

Mini – C++ Project

ON

# ATM Banking System

**A Mini Project Submitted By:**

|             |            |
|-------------|------------|
| Dhanush R.  | 4NM21AI024 |
| Ananya D.   | 4NM21AI013 |
| Nihal Mohan | 4NM21AI045 |

**Under The Guidance Of**

***Mr. Mahesh B.L.***

*Assistant Professor*

**Department Of Artificial Intelligence and Machine Learning**

*In partial fulfilment of the requirements for*  
***Programming in C++ with Examples – 21AM01***

**NMAM Institute of Technology**  
**Nitte - 574110**



**NMAM INSTITUTE  
OF TECHNOLOGY**

Nitte (DU) established under Section 3 of UGC Act 1956 | Accredited with 'A+' Grade by NAAC



**NITTE**  
EDUCATION TRUST

**N.M.A.M. INSTITUTE OF TECHNOLOGY**  
(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi)  
Nitte – 574 110, Karnataka, India

ISO 9001:2015 Certified), Accredited with 'A' Grade by NAAC

## CERTIFICATE

*Certified that the mini project work entitled*

***"ATM Banking System"***

*is a bona fide work carried out by*

***Nihal Mohan Shettigar***

***(4NM21AI045)***

***Dhanush Rajashekar***

***(4NM21AI045)***

***Ananya D***

***(4NM21AI013)***

*in partial fulfilment of the requirements for the award of*

***Bachelor of Engineering Degree in Artificial Intelligence and Machine Learning Engineering***

*prescribed by Visvesvaraya Technological University, Belgaum*

*during the year 2022-2023.*

*It is certified that all corrections/suggestions indicated for Internal Assessment have been*

*incorporated in the report deposited in the departmental library.*

*The mini project report has been approved as it satisfies the academic requirements in respect of the*

*mini project work prescribed for the Bachelor of Engineering Degree.*

Signature of Guide

Signature of HOD

### Evaluation

Name of the Examiners

Signature with Date

1. MAHESH B.L.

2. Sudesh Rao

## **ACKNOWLEDGEMENT**

We believe that our mini project will be complete only after we thank the people who have contributed to make this mini project successful.

First and foremost, our sincere thanks to our beloved principal, **Dr. Niranjan N. Chiplunkar** for giving us an opportunity to carry out our mini project work at our college and providing us with all the needed facilities.

I acknowledge the support and valuable inputs given by, **Dr. Sharada U Shenoy** the Head of the Department, Artificial Intelligence and Machine Learning Engineering, NMAMIT, Nitte.

We express our deep sense of gratitude and indebtedness to our guide **Mr. Mahesh B.L., Assistant Professor** Artificial Intelligence and Machine Learning Engineering, for his inspiring guidance, constant encouragement, support and suggestions for improvement during the course for our mini project.

We also thank all those who have supported us throughout the entire duration of our mini project.

Finally, we thank the staff members of the Department of Artificial Intelligence and Machine Learning Engineering and all our friends for their honest opinions and suggestions throughout the course of our mini project.

**Nihal Mohan Shettigar**

**Dhanush Rajashekar**

**Ananya D.**

## **TABLE OF CONTENTS**

|                                    |              |
|------------------------------------|--------------|
| Title .....                        | <b>1</b>     |
| Certificate .....                  | <b>2</b>     |
| Acknowledgement .....              | <b>3</b>     |
| Table Of Contents .....            | <b>4</b>     |
| Abstract .....                     | <b>5</b>     |
| Introduction .....                 | <b>6-12</b>  |
| ▪ What is C++?                     |              |
| ▪ Why C++?                         |              |
| ▪ C++ Examples                     |              |
| ▪ What's it about?                 |              |
| ▪ Header Files and Functions Used? |              |
| Implementation Of Code .....       | <b>13-16</b> |
| Output .....                       | <b>17-18</b> |
| System Requirements .....          | <b>19</b>    |
| References .....                   | <b>20</b>    |

## **ABSTRACT**

In this program, a Banking/ATM system simulation is created using C++. The system allows for the creation of bank accounts, deposits, withdrawals, and balance inquiries. The program utilizes concepts of Structures to represent the bank accounts, and allows for the storage of account information for the temporary period of time, until the program is being executed. The program also includes basic error handling for invalid account numbers and insufficient funds. Overall, the program demonstrates the use of basic C++ concepts in the simulation of a simple banking system.

