PROGRAMMING FOR PROBLEM SOLVING PROJECT REPORT

BANKING MANAGEMENT SYSTEM

Yashowardhan Samdhani
RA2111026010151
CSE w/s in AI & ML
U1 Section

Aditya Krishnan
RA2111026010172
CSE w/s in AI & ML
U1 Section

AIM

The Banking Management System is an application for maintaining an individual's account in a bank. In this project, we have displayed the working of a Bank's management system by covering the basic functionality which:
i) Withdraws money,
ii) Deposits money,
iii) and Views Account Details
as per the requirement of the client. The project uses the C language for complete functionality.

ALGORITHM

Step 1 - Start

Step 2 - Initialize names[], passwords[], debit_card_numbers[], and balance[]

Step 3 - Declare a function void withdraw

Step 3.1 - If flag of User is greater than 3, print Credit transactions unavailable

Step 3.2 - Else

i) Initialize and input amount

ii) If amount is greater than balance[user]-5000, then print “invalid amount”

iii) Else decrement balance[user] by amount

Step 4 - Declare a function void deposit

Step 4.1 - If flag of User is greater than 3, print Debit transactions unavailable

Step 4.2 - Else

i) Initialize and input amount

ii) If amount is greater than 99999, then print “The amount you are trying to deposit is higher than the usual amount.” and “The debit transaction has been flagged.", then increment flag by 1

iii) If flag of User is greater than 3, print “The Debit transaction was unsuccessful”

iv) Else Increment balance[user] by amount

Step 5 - Declare a function void view_details

Step 5.1 - Print name[user], debit_card_numbers[user], and balance[user]

Step 6 - Declare a function int security

Step 6.1 - Declare check = 0, username, debit_card_number and password

Step 6.2 - Input username, debit_card_number and password

Step 6.3 - Use For Loop

i) If username = names[i], debit_card_number = debit_card_numbers[i] and password = passwords[i], increment check by 1

ii) If check is equal to 3, break the loop

Step 6.4 - If Check is less than 3, print “Invalid Username/Password/Debit Card Number"

Step 6.5 - Return -1

Step 6.6 - Else print “Successfully logged in” and Return i

Step 7 - Declare a function void main

Step 7.1 - Initialize cont

Step 7.2 - Call the function Security and assign the returned value to cont

Step 7.3 - If cont equals -1, exit(0)

Step 7.4 - Else run an infinite while loop

i) Print a menu for operations using print statements, with one value for ending the program

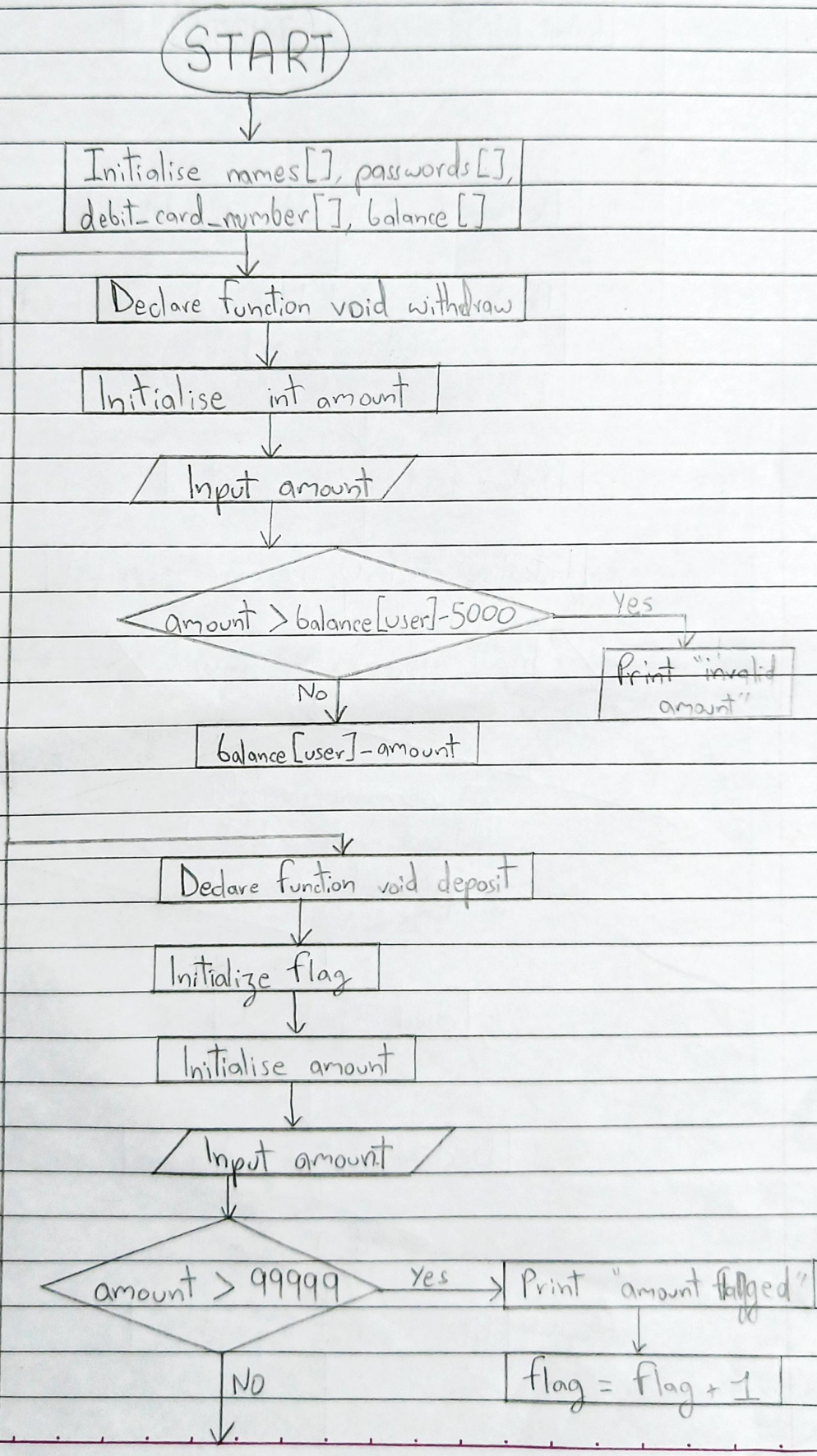
ii) Initialize and input int choice from the user

