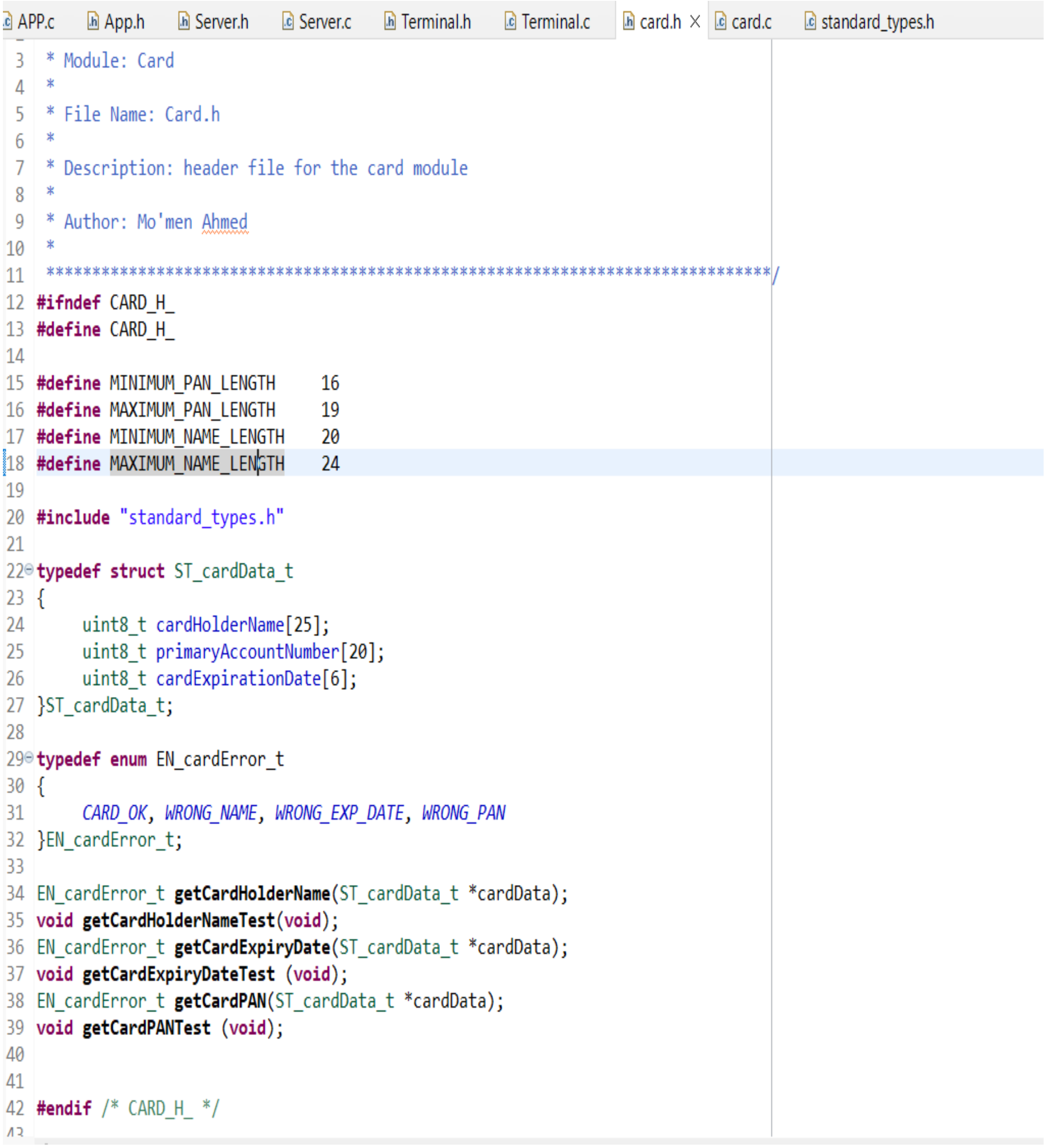
Project explorer:



```
Project Explorer ×
⌄ payment_system
    > Binaries
    > Includes
    > Debug
    > APP.c
    > App.h
    > card.c
    > card.h
    > Server.c
    > Server.h
    > standard_types.h
    > std_types.h
    > Terminal.c
    > Terminal.h
```

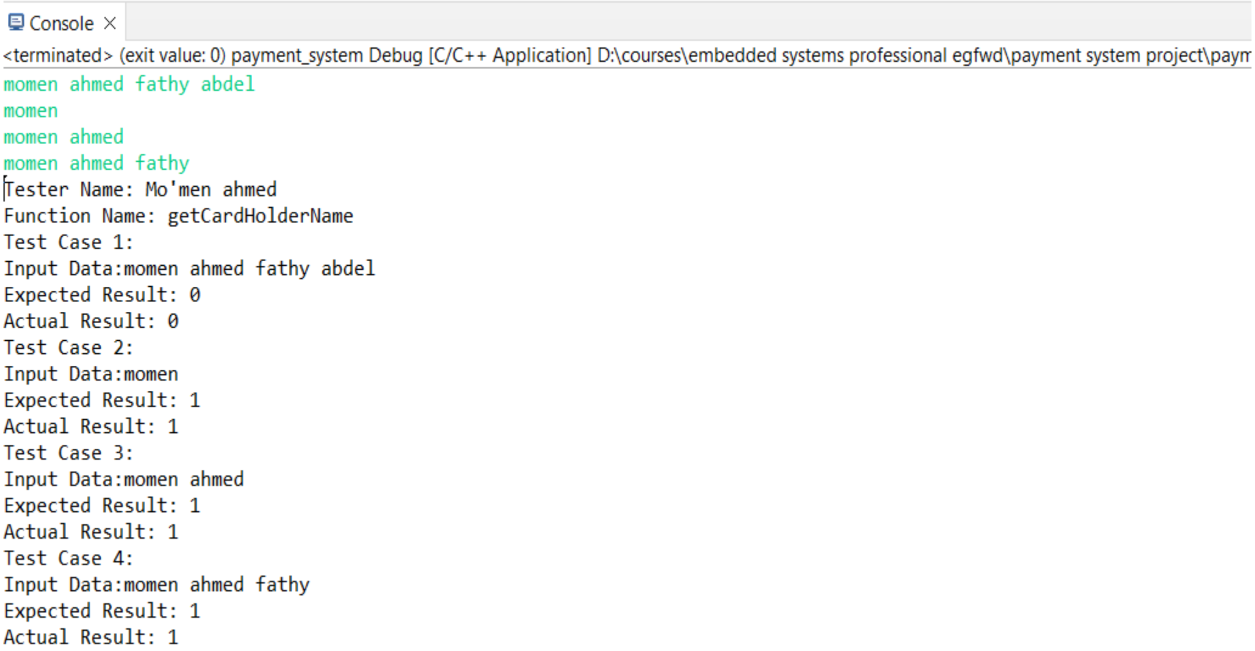
First: Card module:

Header file:

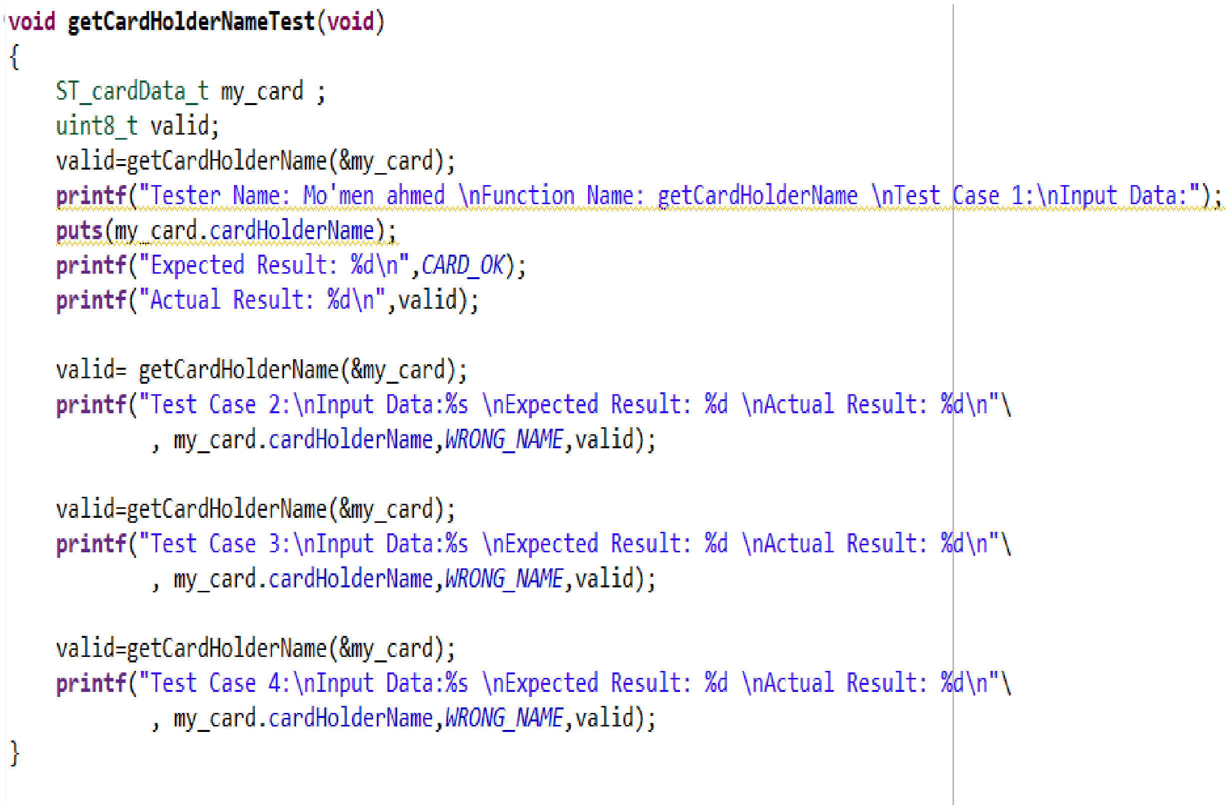


```
 * Module: Card
 *
 * File Name: Card.h
 *
 * Description: header file for the card module
 *
 * Author: Mo'men Ahmed
 *
 ********************************************************************************/
#ifndef CARD_H_
#define CARD_H_

#define MINIMUM_PAN_LENGTH     16
#define MAXIMUM_PAN_LENGTH     19
#define MINIMUM_NAME_LENGTH    20
#define MAXIMUM_NAME_LENGTH    24

#include "standard_types.h"

typedef struct ST_cardData_t
{
     uint8_t cardHolderName[25];
     uint8_t primaryAccountNumber[20];
     uint8_t cardExpirationDate[6];
}ST_cardData_t;

typedef enum EN_cardError_t
{
     CARD_OK, WRONG_NAME, WRONG_EXP_DATE, WRONG_PAN
}EN_cardError_t;

EN_cardError_t getCardHolderName(ST_cardData_t *cardData);
void getCardHolderNameTest(void);
EN_cardError_t getCardExpiryDate(ST_cardData_t *cardData);
void getCardExpiryDateTest (void);
EN_cardError_t getCardPAN(ST_cardData_t *cardData);
void getCardPANTest (void);


#endif /* CARD_H_ */
```

Get card name test function and results:

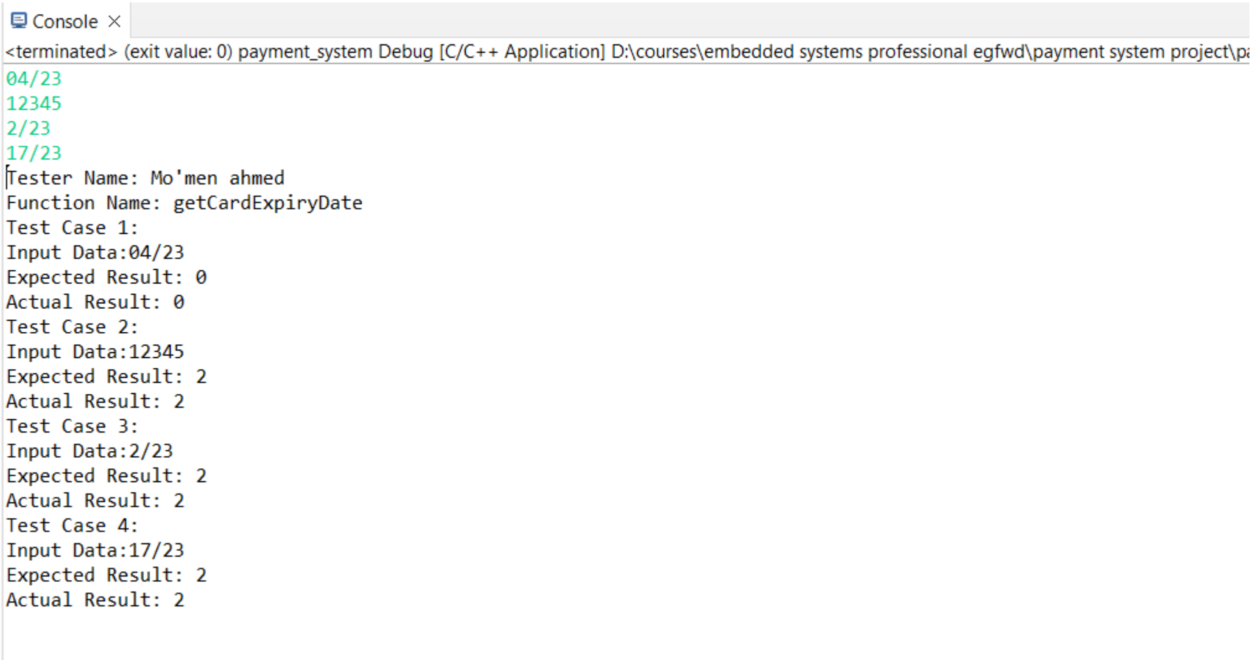


```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\paym
momen ahmed fathy abdel
momen
momen ahmed
momen ahmed fathy
Tester Name: Mo'men ahmed
Function Name: getCardHolderName
Test Case 1:
Input Data:momen ahmed fathy abdel
Expected Result: 0
Actual Result: 0
Test Case 2:
Input Data:momen
Expected Result: 1
Actual Result: 1
Test Case 3:
Input Data:momen ahmed
Expected Result: 1
Actual Result: 1
Test Case 4:
Input Data:momen ahmed fathy
Expected Result: 1
Actual Result: 1
```