The ATM System is a project which is used to access their bank accounts in order to make necessary transactions. Whenever

user want to make these transactions, they can enter their Account Number and verified PIN, and perform these tasks.



Bank Management System  
using

**C++**

# INTRODUCTION

## What is C++?

C++ is a general-purpose programming language that was developed in the early 1980s by **Bjarne Stroustrup** at Bell Labs. It is an extension of the C programming language and provides object-oriented features such as classes, inheritance, and polymorphism. C++ also supports low-level memory manipulation, making it a popular choice for systems programming and the development of operating systems, device drivers, and embedded systems.

C++ is a compiled language, meaning that the source code is translated into machine code by a compiler before it can be executed. This makes C++ programs faster and more efficient than interpreted languages such as Python or JavaScript. C++ also has a large and active community, with a wide range of libraries and frameworks available for use. This makes it a popular choice for many types of software development, including game development, financial modeling, and scientific simulations.

C++ has a steeper learning curve than some other programming languages, due to its more complex syntax and the need for manual memory management. However, its powerful features and performance make it a valuable tool for experienced programmers. C++ is also widely used in the industry, with companies such as Microsoft, Google, and Amazon using it to develop many of their products and services.

In summary, C++ is a powerful and versatile programming language that is widely used in industry for a variety of applications. It has a steep learning curve, but its features and performance make it a valuable tool for experienced programmers. It also has a standardization process which ensures that the language is standardized across the globe.

- C++ is a cross-platform language that can be used to create high-performance applications

- Gives programmers a high level of control over system resources and memory
- A middle-level language rendering it the advantage of programming

## **Why C++?**

C++ is a powerful, high-performance programming language that is widely used in the software industry. One of its main advantages is its ability to create efficient, low-level systems and applications, such as operating systems, device drivers, and embedded systems. C++ provides a level of control over system resources that is not possible with other languages, making it a popular choice for system-level programming.

C++ also supports object-oriented programming, which allows for the creation of reusable, modular code. This can lead to more efficient and maintainable code bases, as well as making it easier to build large, complex systems. Additionally, C++ supports generic programming through templates, which allows for the creation of code that can work with a wide variety of data types.

C++ also has a large and active community, which has led to the development of many libraries and frameworks that can be used to accelerate development and reduce the amount of code that needs to be written. This community also helps to ensure that C++ remains a relevant and up-to-date language, as new features and improvements are continually being added.

Finally, C++ has a relatively low-level memory model, which allows for fine-grained control over memory allocation and management. This can be useful in situations where performance and memory usage are critical, such as in embedded systems and high-performance computing.

Overall, C++ is a versatile, powerful, and efficient programming language that is well-suited for a wide variety of programming tasks. Its ability to create low-level systems and applications, support for

object-oriented and generic programming, large community, and fine-grained control over memory make it a popular choice among developers.

- It is one of the most popular programming languages.
- It is found in many of today's operating systems, graphical interfaces and embedded systems.
- It is an object-oriented programming language which gives a clear structure to programs and allows the code to be reused, lowering development costs.
- It is portable and can be used to develop applications that can be adapted to multiple platforms.

## **C++ Examples**

C++ is a great choice for:

- **Operating Systems:** With the low-level capabilities of C++, developers can structure their code to make even the smallest details of an operating system fast and energy efficient.
- **Game Development:** It's been used to create games, such as Counter-Strike, game engines like Unreal Engine, and gaming consoles, including Xbox, PlayStation, and Nintendo Switch.
- **IoT Devices** IoT (Internet of Things) devices include embedded systems that often rely on C++
- **Databases** C++ is used to build popular database tools like MySQL and MongoDB.
- **Web Browsers** C++ plays a role in web browsers, such as Google Chrome, Mozilla Firefox, Safari, and Opera. It is used to develop back-end services that retrieve information from databases and render code into interactive web pages.

## **What's it About?**

The automated teller machine (ATM) is an automatic banking machine which allows the user to complete basic transactions without any help of bank representatives. In earlier years all the transactions



were to be done manually, still done but very rarely, as it is very difficult task. so now banks use this to give their customers to have easy and faster transactions. This makes transactions easier faster.