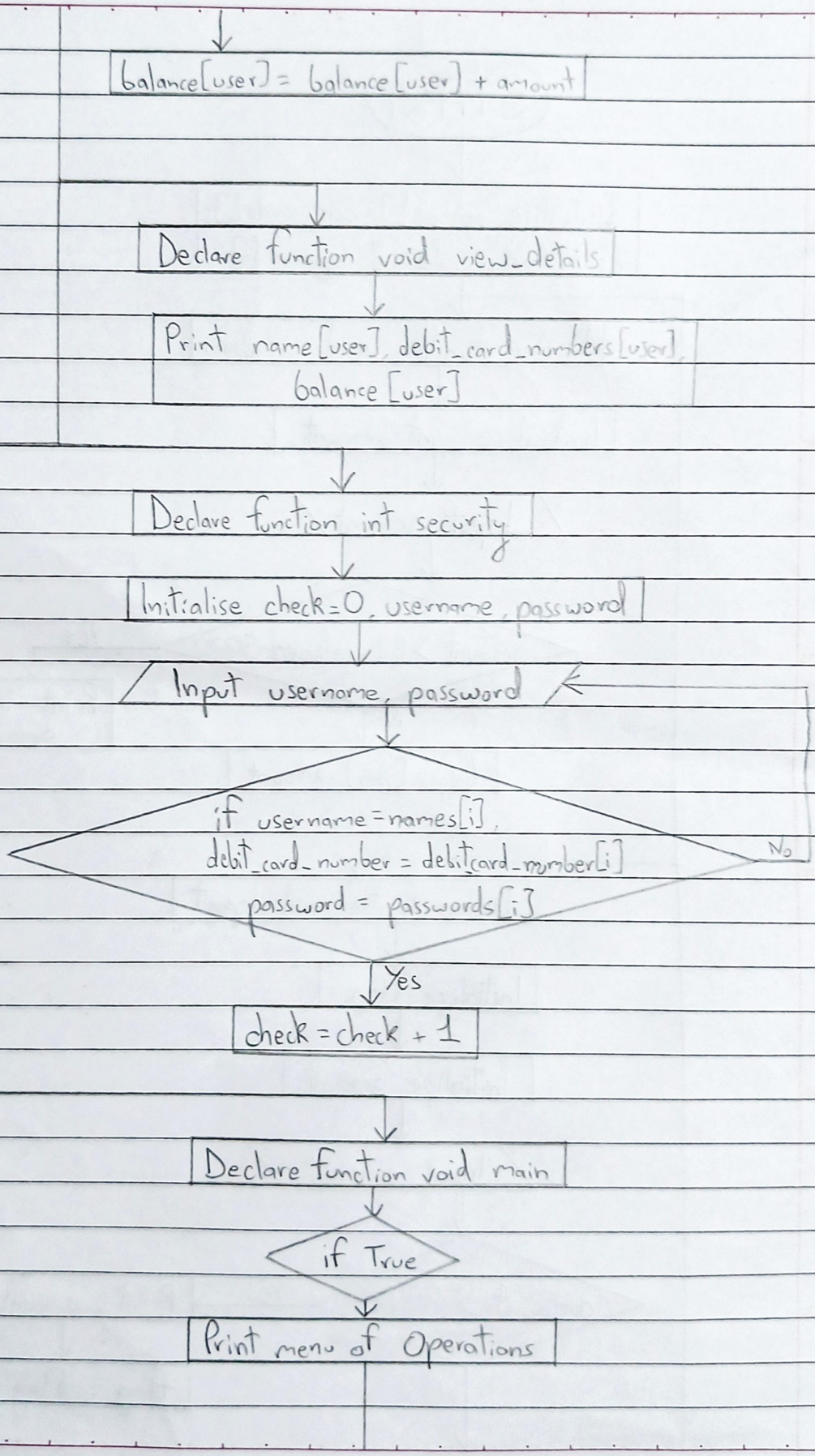
iii) Use switch case to call required function depending upon the value of choice