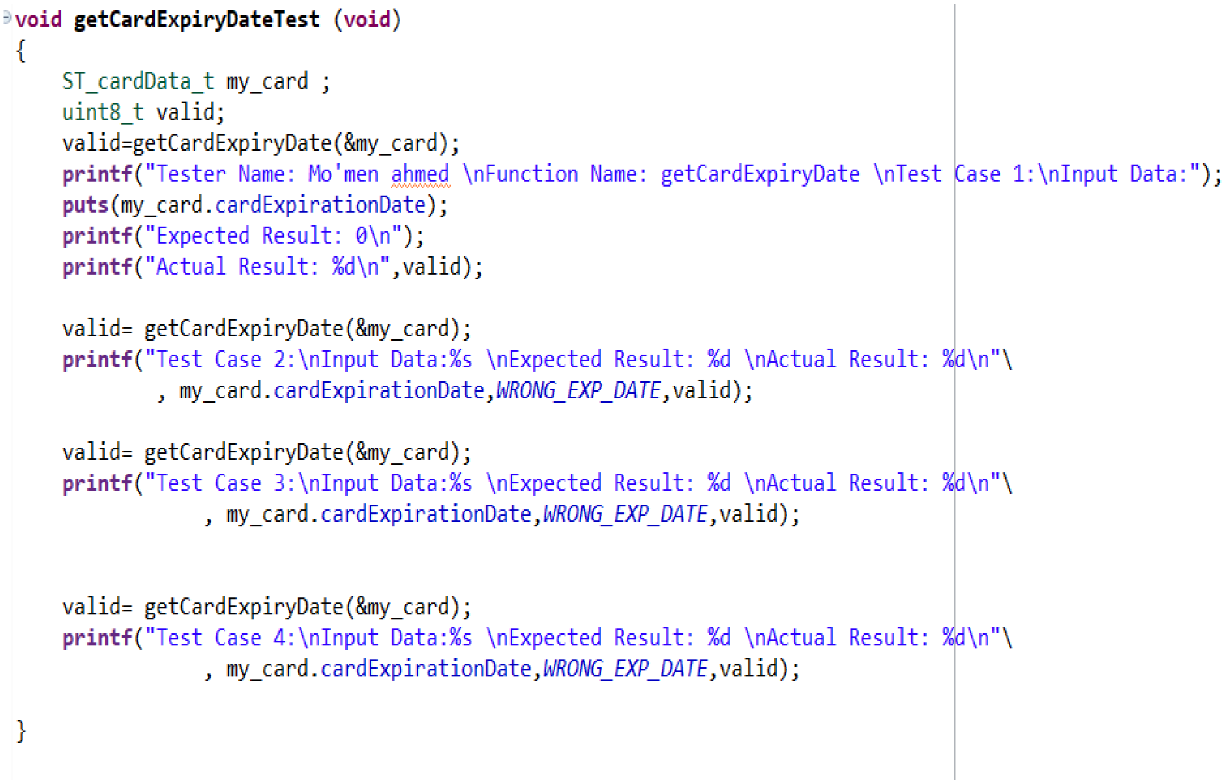


```
void getCardHolderNameTest(void)
{
    ST_cardData_t my_card ;
    uint8_t valid;
    valid=getCardHolderName(&my_card);
    printf("Tester Name: Mo'men ahmed \nFunction Name: getCardHolderName \nTest Case 1:\nInput Data:");
    puts(my_card.cardHolderName);
    printf("Expected Result: %d\n",CARD_OK);
    printf("Actual Result: %d\n",valid);

    valid= getCardHolderName(&my_card);
    printf("Test Case 2:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
            , my_card.cardHolderName,WRONG_NAME,valid);

    valid=getCardHolderName(&my_card);
    printf("Test Case 3:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
            , my_card.cardHolderName,WRONG_NAME,valid);

    valid=getCardHolderName(&my_card);
    printf("Test Case 4:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
            , my_card.cardHolderName,WRONG_NAME,valid);
}
```

Get expiry date test function and results:

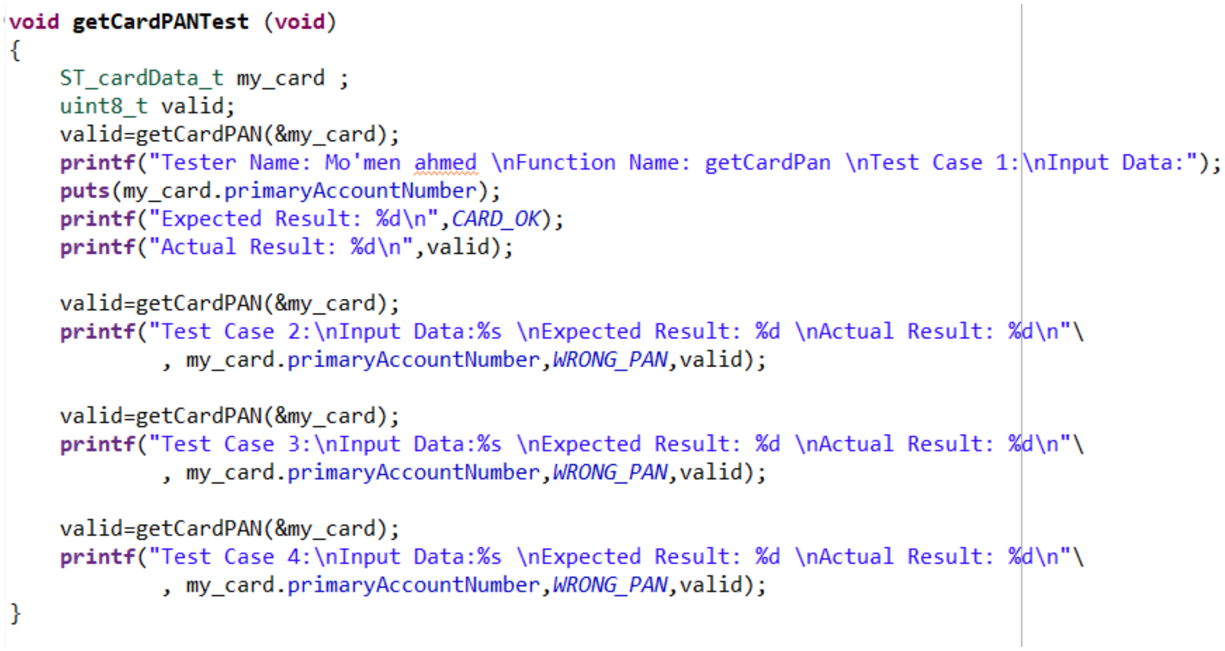


```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\p
04/23
12345
2/23
17/23
Tester Name: Mo'men ahmed
Function Name: getCardExpiryDate
Test Case 1:
Input Data:04/23
Expected Result: 0
Actual Result: 0
Test Case 2:
Input Data:12345
Expected Result: 2
Actual Result: 2
Test Case 3:
Input Data:2/23
Expected Result: 2
Actual Result: 2
Test Case 4:
Input Data:17/23
Expected Result: 2
Actual Result: 2
```

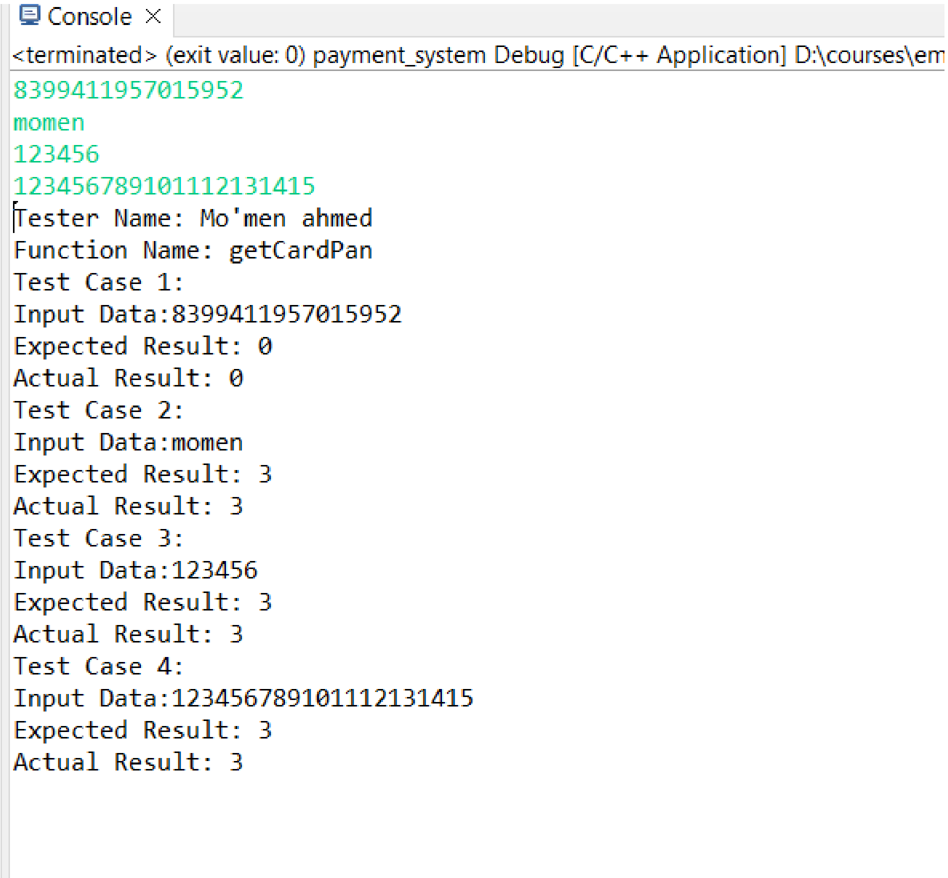


```c
void getCardExpiryDateTest (void)
{
    ST_cardData_t my_card ;
    uint8_t valid;
    valid=getCardExpiryDate(&my_card);
    printf("Tester Name: Mo'men ahmed \nFunction Name: getCardExpiryDate \nTest Case 1:\nInput Data:");
    puts(my_card.cardExpirationDate);
    printf("Expected Result: 0\n");
    printf("Actual Result: %d\n",valid);

    valid= getCardExpiryDate(&my_card);
    printf("Test Case 2:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
            , my_card.cardExpirationDate,WRONG_EXP_DATE,valid);

    valid= getCardExpiryDate(&my_card);
    printf("Test Case 3:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
                , my_card.cardExpirationDate,WRONG_EXP_DATE,valid);


    valid= getCardExpiryDate(&my_card);
    printf("Test Case 4:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
                , my_card.cardExpirationDate,WRONG_EXP_DATE,valid);

}
```

Get card PAN test function and results:



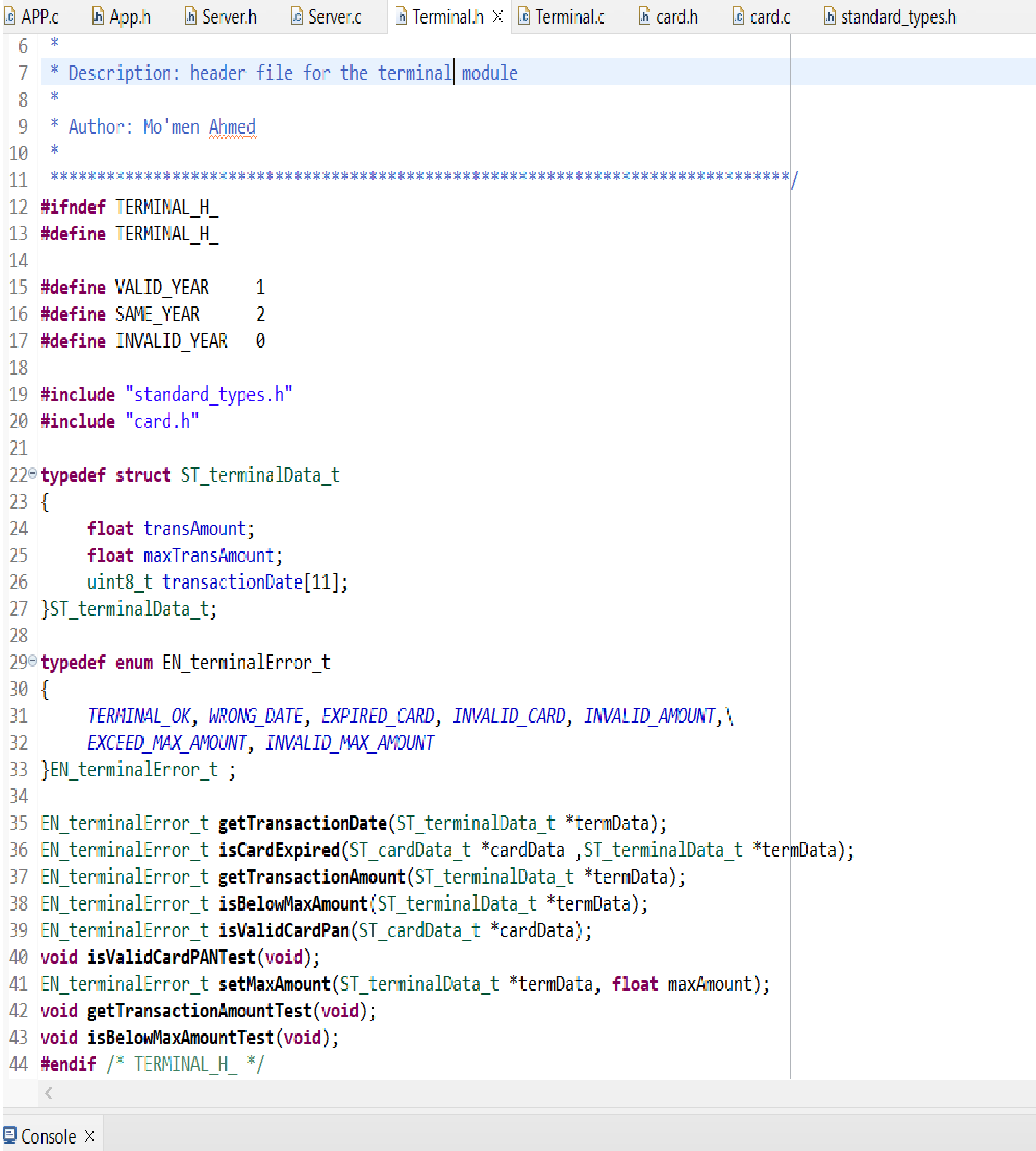
```
void getCardPANTest (void)
{
    ST_cardData_t my_card ;
    uint8_t valid;
    valid=getCardPAN(&my_card);
    printf("Tester Name: Mo'men ahmed \nFunction Name: getCardPan \nTest Case 1:\nInput Data:");
    puts(my_card.primaryAccountNumber);
    printf("Expected Result: %d\n",CARD_OK);
    printf("Actual Result: %d\n",valid);

    valid=getCardPAN(&my_card);
    printf("Test Case 2:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
            , my_card.primaryAccountNumber,WRONG_PAN,valid);

    valid=getCardPAN(&my_card);
    printf("Test Case 3:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
            , my_card.primaryAccountNumber,WRONG_PAN,valid);

    valid=getCardPAN(&my_card);
    printf("Test Case 4:\nInput Data:%s \nExpected Result: %d \nActual Result: %d\n"\
            , my_card.primaryAccountNumber,WRONG_PAN,valid);
}
```



```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\em
8399411957015952
momen
123456
123456789101112131415
Tester Name: Mo'men ahmed
Function Name: getCardPan
Test Case 1:
Input Data:8399411957015952
Expected Result: 0
Actual Result: 0
Test Case 2:
Input Data:momen
Expected Result: 3
Actual Result: 3
Test Case 3:
Input Data:123456
Expected Result: 3
Actual Result: 3
Test Case 4:
Input Data:123456789101112131415
Expected Result: 3
Actual Result: 3
```

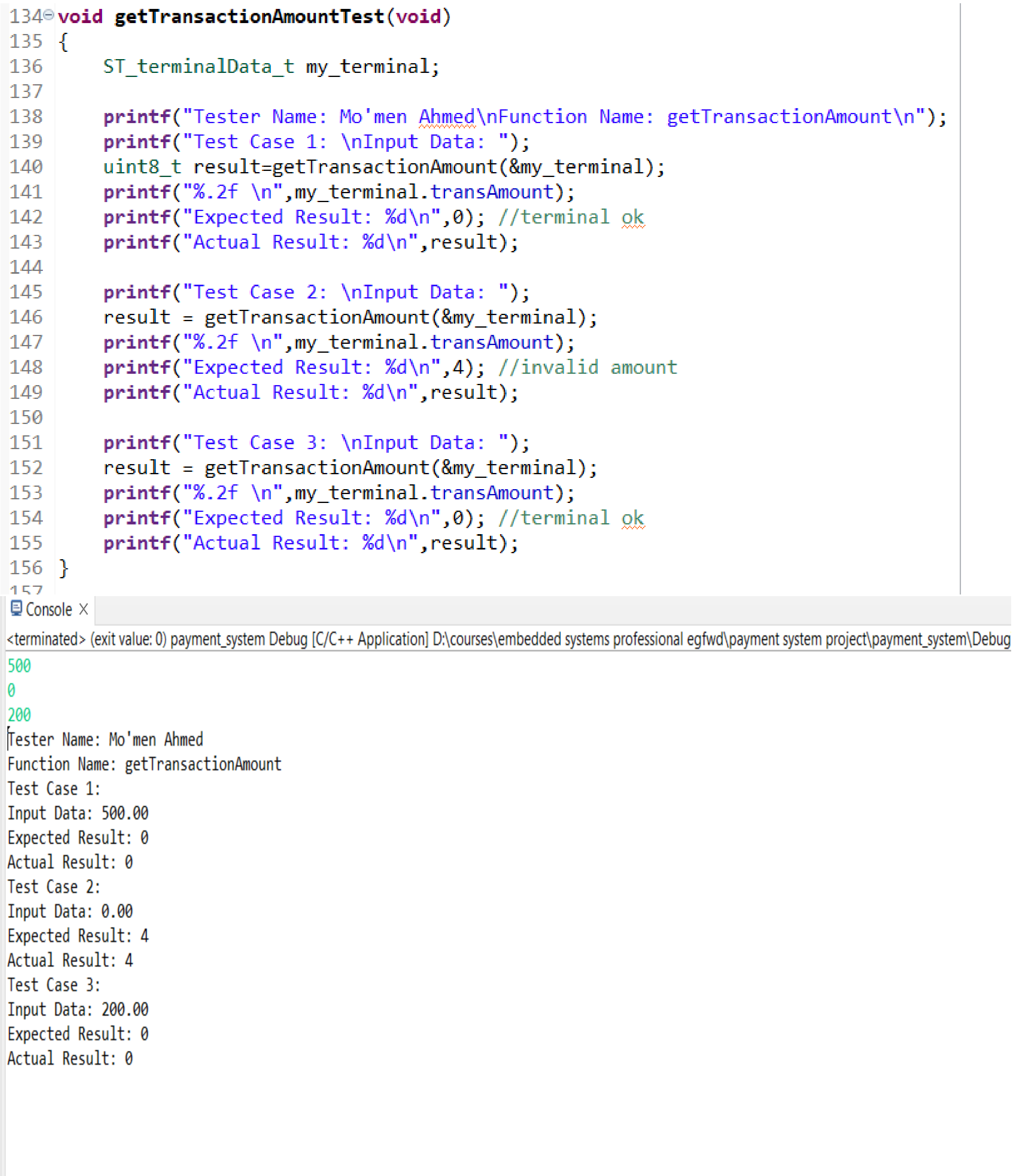
Second: Terminal module:

Header file:



```c
 *
 * Description: header file for the terminal module
 *
 * Author: Mo'men Ahmed
 *
 *******************************************************************************/
#ifndef TERMINAL_H_
#define TERMINAL_H_

#define VALID_YEAR     1
#define SAME_YEAR      2
#define INVALID_YEAR   0

#include "standard_types.h"
#include "card.h"

typedef struct ST_terminalData_t
{
	float transAmount;
	float maxTransAmount;
	uint8_t transactionDate[11];
}ST_terminalData_t;

typedef enum EN_terminalError_t
{
	TERMINAL_OK, WRONG_DATE, EXPIRED_CARD, INVALID_CARD, INVALID_AMOUNT,\
	EXCEED_MAX_AMOUNT, INVALID_MAX_AMOUNT
}EN_terminalError_t ;

EN_terminalError_t getTransactionDate(ST_terminalData_t *termData);
EN_terminalError_t isCardExpired(ST_cardData_t *cardData ,ST_terminalData_t *termData);
EN_terminalError_t getTransactionAmount(ST_terminalData_t *termData);
EN_terminalError_t isBelowMaxAmount(ST_terminalData_t *termData);
EN_terminalError_t isValidCardPan(ST_cardData_t *cardData);
void isValidCardPANTest(void);
EN_terminalError_t setMaxAmount(ST_terminalData_t *termData, float maxAmount);
void getTransactionAmountTest(void);
void isBelowMaxAmountTest(void);
#endif /* TERMINAL_H_ */
```

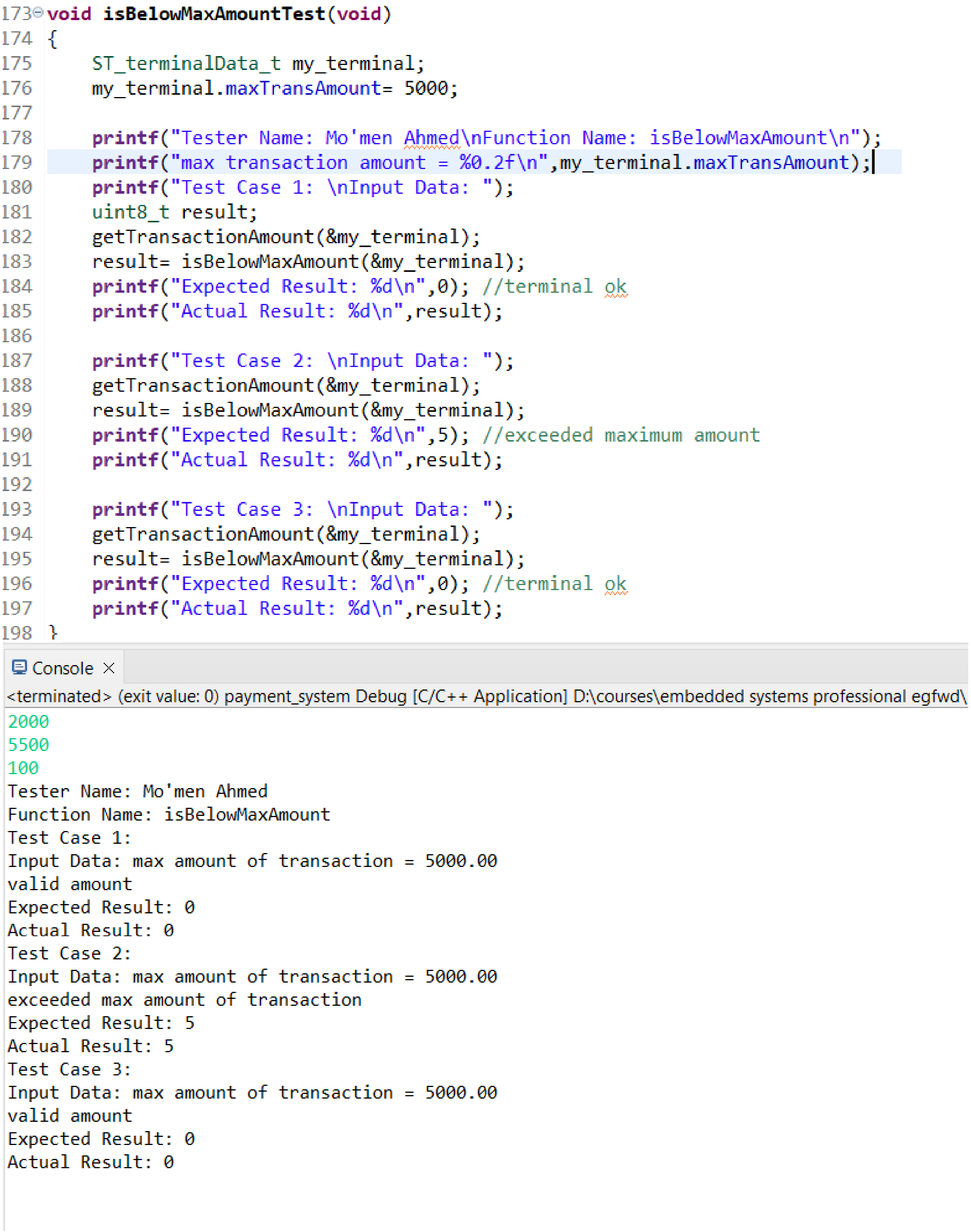
Get transaction amount test function and results:



```c
void getTransactionAmountTest(void)
{
    ST_terminalData_t my_terminal;

    printf("Tester Name: Mo'men Ahmed\nFunction Name: getTransactionAmount\n");
    printf("Test Case 1: \nInput Data: ");
    uint8_t result=getTransactionAmount(&my_terminal);
    printf("%.2f \n",my_terminal.transAmount);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);

    printf("Test Case 2: \nInput Data: ");
    result = getTransactionAmount(&my_terminal);
    printf("%.2f \n",my_terminal.transAmount);
    printf("Expected Result: %d\n",4); //invalid amount
    printf("Actual Result: %d\n",result);

    printf("Test Case 3: \nInput Data: ");
    result = getTransactionAmount(&my_terminal);
    printf("%.2f \n",my_terminal.transAmount);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);
}
```

```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\payment_system\Debug
500
0
200
Tester Name: Mo'men Ahmed
Function Name: getTransactionAmount
Test Case 1:
Input Data: 500.00
Expected Result: 0
Actual Result: 0
Test Case 2:
Input Data: 0.00
Expected Result: 4
Actual Result: 4
Test Case 3:
Input Data: 200.00
Expected Result: 0
Actual Result: 0
```

