

Algorithms Level 3



26+ Years
of Experience

PROGRAMMING ADVICES

LEARN THE
RIGHT WAY

Mohammed Abu-Hadhoud

MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITIL®, MCPD, MCSD



حقوق النشر محفوظة، أسعار الكورسات في المنصة هي أسعار
رمزية جدا، ارجو عدم نشر هذه الوثيقة لان نشرها سيمنعنا من
الاستمرار في تقديم العلم للآخرين

ارجو عدم استخدام هذه الوثيقة من غير وجه حق لأنك ستحرم الاف
الناس من التعلم

ProgrammingAdVICES.com



Problem #49/3 Solution Using C++

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
#include <iomanip>

using namespace std;
const string ClientsFileName = "Clients.txt";

struct sClient
{
    string AccountNumber;
    string PinCode;
    string Name;
    string Phone;
    double AccountBalance;
};

vector<string> SplitString(string S1, string Delim)
{
    vector<string> vString;

    short pos = 0;
    string sWord; // define a string variable

    // use find() function to get the position of the delimiters
    while ((pos = S1.find(Delim)) != std::string::npos)
    {
        sWord = S1.substr(0, pos); // store the word
        if (sWord != "")
        {
            vString.push_back(sWord);
        }

        S1.erase(0, pos + Delim.length());
    }

    if (S1 != "")
    {
        vString.push_back(S1); // it adds last word of the string.
    }

    return vString;
}
```



Problem #49/3 Solution Using C++

```
sClient ConvertLinetoRecord(string Line, string Seperator =
"#//#")
{

    sClient Client;
    vector<string> vClientData;

    vClientData = SplitString(Line, Seperator);

    Client.AccountNumber = vClientData[0];
    Client.PinCode = vClientData[1];
    Client.Name = vClientData[2];
    Client.Phone = vClientData[3];
    Client.AccountBalance = stod(vClientData[4]); //cast string to
double

    return Client;
}

vector <sClient> LoadCleintsDataFromFile(string FileName)
{
    vector <sClient> vClients;

    fstream MyFile;
    MyFile.open(FileName, ios::in); //read Mode

    if (MyFile.is_open())
    {
        string Line;
        sClient Client;

        while (getline(MyFile, Line))
        {

            Client = ConvertLinetoRecord(Line);

            vClients.push_back(Client);
        }

        MyFile.close();
    }

    return vClients;
}
```



Problem #49/3 Solution Using C++

```
void PrintClientCard(sClient Client)
{
    cout << "\nThe following are the client details:\n";
    cout << "\nAccount Number: " << Client.AccountNumber;
    cout << "\nPin Code      : " << Client.PinCode;
    cout << "\nName          : " << Client.Name;
    cout << "\nPhone         : " << Client.Phone;
    cout << "\nAccount Balance: " << Client.AccountBalance;
}

bool FindClientByAccountNumber(string AccountNumber, sClient &
Client )
{
    vector <sClient> vClients =
LoadCleintsDataFromFile(ClientsFileName);

    for (sClient C : vClients)
    {
        if (C.AccountNumber == AccountNumber)
        {
            Client = C;
            return true;
        }
    }
    return false;
}

string ReadClientAccountNumber()
{
    string AccountNumber = "";

    cout << "\nPlease enter AccountNumber? ";
    cin >> AccountNumber;
    return AccountNumber;
}
```



Problem #49/3 Solution Using C++

```
int main()
{
    sClient Client;
    string AccountNumber = ReadClientAccountNumber();

    if (FindClientByAccountNumber(AccountNumber, Client))
    {
        PrintClientCard(Client);
    }
    else
    {
        cout << "\nClient with Account Number (" << AccountNumber <<
") is Not Found!";
    }

    system("pause>0");
    return 0;
}
```