iv) If the default value is called, then print ”invalid choice!”

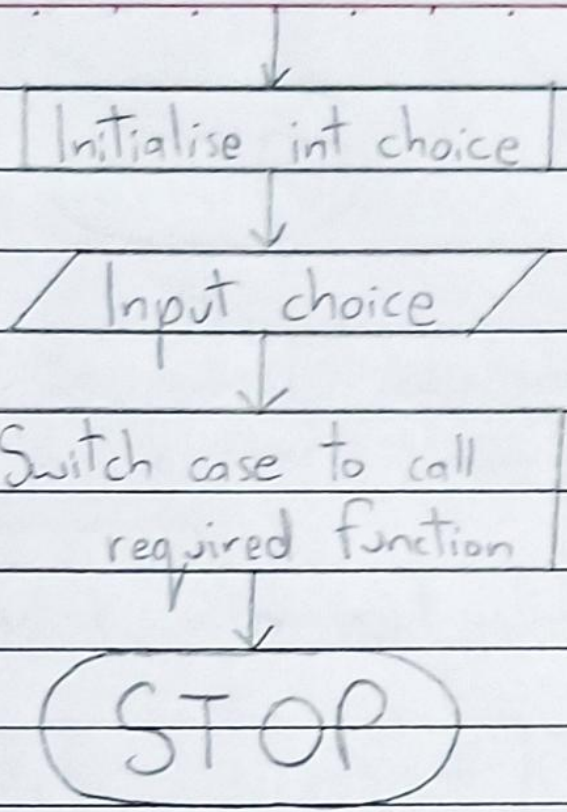
Step 8 - Stop

Date / /


START
Initialise names[], passwords[], debit_card_number[], balance[]
Declare function void withdraw
Initialise int amount
Input amount
amount > balance[user]-5000
Yes
Print "invalid amount"
No
balance[user]-amount
Declare function void deposit
Initialize flag
Initialise amount
Input amount
amount > 99999
Yes
Print "amount flagged"
flag = flag + 1
NO


balance[user] = balance[user] + amount
Declare function void view_details
Print name[user], debit_card_numbers[user], balance[user]
Declare function int security
Initialise check=0, username, password
Input username, password
if username=names[i], debit_card_number = debit_card_number[i] password = passwords[i]
No
Yes
check = check + 1
Declare function void main
if True
Print menu of Operations

Date / /


Initialise int choice
Input choice
Switch case to call required function
STOP

CODE

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

char names[10][100] = {"Yashowardhan Samdhani","Aditya
Krishnan","Siddhanth Vashistha",
        "Bhavishya Pattanayak","Ananya Guatam","Aashriya Bhullar","Abir
Majumder",
        "Amaan Majid","Animesh Gupta","Jaswanth Sunkara"};
int passwords[10] =
{5915,5815,6489,3816,1895,9889,4826,4826,1568,1235}, flag[10];
long int debit_card_numbers[10] =
{687942415726,917429365599,710573661154,971998449229,

