**LAB REPORT #10**

**C++ Functions (Value &Reference Parameters)**

Name: Owais Rao

Roll No: 22L-7638

Class: BSEE-1A2

**Introduction:-**

In C++, there are two types of function parameters:**(a) value parameters, and (b) reference parameters**. Value parameters are used to pass information into a function. A void function with value parameters is declared by enclosing the list of types for the parameter list in the parentheses. Reference parameters are used to pass information in and out of a function. They have the ampersand (&) following their type identifier in the function prototype and function heading.

**Objective:-**

* To get an understanding of reference parameters.
* To learn the difference between value and reference parameters.

**Procedure:-**

With the help of lab manual, I was able to write codes for given exercises. They are as follows with their outputs:-

**Exercise 1:-**

#include<iostream>

using namespace std;

void InputCardDetails(int& cardNumber, double& initialBalance, char& typeOfCard)

{

char a;

int b;

cout << "Enter Card Number(10 digits): ";

cin >> b;

if (b > 999999999 && b < 10000000000)

cardNumber = b;

else

{

cout << "Invalid. Enter 10 digit Card number.";

cout << endl << "Enter Card Number(10 digits): ";

cin >> cardNumber;

}

cout << "Enter Initial Balance: Rs.";

cin >> initialBalance;

cout << "Enter Type of Card[Gold(G) or Silver(S)]: ";

cin >> a;

if (a == 'G' || a == 'S')

typeOfCard = a;

else

{

cout << "Invalid Card Type." ;

cout << endl << "Enter Type of Card[Gold(G) or Silver(S)]: ";

cin >> typeOfCard;

}

}

void main()

{

int cardNumber;

double initialBalance;

char typeOfCard;

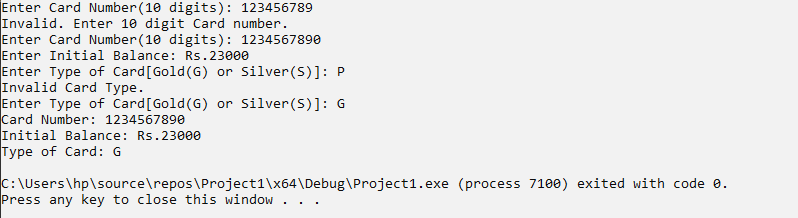
InputCardDetails(cardNumber, initialBalance, typeOfCard);

cout << "Card Number: " << cardNumber << endl;

cout << "Initial Balance: Rs." << initialBalance << endl;

cout << "Type of Card: " << typeOfCard << endl;

}

****

**Exercise 2:-**

#include<iostream>

using namespace std;

void InputCardDetails(int& cardNumber, double& initialBalance, char& typeOfCard)

{

char a;

int b;

cout << "Enter Card Number(10 digits): ";

cin >> b;

if (b > 999999999 && b < 10000000000)

cardNumber = b;

else

{

cout << "Invalid. Enter 10 digit Card number.";

cout << endl << "Enter Card Number(10 digits): ";

cin >> cardNumber;

}

cout << "Enter Initial Balance: Rs.";

cin >> initialBalance;

cout << "Enter Type of Card[Gold(G) or Silver(S)]: ";

cin >> a;

if (a == 'G' || a == 'S')

typeOfCard = a;

else

{

cout << "Invalid Card Type." ;

cout << endl << "Enter Type of Card[Gold(G) or Silver(S)]: ";

cin >> typeOfCard;

}

}

void WithdrawMoney(double& initialBalance, char& typeOfCard, double& withdrawAmount, double& currentBalance)

{

int option;

cout << "\nChoose amount to be withdrawn:- \n\t 1)500 \n\t 2)1000 \n\t 3)2500 \n\t 4)5000 \n\t 5)10000 \n\t 6)Other" << endl;

cin >> option;

switch (option)

{

case 1:

if (500 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.500" << endl;

currentBalance = initialBalance - 500;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 2:

if (1000 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.1000" << endl;

currentBalance = initialBalance - 1000;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 3:

if (2500 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.2500" << endl;

currentBalance = initialBalance - 2500;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 4:

if (5000 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.5000" << endl;

currentBalance = initialBalance - 5000;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 5:

if (10000 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.10000" << endl;

currentBalance = initialBalance - 10000;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 6:

cout << "\nEnter amount to be withdrawn: ";

cin >> withdrawAmount;

if (typeOfCard == 'G')

{

if (withdrawAmount <= 25000 && withdrawAmount <= initialBalance)

{

cout << "\nMoney withdrawn: " << withdrawAmount << endl;

currentBalance = initialBalance - withdrawAmount;

break;

}

else

cout << "\nMoney withdrawn cannot be more than Rs.25000 and your initial balance " << initialBalance << endl;