Automated Teller Machine enables the clients of a bank to have access to their account without going to the bank. The program is designed in such a way that the user has to enter the card and pin number. Once verified, he is provided a menu and he/she had to enter the option provided in the menu, by which he can further access the needed option to be accessed.

So, in this program, we have used the concepts of **Structures** through which we have implemented this program. The user of this program is prompted to create an account first, after which he/she is prompted to enter a security pin for the purpose of Authentication. The account number generated will be a random number generated between 1000 – 9999 using the rand() function included in the libraries.

Once the account is created, the user can access the various features of the programs, that is

- Withdrawal
- Deposit
- Balance Enquiry

These features, again, work with help of the user entering the his/her appropriate account number, and entering the pin for authentication. Then, he/she performs the needed actions required.

## Header Files and Functions Used:

- **iostream Header File:**

'**iostream**' is a C++ library that provides basic input/output functionality. It contains classes such as '**cin**' and '**cout**' which are used to read from, and write to the standard input and output streams, respectively. The '**iostream**' library also includes other classes such as '**cerr**' and '**clog**' for writing to the standard error stream, and a '**stringstream**' class for reading and writing strings as if they were streams of characters. Additionally, it provides manipulators such as '**endl**' and '**flush**' which can be used to control the formatting and flushing of output streams.

- **Stdlib.h Header File:**

'**stdlib.h**' is a C and C++ standard library header file that provides several functions for performing general functions, including memory allocation, random number generation, and converting strings to numerical values.

Some of the important functions provided by '**stdlib.h**' include:

- '**malloc**' and '**calloc**' for allocating memory dynamically
- '**free**' for releasing dynamically allocated memory
- '**rand**' and '**srand**' for generating random numbers
- '**exit**' and '**abort**' for terminating a program
- '**atoi**', '**atof**', and '**atol**' for converting strings to integers, floating-point numbers, and long integers, respectively
- '**qsort**' to sort an array of elements
- '**bsearch**' to perform binary search on an array of elements.

It's important to note that the '**stdlib.h**' header is a C header and it's recommended to use the C++ version '**cstdlib**' instead in C++ programs.

- **Windows.h Header Files:**

‘**windows.h**’ is a C and C++ header file that provides access to the Microsoft Windows Application Programming Interface (API) and is commonly used for developing Windows-based applications. It contains declarations for functions, data types, and constants that are specific to the Windows operating system.

The ‘**windows.h**’ header file provides access to a wide range of Windows features, including:

- GUI (Graphical User Interface) elements such as windows, buttons, and menus
- Windows messaging and event handling
- File and registry access
- Interprocess communication (IPC)
- Networking
- Threading and synchronization
- DirectInput for gaming input
- DirectDraw for 2D and 3D graphics

It's important to note that ‘**windows.h**’ is specific to Windows operating system and it's not portable to other platforms.

- **Structure Function:**

In C++, a structure (often referred to as a struct) is a user-defined data type that groups together variables of different data types. A struct is similar to a class, but the members of a struct are by default public, whereas the members of a class are private by default.

The syntax for defining a struct is:

```
struct struct_name {  
    type member1;  
    type member2;  
    :  
};
```

For example, you could define a struct called "Person" that has members for a person's name, age, and address like this:

```
struct Person {  
    string name;  
    int age;  
    string address;  
};
```

You can then create variables of the "Person" struct type and set their values like this:

```
Person person1;  
person1.name = "John Smith";  
person1.age = 30;  
person1.address = "123 Main St";
```