792087236336,984068812506,175584159453,959305088852,9593050888
52,
        744717302642}, balance[10] =
{894418,8948,4589,489544,898489,89489,561598,589815,4898516,59189
45};

void withdraw(int user)
{

    if(flag[user]>3)
        printf("Debit transactions unavailable\n\n");
    else
    {
        int amount;
        printf("Input the amount to be withdrawn: ");
        scanf("%d",&amount);

        if(amount > balance[user]-5000)
```

```
            printf("Invalid amount\n\n");
        else
        {
            balance[user] = balance[user] - amount;
            printf("Rs. %d debited.\nNew balance: %ld\n\n", amount,
balance[user]);
        }
    }
}

void deposit(int user)
{
    if(flag[user]>3)
        printf("Credit transactions unavailable\n\n");
    else
    {
        int amount;
        printf("Input the amount to be deposited: ");
        scanf("%d",&amount);
        if(amount > 99999)
        {
            printf("The amount you are trying to deposit is higher than the usual
amount.\n");
            printf("The debit transaction has been flagged.\n\n");
            flag[user]++;
        }

        if(flag[user]>3)
            printf("The Credit transaction was unsuccessful");
        else
        {
            balance[user] = balance[user] + amount;
            printf("Rs. %d Credited.\nNew balance: %ld\n\n", amount,
balance[user]);
        }
```

```
    }
}

void view_details(int user)
{
    long i;
    printf("Printing details...\n\n");
    for(i=0;i<=100000000;i++);
    for(i=0;i<=100000000;i++);

    printf("%s\n", names[user]);
    printf("%ld\n", debit_card_numbers[user]);
    printf("%ld\n\n", balance[user]);
}

int security()
{
    int check = 0, i;
    char username[100];
    long int debit_card_number;
    int password;

    printf("Input Username: ");
    scanf("%[^\n]s",username);
    printf("Input Debit Card Number: ");
    scanf("%ld", &debit_card_number);
    printf("Input Password: ");
    scanf("%d", &password);

    for(i=0;i<10;i++)
    {
        if(strcmp(username,names[i]) == 0)
            check++;
        if(password == passwords[i])
            check++;
```

```
        if(debit_card_number == debit_card_numbers[i])
            check++;
        if(check==3)
            break;
        else
            check = 0;
    }

    if(check < 3)
    {
        printf("Invalid Username/Password/Debit Card Number");
        return -1;
    }
    else
    {
        printf("Successfully logged in\n\n");
        return i;
    }
}

void main()
{
    int cont = security();

    if(cont == -1)
        exit(0);
    else
        while(1)
        {
            printf("----------Menu----------\n");
            printf("1. Withdraw Amount\n");
            printf("2. Deposit Amount\n");
            printf("3. Check Details\n");
            printf("4. Log out and Exit\n");
            printf("------------------------\n");
```

```
        printf("\nEnter your choice: ");
        int choice;
        scanf("%d", &choice);

        switch(choice)
        {
            case 1:
            withdraw(cont);
            break;

            case 2:
            deposit(cont);
            break;

            case 3:
            view_details(cont);
            break;

            case 4:
            printf("Thank You, Hope to see you again next time!");
            exit(0);
            break;

            default:
            printf("Invalid Choice!\n\n");
            int i;
            for(i=0;i<=100000000;i++);
            for(i=0;i<=100000000;i++);
            break;
        }
    }
}
```

SAMPLE INPUT AND OUTPUT

Input Username: Yashowardhan Samdhnai
Input Debit Card Number: 687942415726
Input Password: 5915
Successfully logged in

----------Menu----------
1. Withdraw Amount
2. Deposit Amount
3. Check Details
4. Log out and Exit

Enter your choice: 3
Printing details...

Yashowardhan Samdhani
687942415726
894418

----------Menu----------
1. Withdraw Amount
2. Deposit Amount
3. Check Details
4. Log out and Exit

Enter your choice: 1
Input the amount to be withdrawn: 50000
Rs. 50000 debited.
New balance: 844418

----------Menu----------

1. Withdraw Amount
2. Deposit Amount
3. Check Details
4. Log out and Exit

Enter your choice: 2
Input the amount to be deposited: 6000
Rs. 6000 Credited.
New balance: 850418

----------Menu----------
1. Withdraw Amount
2. Deposit Amount
3. Check Details
4. Log out and Exit

Enter your choice: 2
Input the amount to be deposited: 140000
The amount you are trying to deposit is higher than the usual amount.
The credit transaction has been flagged.

Rs. 140000 Credited.
New balance: 990418

----------Menu----------
1. Withdraw Amount
2. Deposit Amount
3. Check Details
4. Log out and Exit

Enter your choice: 4
Thank You, Hope to see you again next time!

RESULT

Using only the C language, we have created a Banking Management Application which enables users to Withdraw money, Deposit money, and View their Account Details as per their requirement.