}

else if (typeOfCard == 'S')

{

if (withdrawAmount <= 10000 && withdrawAmount <= initialBalance)

{

cout << "\nMoney withdrawn: " << withdrawAmount << endl;

currentBalance = initialBalance - withdrawAmount;

break;

}

else

cout << "\nMoney withdrawn cannot be more than Rs.10000 and your initial balance " << initialBalance << endl;

}

break;

}

}

void main()

{

int cardNumber;

double initialBalance, withdrawAmount, currentBalance;

char typeOfCard, x;

InputCardDetails(cardNumber, initialBalance, typeOfCard);

WithdrawMoney(initialBalance, typeOfCard, withdrawAmount, currentBalance);

cout << "\nTo exit the menu, enter EXIT(E).Otherwise enter CONTINUE(C): ";

cin >> x;

while (x != 'E')

{

WithdrawMoney(initialBalance, typeOfCard, withdrawAmount, currentBalance);

cout << "\nTo exit the menu, enter EXIT(E).Otherwise enter CONTINUE(C): ";

cin >> x;

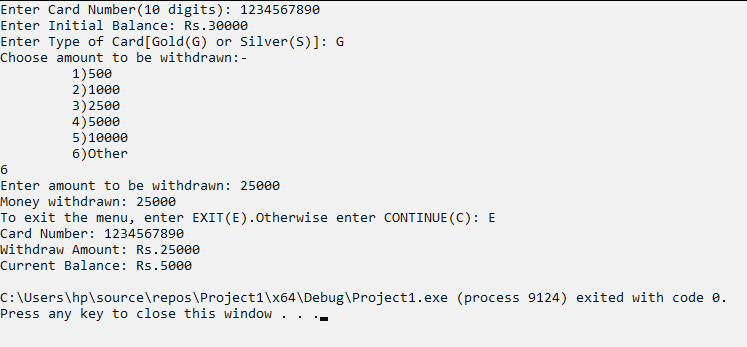
}

cout << "\nCard Number: " << cardNumber << endl;

cout << "\nWithdraw Amount: Rs." << withdrawAmount << endl;

cout << "\nCurrent Balance: Rs." << currentBalance << endl;

}

****

**Exercise 3:-**

#include<iostream>

using namespace std;

void InputCardDetails(int& cardNumber, double& initialBalance, char& typeOfCard)

{

char a;

int b;

cout << "Enter Card Number(10 digits): ";

cin >> b;

if (b > 999999999 && b < 10000000000)

cardNumber = b;

else

{

cout << "Invalid. Enter 10 digit Card number.";

cout << endl << "Enter Card Number(10 digits): ";

cin >> cardNumber;

}

cout << "Enter Initial Balance: Rs.";

cin >> initialBalance;

cout << "Enter Type of Card[Gold(G) or Silver(S)]: ";

cin >> a;

if (a == 'G' || a == 'S')

typeOfCard = a;

else

{

cout << "Invalid Card Type." ;

cout << endl << "Enter Type of Card[Gold(G) or Silver(S)]: ";

cin >> typeOfCard;

}

}

void WithdrawMoney(double& initialBalance, char& typeOfCard, double& withdrawAmount, double& currentBalance)

{

int option;

cout << "\nChoose amount to be withdrawn:- \n\t 1)500 \n\t 2)1000 \n\t 3)2500 \n\t 4)5000 \n\t 5)10000 \n\t 6)Other" << endl;

cin >> option;

switch (option)

{

case 1:

if (500 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.500" << endl;

currentBalance = initialBalance - 500;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 2:

if (1000 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.1000" << endl;

currentBalance = initialBalance - 1000;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 3:

if (2500 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.2500" << endl;

currentBalance = initialBalance - 2500;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 4:

if (5000 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.5000" << endl;

currentBalance = initialBalance - 5000;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 5:

if (10000 <= initialBalance)

{

cout << "\nMoney withdrawn = Rs.10000" << endl;

currentBalance = initialBalance - 10000;

}

else

{

cout << "\nAmount to be withdrawn should be less than " << initialBalance;

}

break;

case 6:

cout << "\nEnter amount to be withdrawn: Rs.";

cin >> withdrawAmount;

if (typeOfCard == 'G')

{

if (withdrawAmount <= 25000 && withdrawAmount <= initialBalance)

{

cout << "\nMoney withdrawn: Rs." << withdrawAmount << endl;

currentBalance = initialBalance - withdrawAmount;

break;

}

else

cout << "\nMoney withdrawn cannot be more than Rs.25000 and your initial balance " << initialBalance << endl;

}

else if (typeOfCard == 'S')

{

if (withdrawAmount <= 10000 && withdrawAmount <= initialBalance)