## IMPLEMENTATION OF CODE:

```
#include <iostream>
#include <stdlib.h>
#include <windows.h>
using namespace std;
void create();
void deposit();
void withdraw();
void balance_enquiry();

struct bank
{
    int acno;
    int pin;
    char name[50];
    float balance;
}account[100];

int i=0;
int lb_accno=1000,ub_accno=9999;

int main()
{
    int choice;
restart:
    cout<<"\n\n\tATM Banking Program";
    cout<<endl<<"1. Create a New Account";
    cout<<endl<<"2. Cash Deposit";
    cout<<endl<<"3. Cash Withdraw";
    cout<<endl<<"4. Balance Enquiry";
    cout<<endl<<"5. Exit";
    cout<<endl<<"Enter the Choice : ";
    cin>>choice;

    switch (choice)
    {
        case 1:
            create();
            goto restart;

        case 2:
            deposit();
            goto restart;

        case 3:
            withdraw();
            goto restart;

        case 4:
            balance_enquiry();
            goto restart;

        case 5:
            exit(0);

        default:
            cout<<endl<<"Invalid Choice!"<<endl<<endl;
            Sleep(2000);
            goto restart;
    }
}
```

```

}

void create()
{
    cout<<endl<<"\tNEW ACCOUNT CREATION"<<endl;
    int j, pin1, pin2;
    std::srand(time(0));
    j=(rand()%(ub_accno-lb_accno+1))+lb_accno;
    cout<<"Your Account Number is : "<<j;
    account[i].acno=j;

    cout<<endl<<"Enter Your Name : ";
    cin>>account[i].name;
    loop:
    cout<<endl<<"Enter a 4 Digit Pin, to keep Your Account Secure : ";
    cin>>pin1;
    cout<<"Please Verify, by Entering it Again : ";
    cin>>pin2;
    if(pin1!=pin2)
    {
        cout<<"Pin Doesn't Match !";
        Sleep(2000);
        cout<<"\n";
        goto loop;
    }
    else
        account[i].pin=pin2;
    cout<<endl<<"The Minimum Deposit is Rs.500"<<endl;
    cout<<"Rs.500 has been Deposited !"<<endl;
    account[i].balance=500;
    i++;
    Sleep(4000);
}

void deposit()
{
    int acc, b, m=0, pin;
    float dep_amt;
    cout<<endl<<"\tCASH DEPOSIT"<<endl;
    cout<<"Enter Your Account Number : \t";
    cin>>acc;

    for(b=0; b<i; b++)
    {
        if(account[b].acno==acc)
            m=b;
    }

    if(account[m].acno==acc)
    {
        loop:
        cout<<endl<<"Enter Your 4 Digit Pin : ";
        cin>>pin;
        if(pin!=account[m].pin)
        {
            cout<<endl<<"Incorrect Pin !";
            Sleep(2000);
            cout<<"\n";
            goto loop;
        }
    }
}

```

```

        cout<<endl<<"Account Number : "<<account[m].acno;
        cout<<endl<<"Name : "<<account[m].name;
        cout<<endl<<"Current Balance : "<<account[m].balance;
        loop2:
        cout<<endl<<endl<<"Enter the Amount to Deposit : ";
        cin>>dep_amt;
        if(dep_amt<0)
        {
            cout<<endl<<"Invalid Amount Entered !";
            goto loop2;
        }
        account[m].balance = account[m].balance + dep_amt;
        cout<<endl<<"Rs."<<dep_amt<<" Successfully Deposited !";
        cout<<endl<<endl<<"Account Balance After Deposit : "<<account[m].balance;
        Sleep(4000);
    }
    else
    {
        cout<<"The Account Number is Invalid ! ";
        Sleep(4000);
    }
}

void withdraw()
{
    int acc,b,m=0,pin;
    float wit_amt;
    cout<<endl<<"\tCASH WITHDRAWAL";
    cout<<endl<<"Enter Your Account Number : \t";
    cin>>acc;

    for(b=0;b<i;b++)
    {
        if(account[b].acno==acc)
            m=b;
    }

    if(account[m].acno==acc)
    {
        loop:
        cout<<endl<<"Enter Your 4 Digit Pin : ";
        cin>>pin;
        if(pin!=account[m].pin)
        {
            cout<<endl<<"Incorrect Pin !";
            Sleep(2000);
            cout<<"\n";
            goto loop;
        }
        cout<<endl<<"Account Number : "<<account[m].acno;
        cout<<endl<<"Name : "<<account[m].name;
        cout<<endl<<"Current Balance : "<<account[m].balance;
        loop2:
        cout<<endl<<endl<<"Enter the Amount to Withdraw : ";
        cin>>wit_amt;
        if(wit_amt<0)
        {
            cout<<endl<<"Invalid Amount Entered !";
            goto loop2;
        }
    }
}