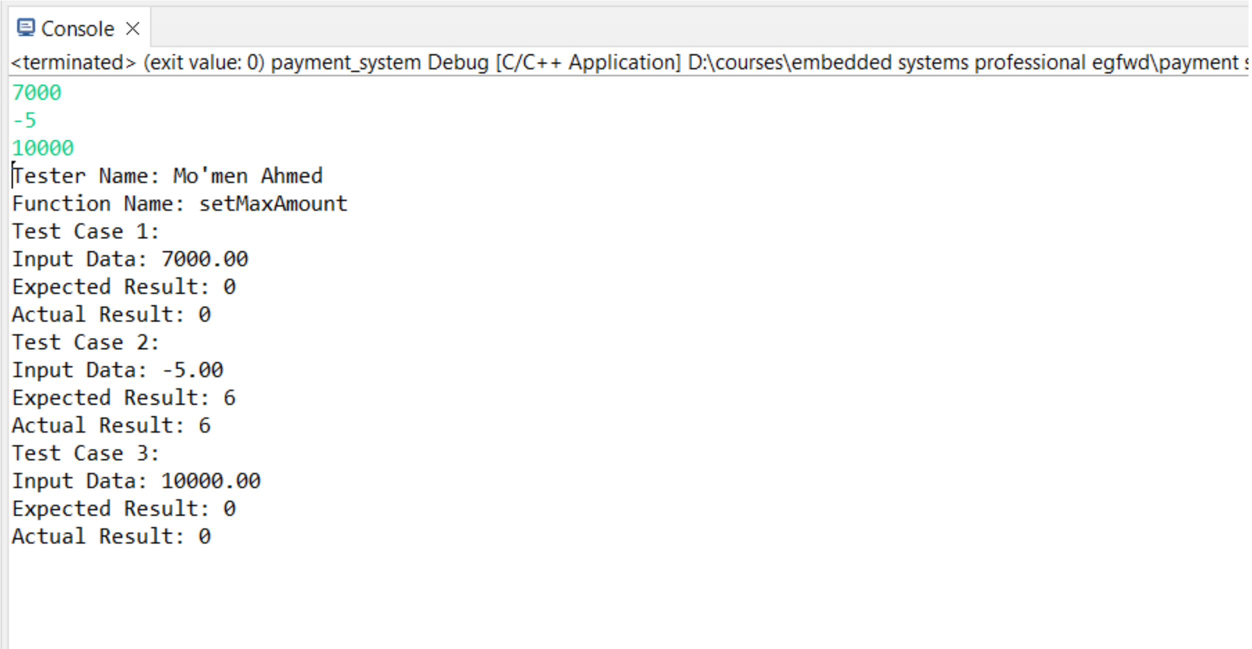
Is below maximum test function and results:



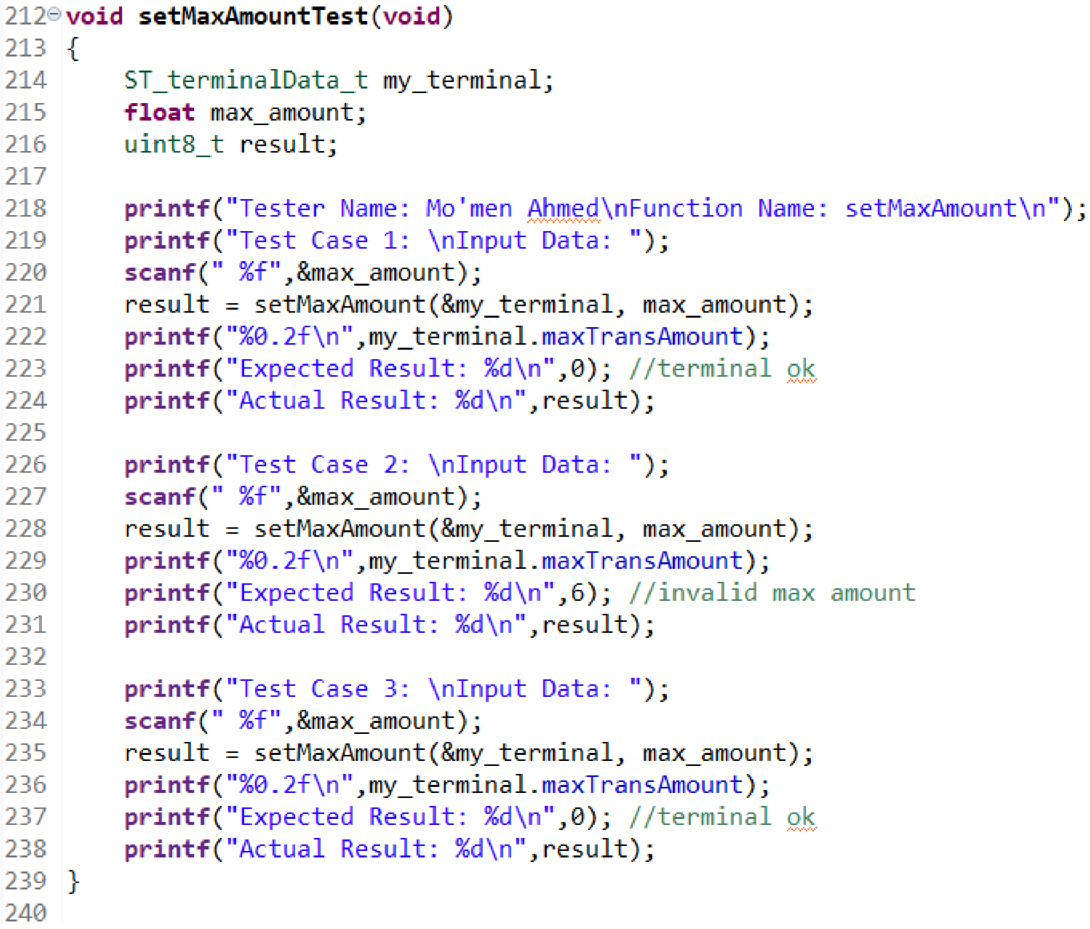
```c
void isBelowMaxAmountTest(void)
{
    ST_terminalData_t my_terminal;
    my_terminal.maxTransAmount= 5000;

    printf("Tester Name: Mo'men Ahmed\nFunction Name: isBelowMaxAmount\n");
    printf("max transaction amount = %0.2f\n",my_terminal.maxTransAmount);
    printf("Test Case 1: \nInput Data: ");
    uint8_t result;
    getTransactionAmount(&my_terminal);
    result= isBelowMaxAmount(&my_terminal);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);

    printf("Test Case 2: \nInput Data: ");
    getTransactionAmount(&my_terminal);
    result= isBelowMaxAmount(&my_terminal);
    printf("Expected Result: %d\n",5); //exceeded maximum amount
    printf("Actual Result: %d\n",result);

    printf("Test Case 3: \nInput Data: ");
    getTransactionAmount(&my_terminal);
    result= isBelowMaxAmount(&my_terminal);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);
}
```

```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\
2000
5500
100
Tester Name: Mo'men Ahmed
Function Name: isBelowMaxAmount
Test Case 1:
Input Data: max amount of transaction = 5000.00
valid amount
Expected Result: 0
Actual Result: 0
Test Case 2:
Input Data: max amount of transaction = 5000.00
exceeded max amount of transaction
Expected Result: 5
Actual Result: 5
Test Case 3:
Input Data: max amount of transaction = 5000.00
valid amount
Expected Result: 0
Actual Result: 0
```

Set max test function and results:

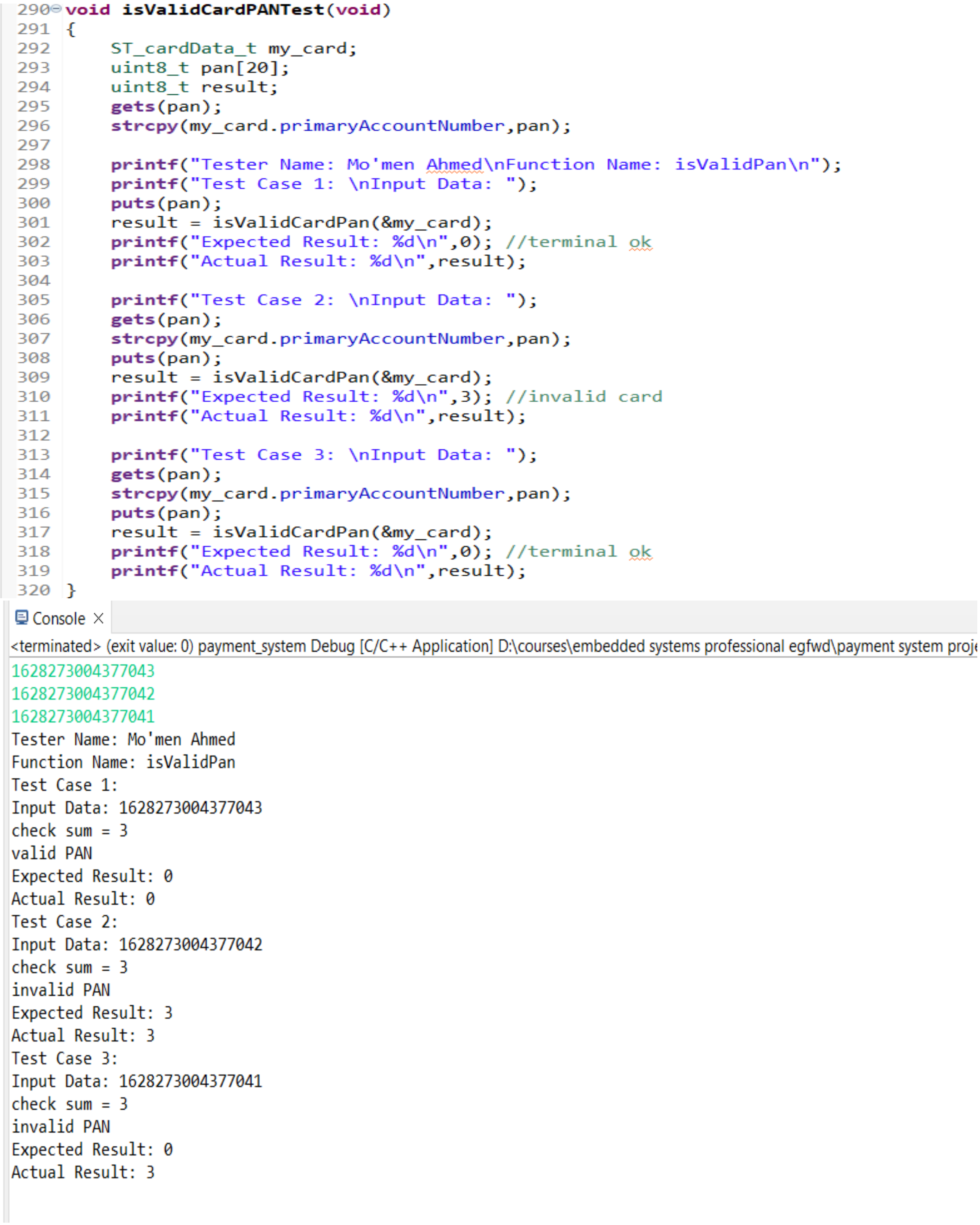


```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment s
7000
-5
10000
Tester Name: Mo'men Ahmed
Function Name: setMaxAmount
Test Case 1:
Input Data: 7000.00
Expected Result: 0
Actual Result: 0
Test Case 2:
Input Data: -5.00
Expected Result: 6
Actual Result: 6
Test Case 3:
Input Data: 10000.00
Expected Result: 0
Actual Result: 0
```



```c
void setMaxAmountTest(void)
{
    ST_terminalData_t my_terminal;
    float max_amount;
    uint8_t result;

    printf("Tester Name: Mo'men Ahmed\nFunction Name: setMaxAmount\n");
    printf("Test Case 1: \nInput Data: ");
    scanf(" %f",&max_amount);
    result = setMaxAmount(&my_terminal, max_amount);
    printf("%0.2f\n",my_terminal.maxTransAmount);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);

    printf("Test Case 2: \nInput Data: ");
    scanf(" %f",&max_amount);
    result = setMaxAmount(&my_terminal, max_amount);
    printf("%0.2f\n",my_terminal.maxTransAmount);
    printf("Expected Result: %d\n",6); //invalid max amount
    printf("Actual Result: %d\n",result);

    printf("Test Case 3: \nInput Data: ");
    scanf(" %f",&max_amount);
    result = setMaxAmount(&my_terminal, max_amount);
    printf("%0.2f\n",my_terminal.maxTransAmount);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);
}
```

Is valid PAN test function and results:

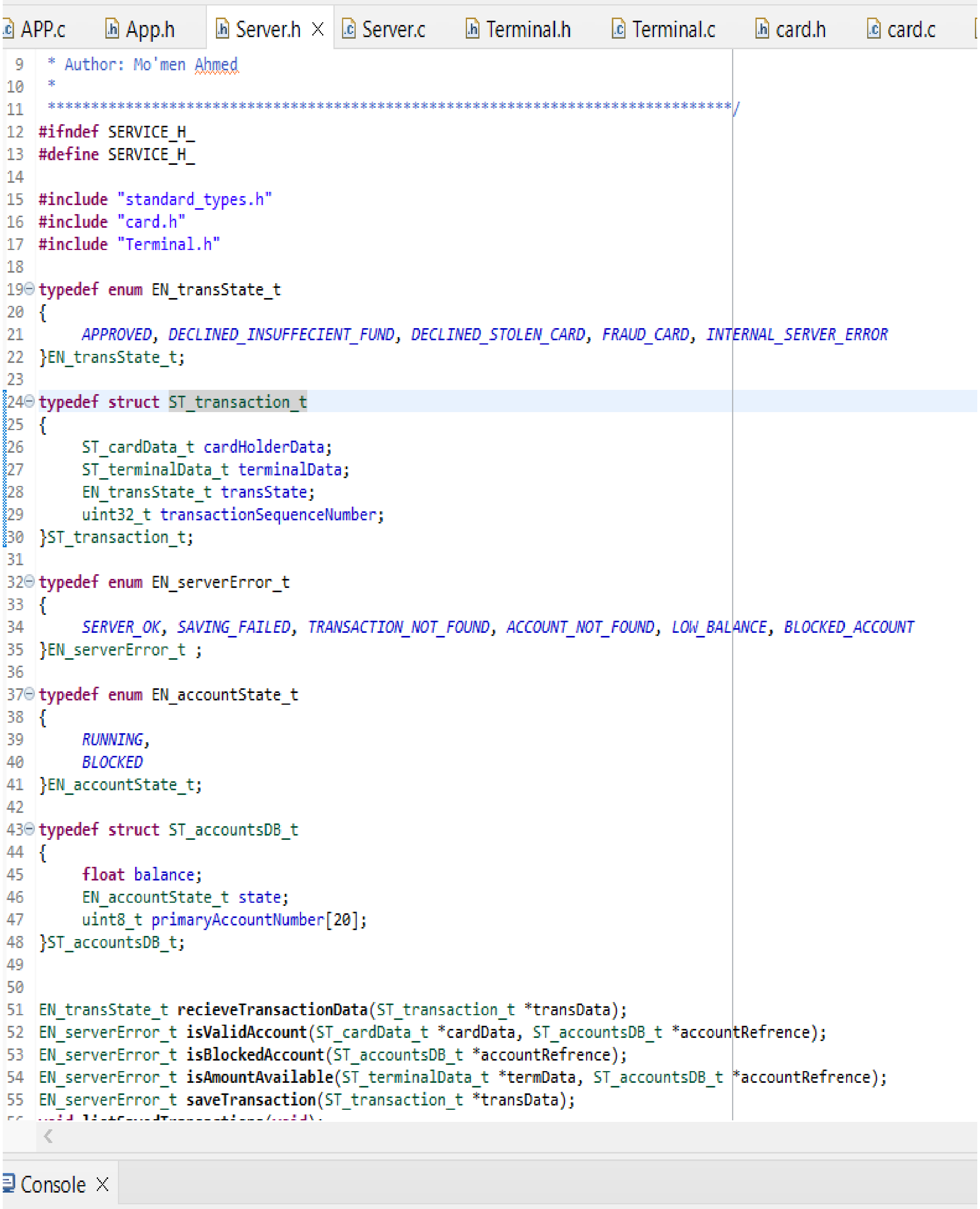


```
void isValidCardPANTest(void)
{
    ST_cardData_t my_card;
    uint8_t pan[20];
    uint8_t result;
    gets(pan);
    strcpy(my_card.primaryAccountNumber,pan);

    printf("Tester Name: Mo'men Ahmed\nFunction Name: isValidPan\n");
    printf("Test Case 1: \nInput Data: ");
    puts(pan);
    result = isValidCardPan(&my_card);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);

    printf("Test Case 2: \nInput Data: ");
    gets(pan);
    strcpy(my_card.primaryAccountNumber,pan);
    puts(pan);
    result = isValidCardPan(&my_card);
    printf("Expected Result: %d\n",3); //invalid card
    printf("Actual Result: %d\n",result);

    printf("Test Case 3: \nInput Data: ");
    gets(pan);
    strcpy(my_card.primaryAccountNumber,pan);
    puts(pan);
    result = isValidCardPan(&my_card);
    printf("Expected Result: %d\n",0); //terminal ok
    printf("Actual Result: %d\n",result);
}
```

```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system proje
1628273004377043
1628273004377042
1628273004377041
Tester Name: Mo'men Ahmed
Function Name: isValidPan
Test Case 1:
Input Data: 1628273004377043
check sum = 3
valid PAN
Expected Result: 0
Actual Result: 0
Test Case 2:
Input Data: 1628273004377042
check sum = 3
invalid PAN
Expected Result: 3
Actual Result: 3
Test Case 3:
Input Data: 1628273004377041
check sum = 3
invalid PAN
Expected Result: 0
Actual Result: 3
```

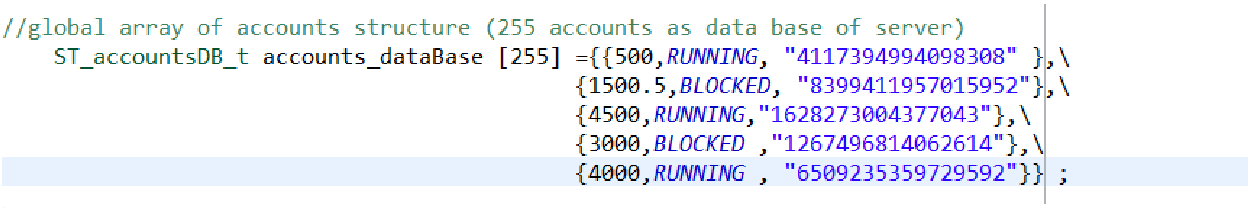
Third: Server module:

Header file:

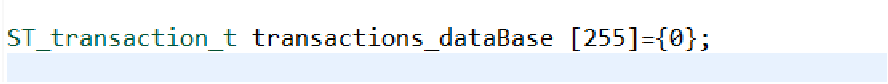


```c
 * Author: Mo'men Ahmed
 *
 *******************************************************************************/
#ifndef SERVICE_H_
#define SERVICE_H_

#include "standard_types.h"
#include "card.h"
#include "Terminal.h"

typedef enum EN_transState_t
{
    APPROVED, DECLINED_INSUFFECIENT_FUND, DECLINED_STOLEN_CARD, FRAUD_CARD, INTERNAL_SERVER_ERROR
}EN_transState_t;

typedef struct ST_transaction_t
{
    ST_cardData_t cardHolderData;
    ST_terminalData_t terminalData;
    EN_transState_t transState;
    uint32_t transactionSequenceNumber;
}ST_transaction_t;

typedef enum EN_serverError_t
{
    SERVER_OK, SAVING_FAILED, TRANSACTION_NOT_FOUND, ACCOUNT_NOT_FOUND, LOW_BALANCE, BLOCKED_ACCOUNT
}EN_serverError_t ;

typedef enum EN_accountState_t
{
    RUNNING,
    BLOCKED
}EN_accountState_t;

typedef struct ST_accountsDB_t
{
    float balance;
    EN_accountState_t state;
    uint8_t primaryAccountNumber[20];
}ST_accountsDB_t;


EN_transState_t recieveTransactionData(ST_transaction_t *transData);
EN_serverError_t isValidAccount(ST_cardData_t *cardData, ST_accountsDB_t *accountRefrence);
EN_serverError_t isBlockedAccount(ST_accountsDB_t *accountRefrence);
EN_serverError_t isAmountAvailable(ST_terminalData_t *termData, ST_accountsDB_t *accountRefrence);
EN_serverError_t saveTransaction(ST_transaction_t *transData);
void listSavedTransactions(void);
```

Accounts and transactions databases initially:

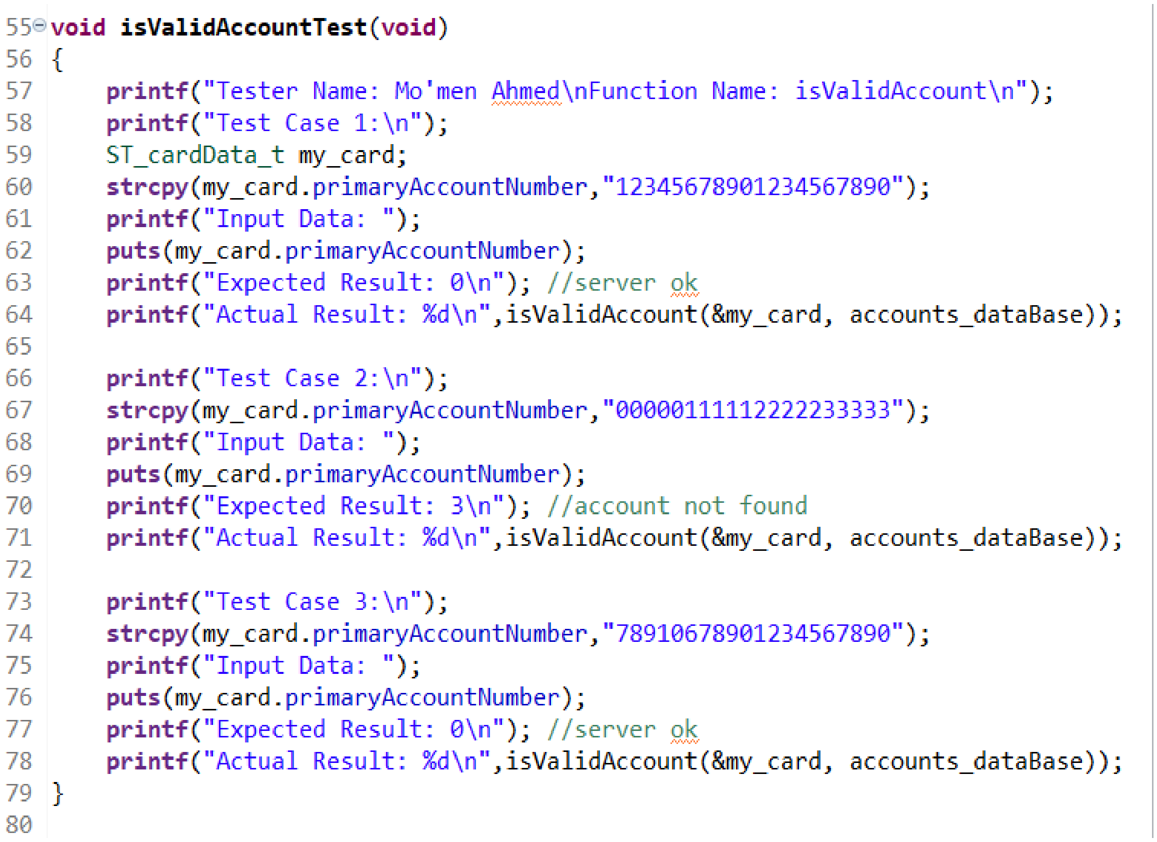


```
//global array of accounts structure (255 accounts as data base of server)
    ST_accountsDB_t accounts_dataBase [255] ={{500,RUNNING, "4117394994098308" },\
                                            {1500.5,BLOCKED, "8399411957015952"},\
                                            {4500,RUNNING,"1628273004377043"},\
                                            {3000,BLOCKED ,"1267496814062614"},\
                                            {4000,RUNNING , "6509235359729592"}} ;
```

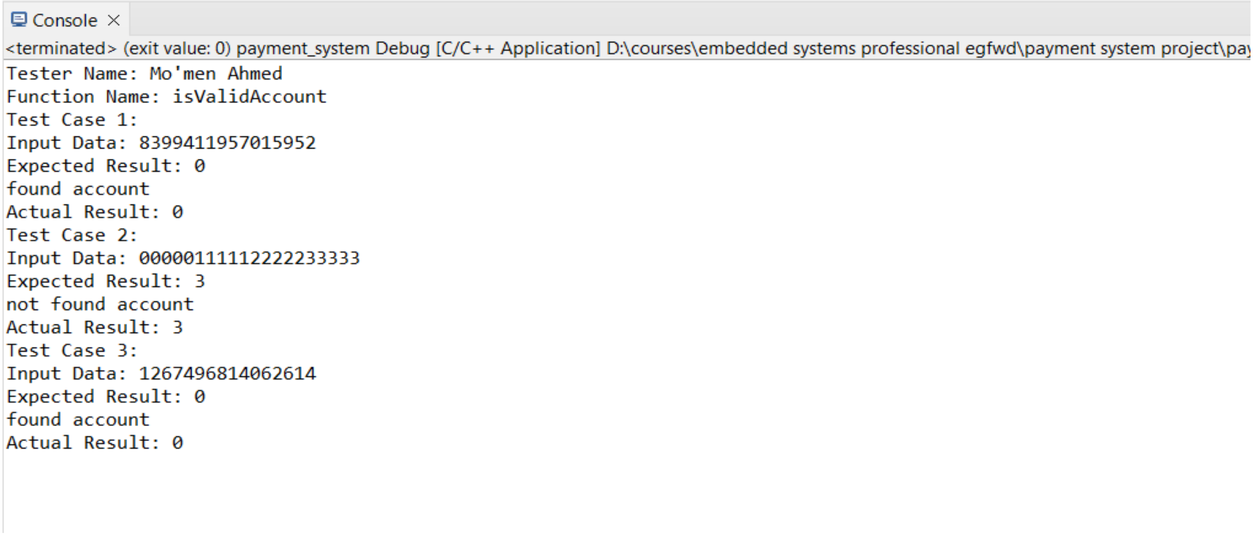


```
ST_transaction_t transactions_dataBase [255]={0};
```

Is valid account test function and results:



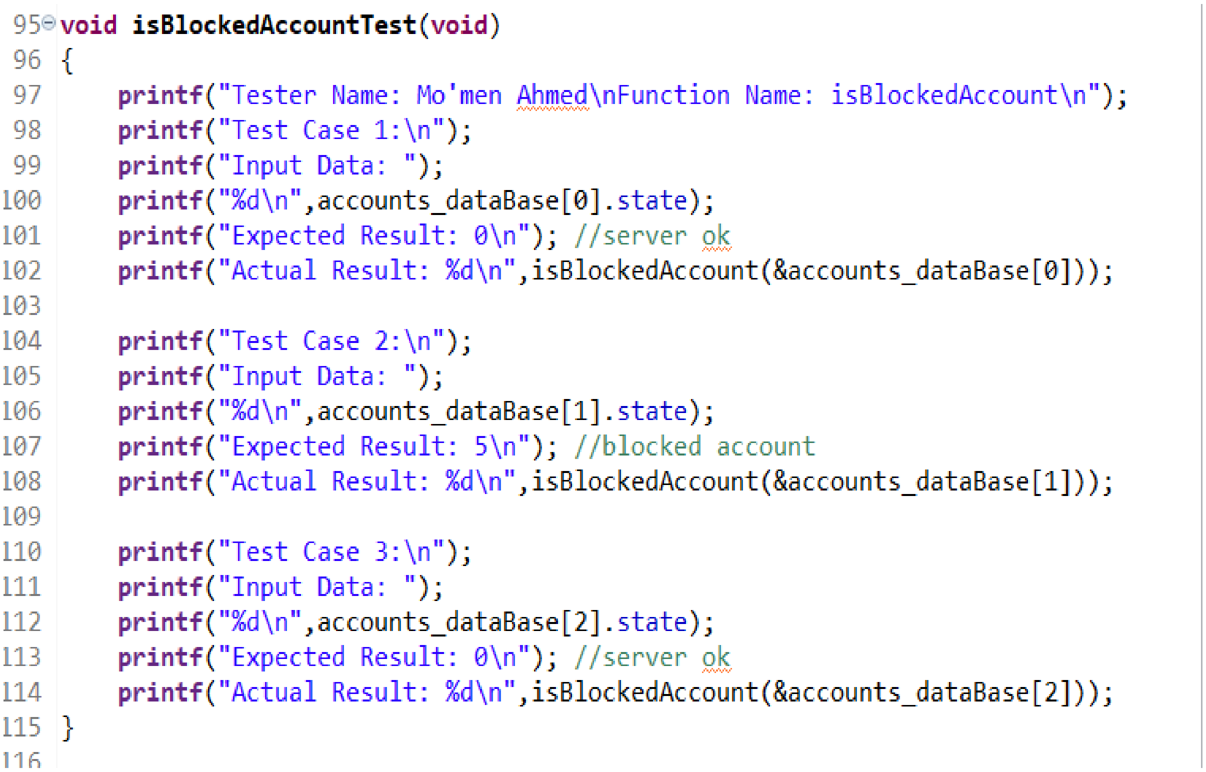
```
void isValidAccountTest(void)
{
    printf("Tester Name: Mo'men Ahmed\nFunction Name: isValidAccount\n");
    printf("Test Case 1:\n");
    ST_cardData_t my_card;
    strcpy(my_card.primaryAccountNumber,"12345678901234567890");
    printf("Input Data: ");
    puts(my_card.primaryAccountNumber);
    printf("Expected Result: 0\n"); //server ok
    printf("Actual Result: %d\n",isValidAccount(&my_card, accounts_dataBase));

    printf("Test Case 2:\n");
    strcpy(my_card.primaryAccountNumber,"00000111112222233333");
    printf("Input Data: ");
    puts(my_card.primaryAccountNumber);
    printf("Expected Result: 3\n"); //account not found
    printf("Actual Result: %d\n",isValidAccount(&my_card, accounts_dataBase));

    printf("Test Case 3:\n");
    strcpy(my_card.primaryAccountNumber,"78910678901234567890");
    printf("Input Data: ");
    puts(my_card.primaryAccountNumber);
    printf("Expected Result: 0\n"); //server ok
    printf("Actual Result: %d\n",isValidAccount(&my_card, accounts_dataBase));
}
```



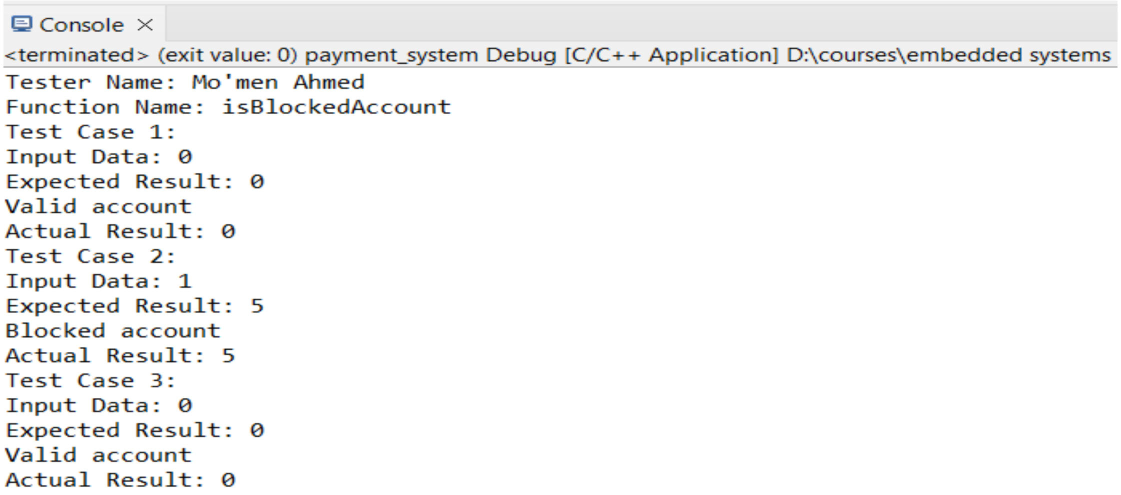
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\pay

```
Tester Name: Mo'men Ahmed
Function Name: isValidAccount
Test Case 1:
Input Data: 8399411957015952
Expected Result: 0
found account
Actual Result: 0
Test Case 2:
Input Data: 00000111112222233333
Expected Result: 3
not found account
Actual Result: 3
Test Case 3:
Input Data: 1267496814062614
Expected Result: 0
found account
Actual Result: 0
```

Is blocked account test function and results:

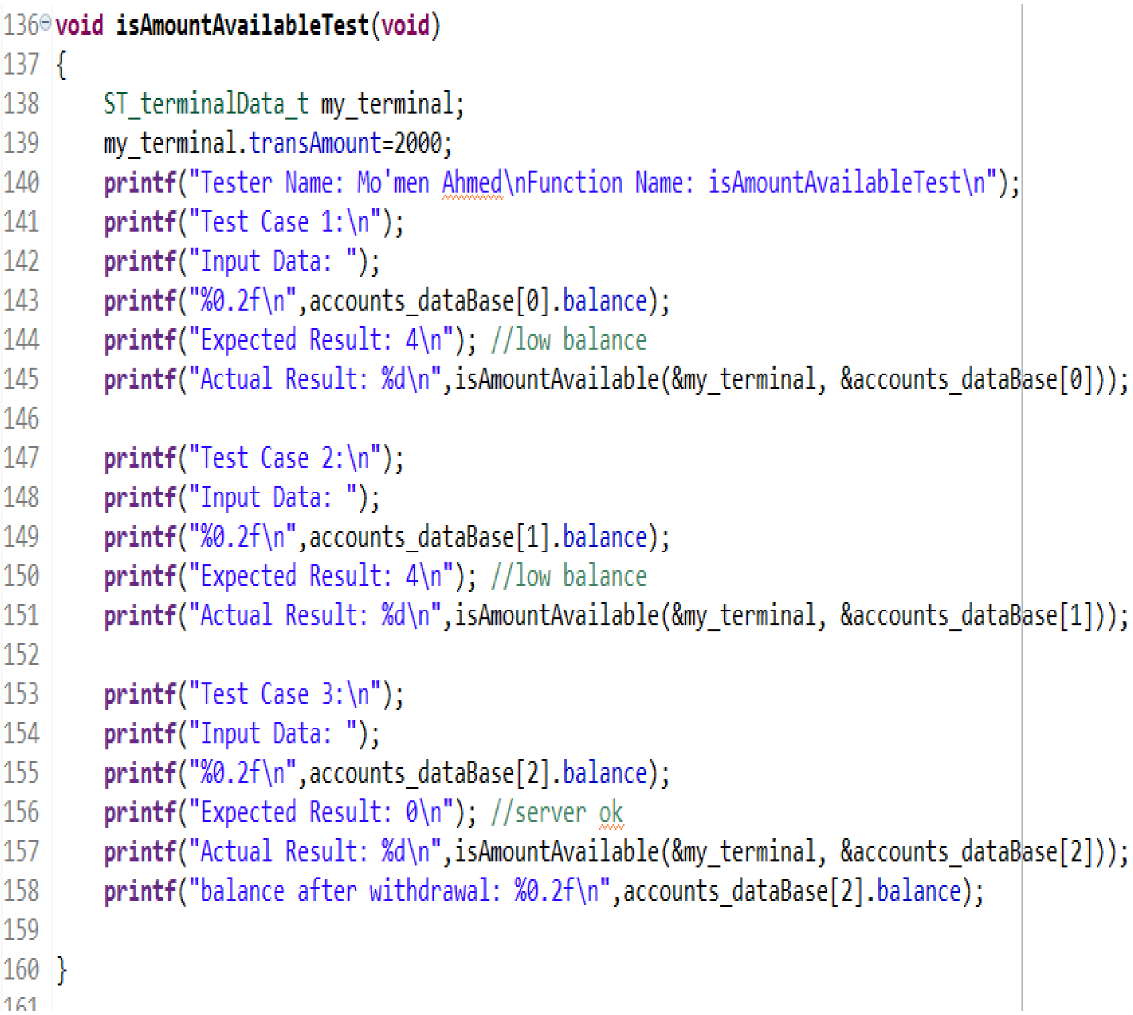


```c
void isBlockedAccountTest(void)
{
    printf("Tester Name: Mo'men Ahmed\nFunction Name: isBlockedAccount\n");
    printf("Test Case 1:\n");
    printf("Input Data: ");
    printf("%d\n",accounts_dataBase[0].state);
    printf("Expected Result: 0\n"); //server ok
    printf("Actual Result: %d\n",isBlockedAccount(&accounts_dataBase[0]));

    printf("Test Case 2:\n");
    printf("Input Data: ");
    printf("%d\n",accounts_dataBase[1].state);
    printf("Expected Result: 5\n"); //blocked account
    printf("Actual Result: %d\n",isBlockedAccount(&accounts_dataBase[1]));

    printf("Test Case 3:\n");
    printf("Input Data: ");
    printf("%d\n",accounts_dataBase[2].state);
    printf("Expected Result: 0\n"); //server ok
    printf("Actual Result: %d\n",isBlockedAccount(&accounts_dataBase[2]));
}
```

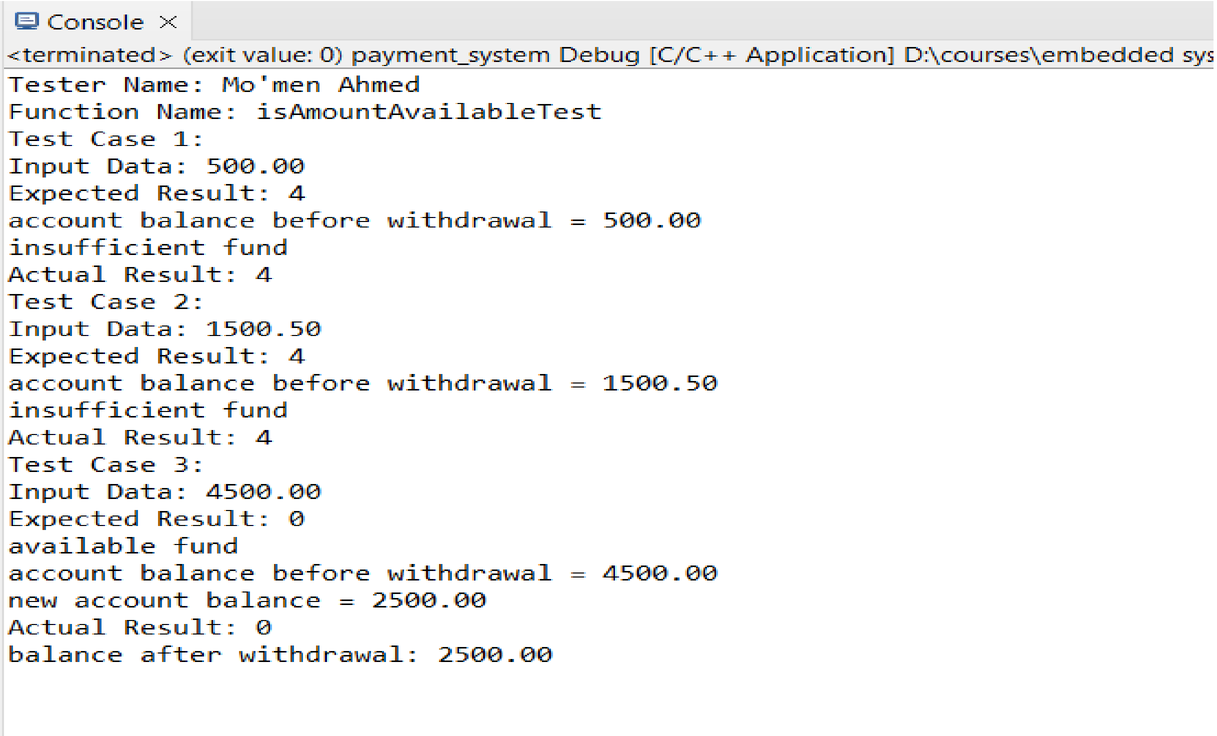


```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems
Tester Name: Mo'men Ahmed
Function Name: isBlockedAccount
Test Case 1:
Input Data: 0
Expected Result: 0
Valid account
Actual Result: 0
Test Case 2:
Input Data: 1
Expected Result: 5
Blocked account
Actual Result: 5
Test Case 3:
Input Data: 0
Expected Result: 0
Valid account
Actual Result: 0
```

Is amount available test function and results:

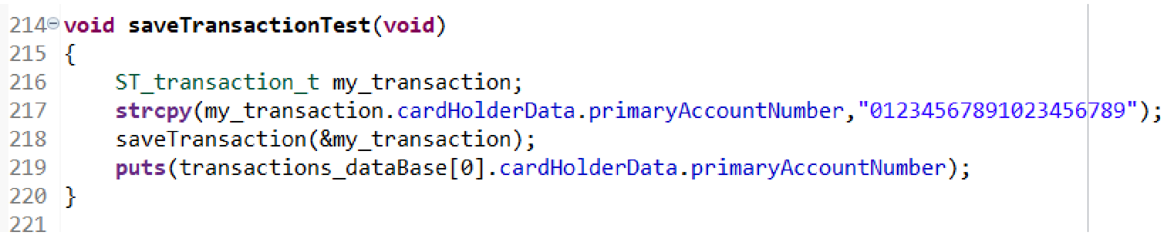


```c
void isAmountAvailableTest(void)
{
    ST_terminalData_t my_terminal;
    my_terminal.transAmount=2000;
    printf("Tester Name: Mo'men Ahmed\nFunction Name: isAmountAvailableTest\n");
    printf("Test Case 1:\n");
    printf("Input Data: ");
    printf("%0.2f\n",accounts_dataBase[0].balance);
    printf("Expected Result: 4\n"); //low balance
    printf("Actual Result: %d\n",isAmountAvailable(&my_terminal, &accounts_dataBase[0]));

    printf("Test Case 2:\n");
    printf("Input Data: ");
    printf("%0.2f\n",accounts_dataBase[1].balance);
    printf("Expected Result: 4\n"); //low balance
    printf("Actual Result: %d\n",isAmountAvailable(&my_terminal, &accounts_dataBase[1]));

    printf("Test Case 3:\n");
    printf("Input Data: ");
    printf("%0.2f\n",accounts_dataBase[2].balance);
    printf("Expected Result: 0\n"); //server ok
    printf("Actual Result: %d\n",isAmountAvailable(&my_terminal, &accounts_dataBase[2]));
    printf("balance after withdrawal: %0.2f\n",accounts_dataBase[2].balance);

}
```

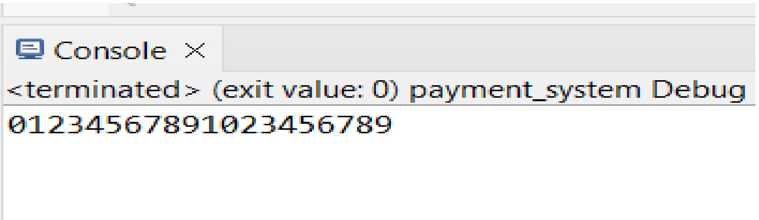


```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded sys
Tester Name: Mo'men Ahmed
Function Name: isAmountAvailableTest
Test Case 1:
Input Data: 500.00
Expected Result: 4
account balance before withdrawal = 500.00
insufficient fund
Actual Result: 4
Test Case 2:
Input Data: 1500.50
Expected Result: 4
account balance before withdrawal = 1500.50
insufficient fund
Actual Result: 4
Test Case 3:
Input Data: 4500.00
Expected Result: 0
available fund
account balance before withdrawal = 4500.00
new account balance = 2500.00
Actual Result: 0
balance after withdrawal: 2500.00
```

Save transaction test function and results:

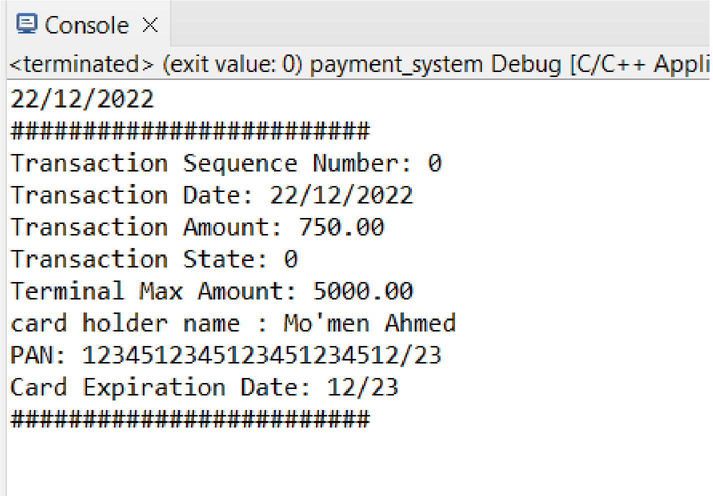


```c
void saveTransactionTest(void)
{
    ST_transaction_t my_transaction;
    strcpy(my_transaction.cardHolderData.primaryAccountNumber,"01234567891023456789");
    saveTransaction(&my_transaction);
    puts(transactions_dataBase[0].cardHolderData.primaryAccountNumber);
}
```

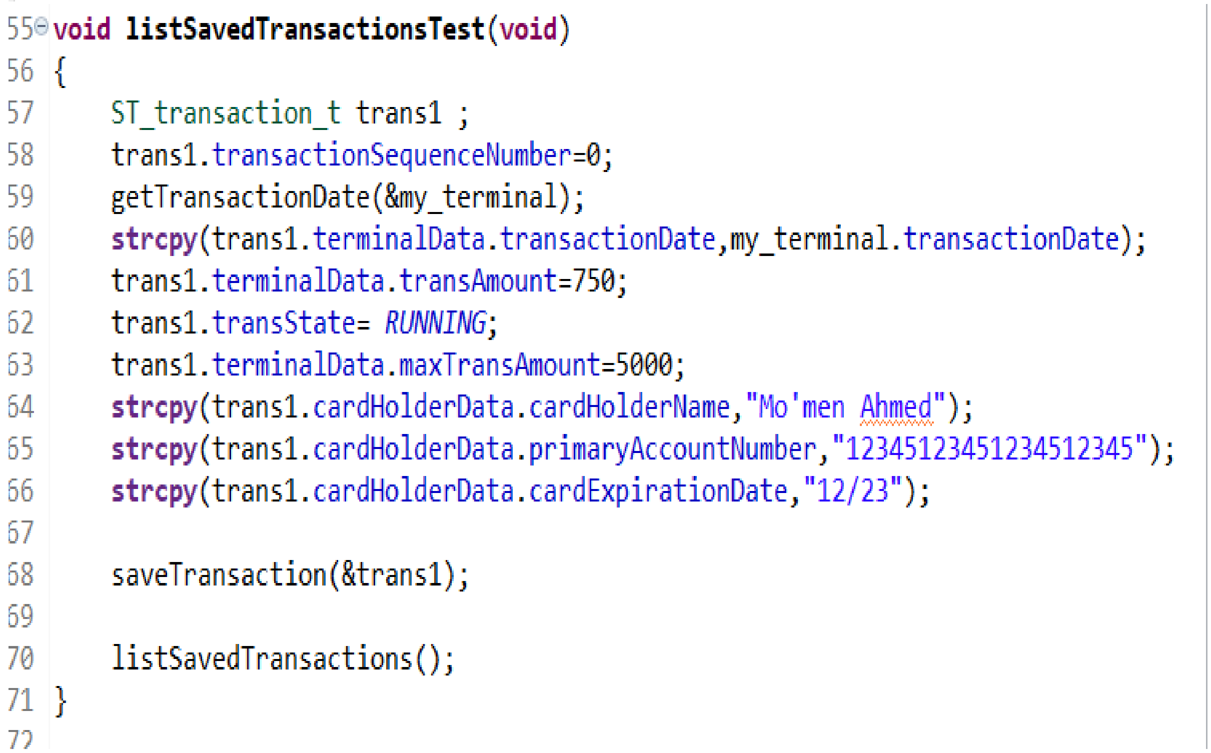


```
<terminated> (exit value: 0) payment_system Debug
01234567891023456789
```

List transactions test function and results:



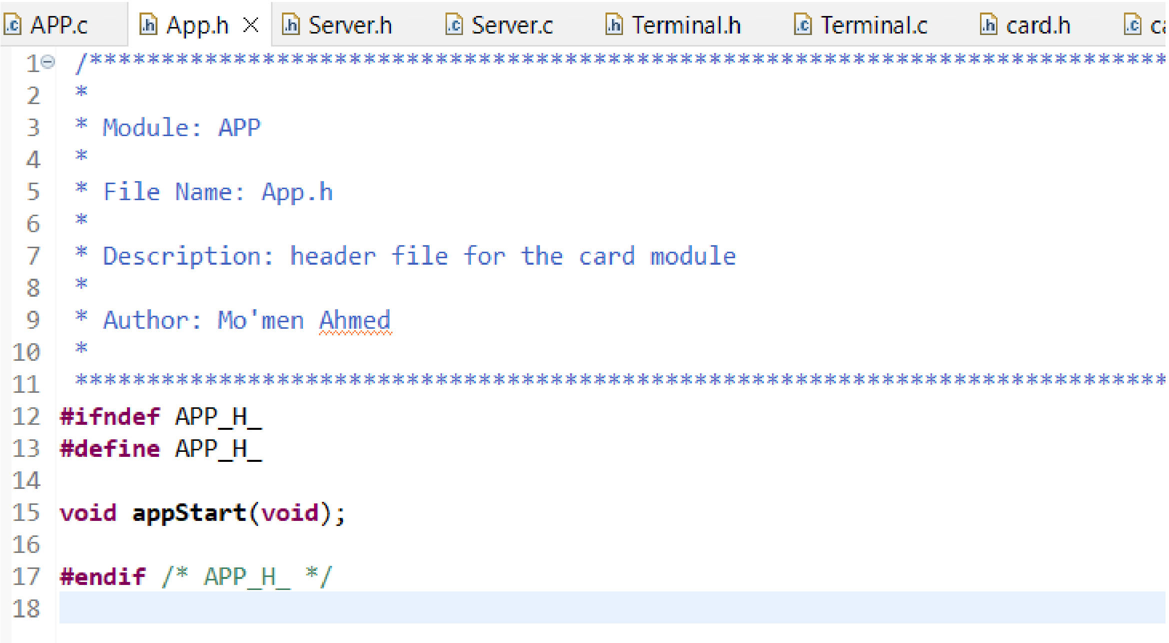
```
<terminated> (exit value: 0) payment_system Debug [C/C++ Appli
22/12/2022
#########################
Transaction Sequence Number: 0
Transaction Date: 22/12/2022
Transaction Amount: 750.00
Transaction State: 0
Terminal Max Amount: 5000.00
card holder name : Mo'men Ahmed
PAN: 1234512345123451234512/23
Card Expiration Date: 12/23
#########################
```



```c
void listSavedTransactionsTest(void)
{
    ST_transaction_t trans1 ;
    trans1.transactionSequenceNumber=0;
    getTransactionDate(&my_terminal);
    strcpy(trans1.terminalData.transactionDate,my_terminal.transactionDate);
    trans1.terminalData.transAmount=750;
    trans1.transState= RUNNING;
    trans1.terminalData.maxTransAmount=5000;
    strcpy(trans1.cardHolderData.cardHolderName,"Mo'men Ahmed");
    strcpy(trans1.cardHolderData.primaryAccountNumber,"12345123451234512345");
    strcpy(trans1.cardHolderData.cardExpirationDate,"12/23");

    saveTransaction(&trans1);

    listSavedTransactions();
}
```

Fourth: App module & test cases:

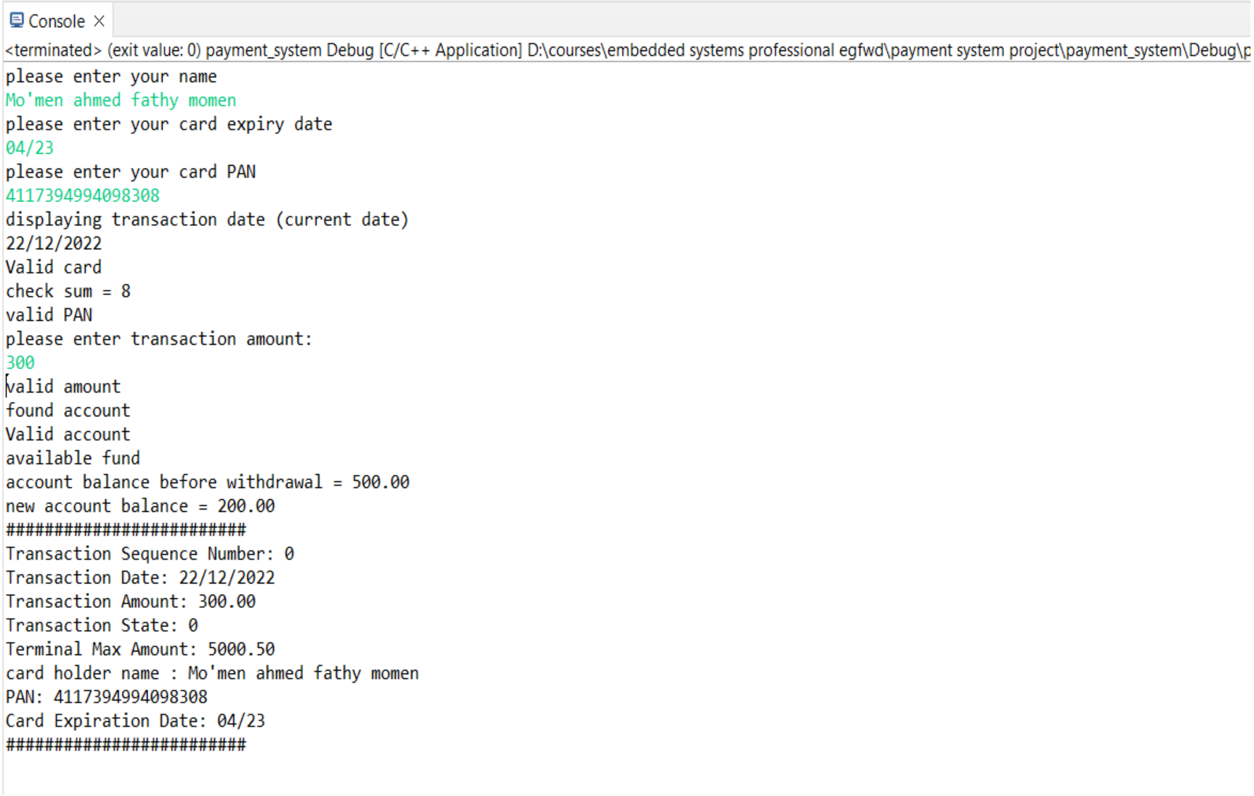
Header file:



```c
/*****************************************************************************
 *
 * Module: APP
 *
 * File Name: App.h
 *
 * Description: header file for the card module
 *
 * Author: Mo'men Ahmed
 *
 *****************************************************************************
#ifndef APP_H_
#define APP_H_

void appStart(void);

#endif /* APP_H_ */