{

cout << "\nMoney withdrawn: Rs." << withdrawAmount << endl;

currentBalance = initialBalance - withdrawAmount;

break;

}

else

cout << "\nMoney withdrawn cannot be more than Rs.10000 and your initial balance " << initialBalance << endl;

}

break;

}

}

void BalanceInquiry(int& cardNumber, double& currentBalance)

{

cout << "\nCard Number: " << cardNumber;

cout << "\nCurrent Balance: Rs." << currentBalance << endl;

}

void TransferMoney(int& accountNumber, double& currentBalance, double& transferAmount)

{

int b;

cout << "\nEnter Account Number to transfer money(10 digit): ";

cin >> b;

if (b > 999999999 && b < 10000000000)

accountNumber = b;

else

{

cout << "Invalid. Enter 10 digit Account number.";

cout << endl << "Enter Account Number(10 digits): ";

cin >> accountNumber;

}

cout << "Enter amount you wish to transfer: Rs.";

cin >> transferAmount;

if (transferAmount <= currentBalance)

{

cout << transferAmount << "-/Rs transfered to account number " << accountNumber;

}

else

{

cout << "Transfer amount cannot be more than Rs." << currentBalance;

}

currentBalance = currentBalance - transferAmount;

cout << "\nCurrent Balance: Rs." << currentBalance;

}

void main()

{

int cardNumber, accountNumber;

double initialBalance, withdrawAmount, currentBalance, transferAmount;

char typeOfCard, x;

InputCardDetails(cardNumber, initialBalance, typeOfCard);

WithdrawMoney(initialBalance, typeOfCard, withdrawAmount, currentBalance);

cout << "\nTo exit the menu, enter EXIT(E).Otherwise enter CONTINUE(C): ";

cin >> x;

while (x != 'E')

{

WithdrawMoney(initialBalance, typeOfCard, withdrawAmount, currentBalance);

cout << "\nTo exit the menu, enter EXIT(E).Otherwise enter CONTINUE(C): ";

cin >> x;

}

BalanceInquiry(cardNumber, currentBalance);

TransferMoney(accountNumber, currentBalance, transferAmount);

cout << "\n\nTo exit the menu, enter EXIT(E).Otherwise enter CONTINUE(C): ";

cin >> x;

while (x != 'E')

{

TransferMoney(accountNumber, currentBalance, transferAmount);

cout << "\n\nTo exit the menu, enter EXIT(E).Otherwise enter CONTINUE(C): ";

cin >> x;

}

cout << "\nCard Number: " << cardNumber;

cout << "\nCard Type: " << typeOfCard;

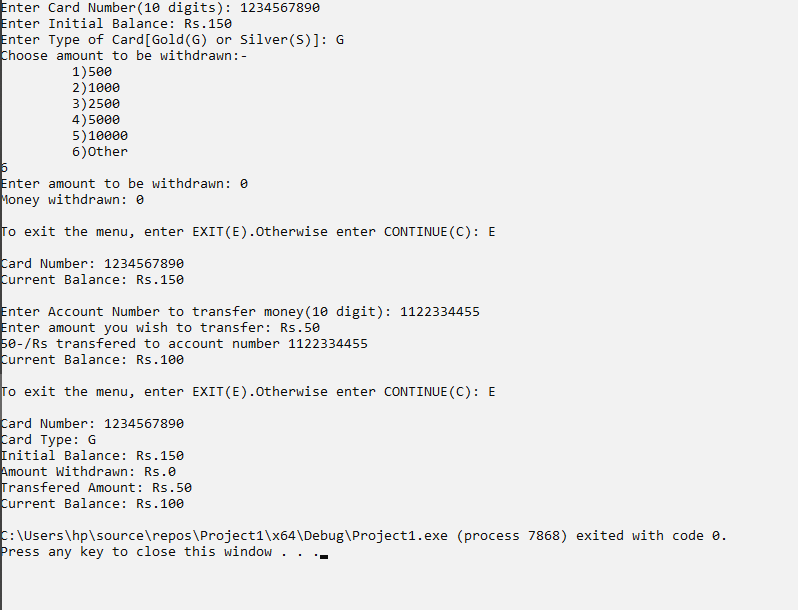
cout << "\nInitial Balance: Rs." << initialBalance;

cout << "\nAmount Withdrawn: Rs." << withdrawAmount;

cout << "\nTransfered Amount: Rs." << transferAmount;

cout << "\nCurrent Balance: Rs." << currentBalance << endl;

}

****

**Issues:-**

No issues were faced.

**Conclusion:-**

* I was able to get an understanding of reference parameters.
* I was able to learn the difference between value and reference parameters.

**Applications:-**

* Functions are used to improve the readability and reusability of the code.
* References are used to modify the passed parameters in a function.