```

```

    }
    if(account[m].balance<wit_amt+500)
    {
        cout<<"Minimum Account Balance Must Be Rs.500"<<endl;
        cout<<endl<<"Insufficient Funds! Transaction Declined";
        Sleep(4000);
    }
    else
    {
        account[m].balance= account[m].balance - wit_amt;
        cout<<endl<<"Rs."<<wit_amt<<"Successfully Withdrawn !";
        cout<<endl<<endl<<"Account Balance After Withdrawal : "<<account[m].balance
    }
}
else
{
    cout<<endl<<"The Account Number is Invalid !";
    Sleep(4000);
}
}

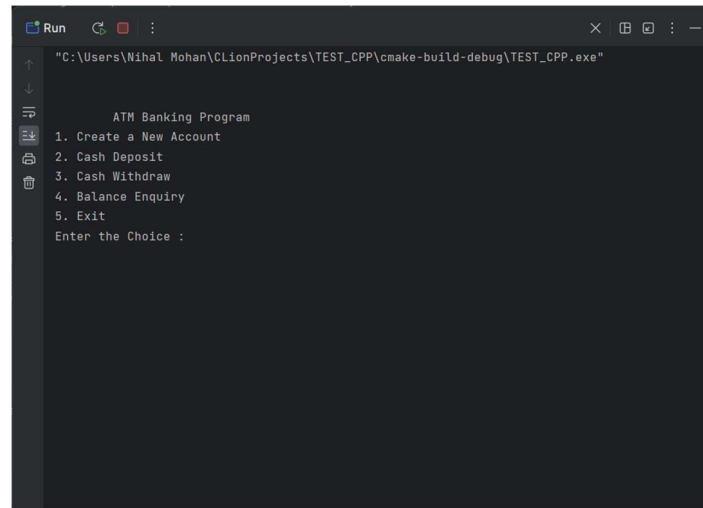
void balance_enquiry()
{
    int acc,b,m=0,pin;
    cout<<endl<<"\tBALANCE ENQUIRY";
    cout<<endl<<"Enter Your Account Number :\t";
    cin>>acc;

    for(b=0;b<i;b++)
    {
        if(account[b].acno==acc)
            m=b;
    }
    if(account[m].acno==acc)
    {
        loop:
        cout<<endl<<"Enter Your 4 Digit Pin : ";
        cin>>pin;
        if(pin!=account[m].pin)
        {
            cout<<endl<<"Incorrect Pin !";
            Sleep(2000);
            cout<<"\n";
            goto loop;
        }
        cout<<endl<<"Account Number : "<<account[m].acno;
        cout<<endl<<"Name : "<<account[m].name;
        cout<<endl<<"Current Balance : "<<account[m].balance;
        Sleep(4000);
    }
    else
    {
        cout<<endl<<"The Account Number is Invalid !";
        Sleep(4000);
    }
}
}

```



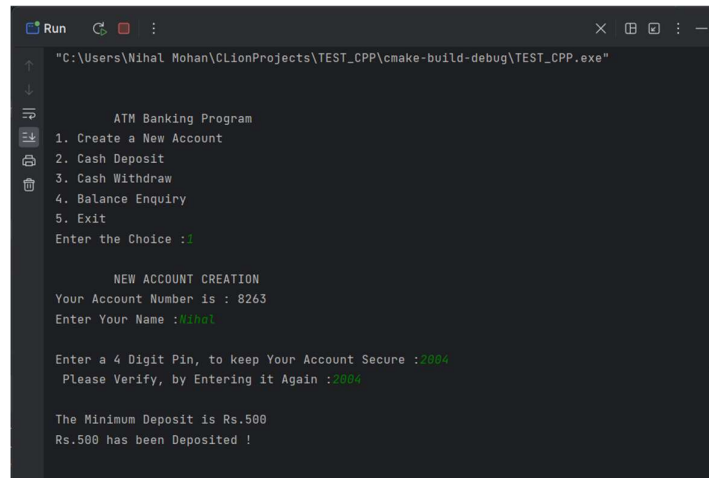
## OUTPUT:



```
Run "C:\Users\Nihal Mohan\CLionProjects\TEST_CPP\cmake-build-debug\TEST_CPP.exe"

ATM Banking Program
1. Create a New Account
2. Cash Deposit
3. Cash Withdraw
4. Balance Enquiry
5. Exit
Enter the Choice :
```