```

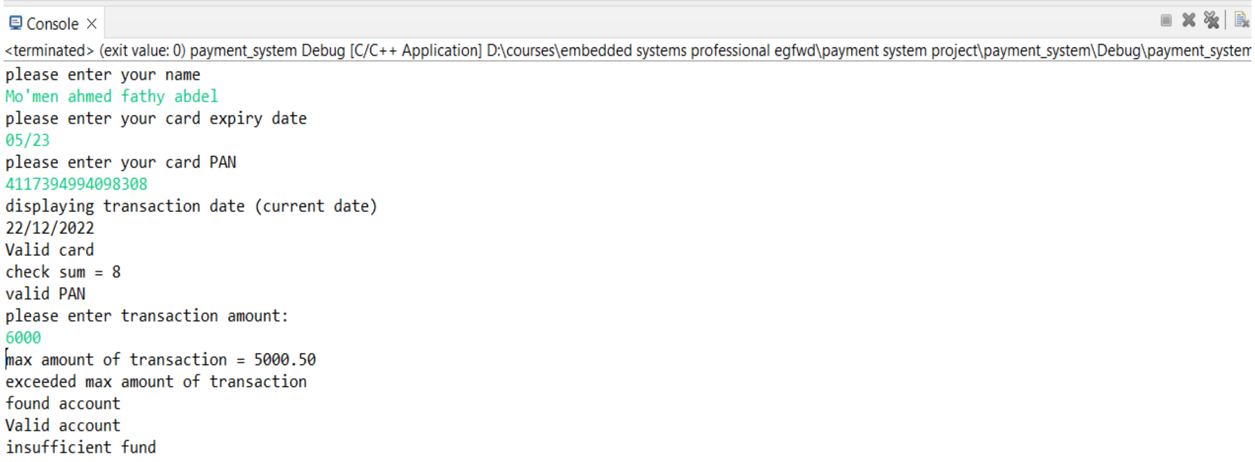
Test cases:

1-approved transaction:



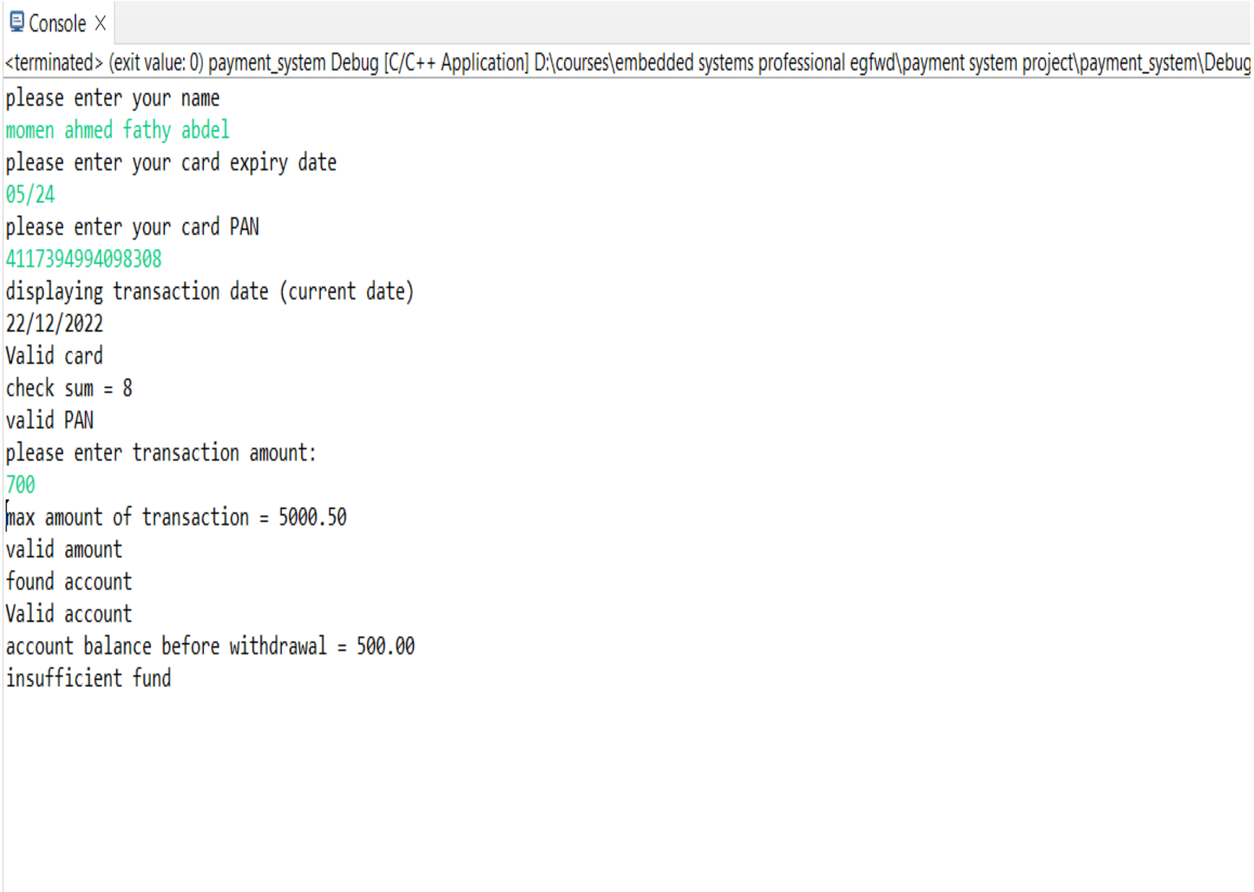
```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\payment_system\Debug\p
please enter your name
Mo'men ahmed fathy momen
please enter your card expiry date
04/23
please enter your card PAN
4117394994098308
displaying transaction date (current date)
22/12/2022
Valid card
check sum = 8
valid PAN
please enter transaction amount:
300
valid amount
found account
Valid account
available fund
account balance before withdrawal = 500.00
new account balance = 200.00
#########################
Transaction Sequence Number: 0
Transaction Date: 22/12/2022
Transaction Amount: 300.00
Transaction State: 0
Terminal Max Amount: 5000.50
card holder name : Mo'men ahmed fathy momen
PAN: 4117394994098308
Card Expiration Date: 04/23
#########################
```

2- exceeded the max amount of transaction case:



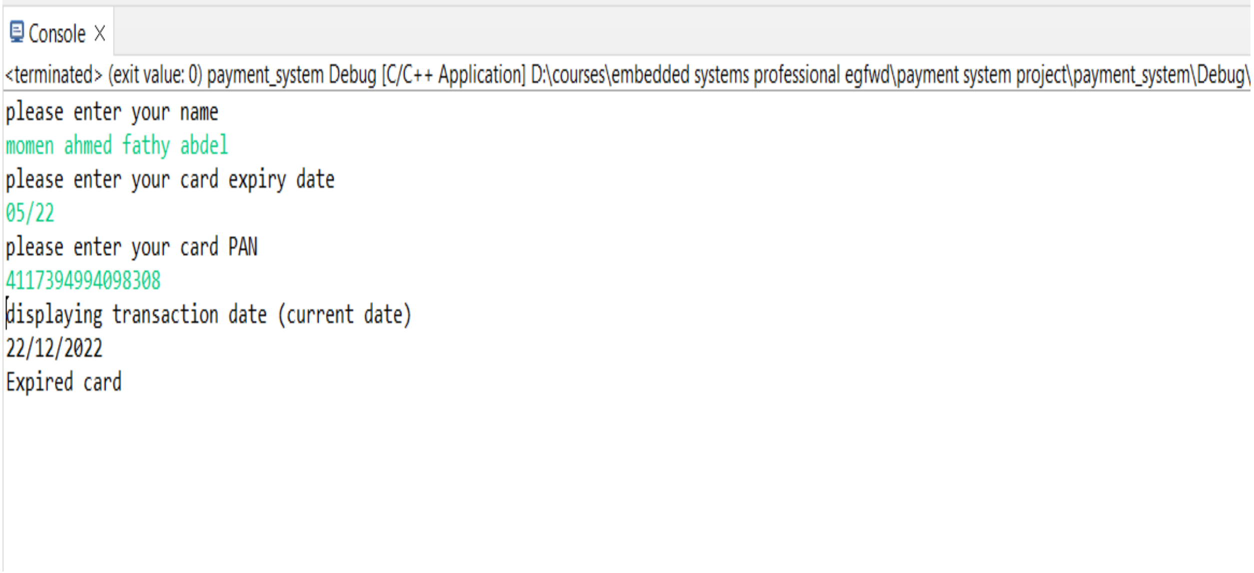
```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\payment_system\Debug\payment_system
please enter your name
Mo'men ahmed fathy abdel
please enter your card expiry date
05/23
please enter your card PAN
4117394994098308
displaying transaction date (current date)
22/12/2022
Valid card
check sum = 8
valid PAN
please enter transaction amount:
6000
max amount of transaction = 5000.50
exceeded max amount of transaction
found account
Valid account
insufficient fund
```

3- insufficient fund case:



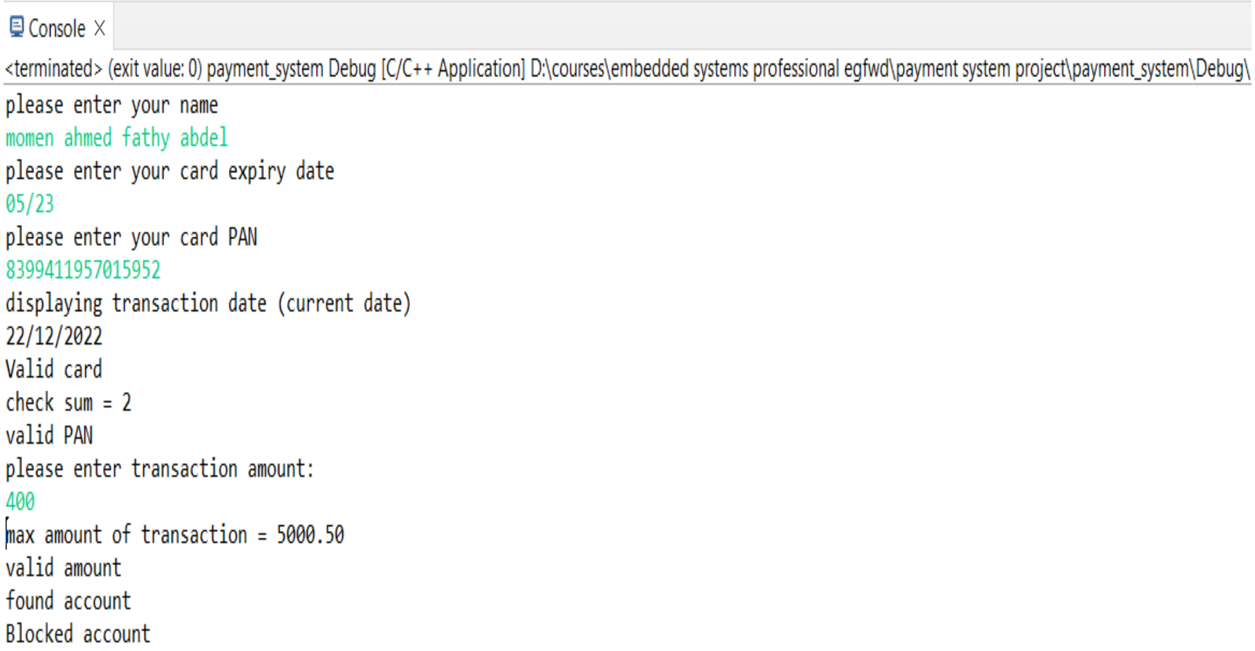
```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\payment_system\Debug
please enter your name
momen ahmed fathy abdel
please enter your card expiry date
05/24
please enter your card PAN
4117394994098308
displaying transaction date (current date)
22/12/2022
Valid card
check sum = 8
valid PAN
please enter transaction amount:
700
max amount of transaction = 5000.50
valid amount
found account
Valid account
account balance before withdrawal = 500.00
insufficient fund
```

4- expired card case:



```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\payment_system\Debug\
please enter your name
momen ahmed fathy abdel
please enter your card expiry date
05/22
please enter your card PAN
4117394994098308
displaying transaction date (current date)
22/12/2022
Expired card
```

5- stolen card case (blocked):



```
<terminated> (exit value: 0) payment_system Debug [C/C++ Application] D:\courses\embedded systems professional egfwd\payment system project\payment_system\Debug\
please enter your name
momen ahmed fathy abdel
please enter your card expiry date
05/23
please enter your card PAN
8399411957015952
displaying transaction date (current date)
22/12/2022
Valid card
check sum = 2
valid PAN
please enter transaction amount:
400
max amount of transaction = 5000.50
valid amount
found account
Blocked account
```