Semester 1st

**Pragramming Fundamental Lab**



**Lab # Semester Project**

* **Group Memebers :**
* Ahmed Hassan (245212)
* Muhammad Raza Meer Hayat(245226)
* Mohammad Saif Ali Khan (245150)

* Submitted To: Sir.Mansoor

**DEPARMENT OF COMPUTER SCIENCE**

AIR UNIVERSITY, AEROSPACE AND AVIATION CAMPUS KAMRA

# Bank Management System Code Explanation

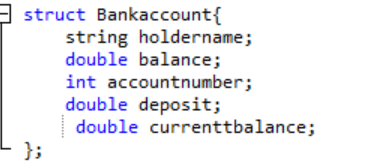
## Introduction

This program is a simple Bank Management System implemented in C++. It allows users to create a bank account, deposit cash, withdraw cash, and view account details. The program utilizes functions and a structure to organize data and operations.

## 2. Code Structure

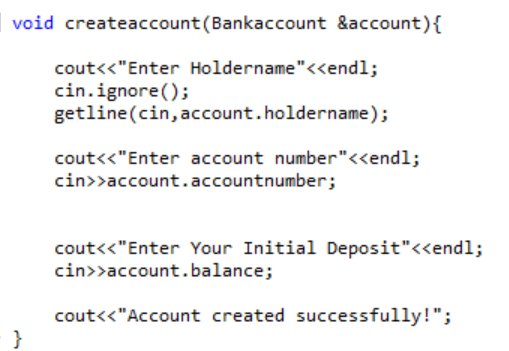
### a. Struct Definition

The structure Bank-account is used to define the structure of a bank account. It includes the following fields:  
- `holder-name`: A string to store the name of the account holder.  
- `balance`: A double to store the account balance.  
- `account-number`: An integer to store the account number.  
- `deposit` and `withdraw`: Doubles for deposit and withdrawal operations (not directly used).  
- `counterbalance`: A double for current balance (not directly used).

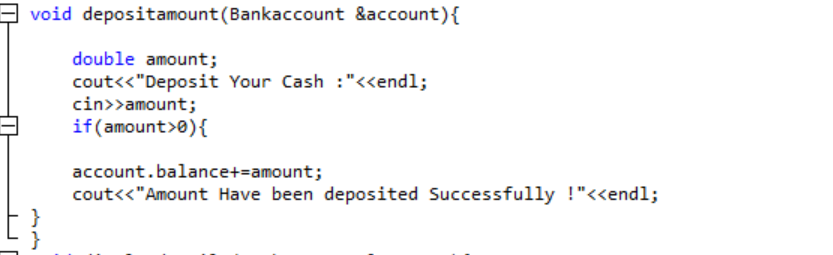


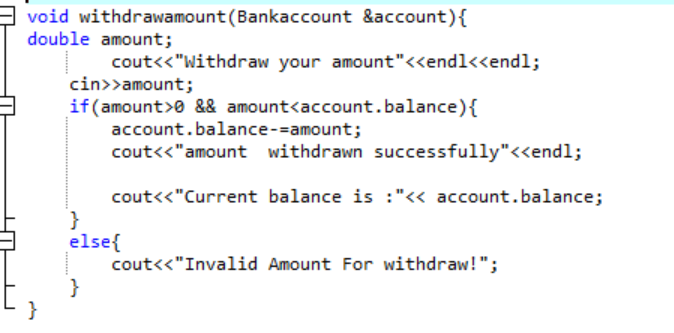
### b. Functions

1. `create account`: Prompts the user to input the account holder's name, account number, and initial deposit. It initializes the `Bank-account` structure.



2. `deposit-amount`: Allows the user to deposit cash into their account. The balance is updated accordingly.

  
3. `withdraw-amount`: Allows the user to withdraw cash from their account. It validates the withdrawal amount to ensure it doesn't exceed the balance.



4. `display-details`: Displays the account holder's name, account number, and current balance.

### alldetails

### c. Main Function

The `main` function acts as the entry point of the program. It provides a menu-driven interface for users to interact with the system. The user can select options to create an account, deposit, withdraw, or view details. The program runs in a loop until the user chooses to exit.

## 3. Features

The program includes the following features:  
- Input validation for deposit and withdrawal operations.  
- A user-friendly menu-driven interface.  
- Separation of concerns using functions for modular code.

## 4. Limitations

1. The program handles only one bank account at a time.  
2. No persistent storage: Data is lost when the program exits.  
3. Limited error handling: User input is not fully sanitized.

## 5. Conclusion

This Bank Management System is a simple implementation of basic banking operations. It demonstrates the use of structures, functions, and a menu-driven interface in C++. Future improvements can include support for multiple accounts, persistent data storage, and enhanced error handling.