(i) General Interface



```
Run "C:\Users\Nihal Mohan\CLionProjects\TEST_CPP\cmake-build-debug\TEST_CPP.exe"

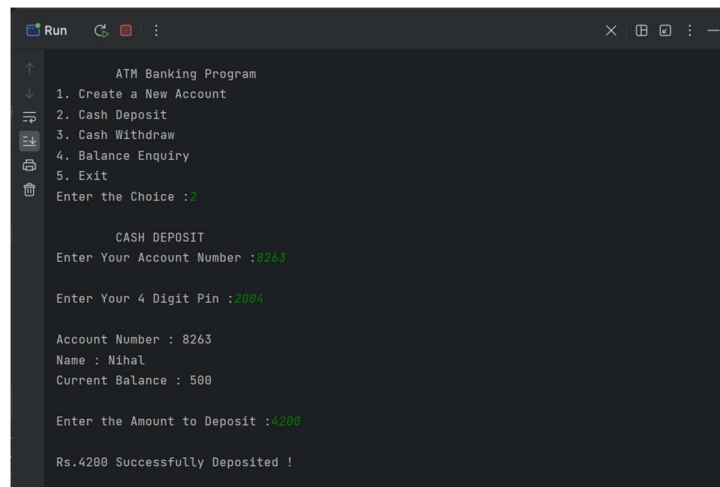
ATM Banking Program
1. Create a New Account
2. Cash Deposit
3. Cash Withdraw
4. Balance Enquiry
5. Exit
Enter the Choice :1

NEW ACCOUNT CREATION
Your Account Number is : 8263
Enter Your Name :Nihal

Enter a 4 Digit Pin, to keep Your Account Secure :2004
Please Verify, by Entering it Again :2004

The Minimum Deposit is Rs.500
Rs.500 has been Deposited !
```

(ii) Account Creation



```
Run "C:\Users\Nihal Mohan\CLionProjects\TEST_CPP\cmake-build-debug\TEST_CPP.exe"

ATM Banking Program
1. Create a New Account
2. Cash Deposit
3. Cash Withdraw
4. Balance Enquiry
5. Exit
Enter the Choice :2

CASH DEPOSIT
Enter Your Account Number :8263

Enter Your 4 Digit Pin :2004

Account Number : 8263
Name : Nihal
Current Balance : 500

Enter the Amount to Deposit :4200

Rs.4200 Successfully Deposited !
```

(iii) Cash Deposit

```
Run  [Icons] :
ATM Banking Program
1. Create a New Account
2. Cash Deposit
3. Cash Withdraw
4. Balance Enquiry
5. Exit
Enter the Choice :3

CASH WITHDRAWAL
Enter Your Account Number :8263

Enter Your 4 Digit Pin :2004

Account Number : 8263
Name : Nihal
Current Balance : 4700

Enter the Amount to Withdraw :300

Rs.300Successfully Withdrawn !

Account Balance After Withdrawal : 4400
```

(iv) Cash Withdrawal

```
Run  [Icons] :
ATM Banking Program
1. Create a New Account
2. Cash Deposit
3. Cash Withdraw
4. Balance Enquiry
5. Exit
Enter the Choice :4

BALANCE ENQUIRY
Enter Your Account Number :8263

Enter Your 4 Digit Pin :2004

Account Number : 8263
Name : Nihal
Current Balance : 4400
```

(v) Balance Enquiry

## **SYSTEM REQUIREMENTS**

### **SOFTWARE:**

- Visual Studio Code / CLion / CodeBlocks, or any IDEs which support C++
- MinGW Compiler
- Necessary Packages / Libraries

### **OS:**

- Windows / Ubuntu / MacOS Based Operating System, whichever's convenient.

### **HARDWARE:**

- 100MB Free Disk space
- RAM: 4GB or above

## **References**

- <https://www.geeksforgeeks.org/c-plus-plus/>
- <https://www.geeksforgeeks.org/structures-in-cpp/>
- <https://www.geeksforgeeks.org/atm-management-system-using-cpp/>
- [https://www.w3schools.com/cpp/cpp\\_structs.asp](https://www.w3schools.com/cpp/cpp_structs.asp)