VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CSE-2006 Microprocessor and Interfacing

Slot : L35+L36

BANK MANAGEMENT SYSTEM

Submitted By :

1)Varun Goel-19BCE2296

2) Harshvardhan Mishra-19BCB0125

3) Zeba Mariam-19BCE2439

4)Hemang Mehta-19BCE2197

Guided By :

Prof. Arvind Kumar

School of Computer Science & Engineering

VIT University - India

FALL SEMESTER 2020-21

APPENDIX

ABSTRACT

Bank Management System is an application for keeping a person's record in a bank and for solving financial applications of a customer in a banking environment in order to nurture the needs of an end banking user by performing tasks

This is a Simple yet complex management system that we will make using the software EMU8086 and Assembly language programming. We will look at the following functions and understand the working of the same

- Create / Remove / Update account
- Deposit / Withdraw money
- Print account details
- Menu system
- Dynamic pin system
- Error detection

The main objective of bank management is to build an organic and optimal interaction system between the elements of banking mechanisms with a view to profit. The Bank Management System will also provide an improved design methodology, which envisages the future expansion, and modification, which is necessary for a core sector like banking.

INTRODUCTION

The bank management system is considered to be the most important thing in the economic world.

It allows the customers to complete basic banking transactions by sitting at their homes or offices through PC or laptop. The system provides the access to the customer to create an account, deposit, withdraw the cash from their account, you can also go through the accounts which are already present. The customers can access the bank website for viewing their account details and complete the transactions on account as per their requirements and wishes. It is designed to handle all the primary information required to calculate monthly statements of customer accounts which include monthly statements of any month. A separate database is maintained to safeguard all the details required in order to have correct statement calculation and statement generation. This project intends to introduce more user-friendliness in various activities such as record updating, maintenance, and searching. The searching of records has been made quite simple as all the details of the customer can be obtained by simply keying in the identification or account number of that customer. Similarly, record maintenance and updating can also be accomplished by using the account number with all the details being automatically generated. These details are also being promptly automatically updated in the master Me thus keeping the record absolutely up-to-date.

Today's banking is no longer confined to only branches, E-banking helps banking transactions by customers round the clock globally.

LITERATURE SURVEY

S. No	Paper Title	Name of the Conference/Journal, Year	Technology Used
1.	A Practice Approach to Teach Computer Organization and Interface Technology	2021 2nd Information Communication Technologies Conference (ICTC)	This paper proposes a new practice approach to train the students which can be divided into three aspects: 1) assembly language programming; 2) prototype computer; 3) embedded computer and its applications.
2.	Enriched Blended Learning through Virtual Experience in Microprocessors and Microcontrollers Course	Volume 34, Special Issue, ICTIEE 2021, January 2021	This paper highlights the effectiveness of using open source emulating environments like EMU8086 for providing a virtual laboratory experience. This virtual environment provides a complete visualization of the

			internal functionality of the microprocessor architecture.
3.	**Design of Embedded Based Dual Identification ATM Card Security System**	**2019 9th International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-19)**	**This paper aims to prevent the crime related to ATM card frauds and secure transactions. In this paper, two options are included like One Time Password (OTP) and Fingerprint detection for a successful transaction. The user can use any of the two options mentioned above for ATM transactions**
4.	**The Effect of Information Technology Management on Customer Satisfaction in E-Bank Services**	**THE 3rd INTERNATIONAL CONFERENCE ON ECONOMICS AND BUSINESS 2021**	**This research is a form of replication that will use the subject of e-banking user bank customers which aims to determine the factors that affect customer satisfaction of e-banking service users. This research uses quantitative methods with multiple linear regression analysis.**
5.	**Core Banking Technology and Its**	**Journal of Advanced Research**	**Acronym "Core" represents the**

	Top 6 Implementation Challenges	**in Operational and Marketing Management Volume 4, Issue 1 - 2021**	**Centralized Online Real-time Exchange. It is an operating system or software that processes financial transactions between different branches of a bank. It is a cost and time-efficient mode of banking that enables customers to perform basic banking operations without having to stop by the bank.**
6.	**The Role of Emerging Banking Technologies for Risk Management and Mitigation to Reduce Non-Performing Assets and Bank Frauds in the Indian Banking System**	**International Journal of e-Collaboration (IJeC), 2021**	**The paper focuses on how the newly emerging digital banking tools can be applied with improved skills of lending/risk managers to control and mitigate risk to resolve the issue of NPA and frauds through the support of Advance technology, Data Analytics, and MIS Reports.**

DRAWBACKS OF EXISTING WORKS

- **Technology and Service Interruptions**

  Anytime we use computers or internet service, we are at the mercy of the system's stability and efficiency. Our ability to access accounts online will naturally be affected if your internet service is running slowly or completely out for a period of time. Similarly, if the bank's servers go down or are temporarily unavailable due to scheduled site maintenance, you won't be able to gain online or mobile access to your banking information.

- **Security and Identity Theft Concerns**

  In general, online banking sites and mobile apps are designed to be secure and banks are continually putting updated security protocols in place. However, no system is completely foolproof and accounts can be hacked, resulting in identity theft via stolen login credentials.

- **Convenient but Not Always Faster**

  While it may take very little time to deposit a check via a bank's mobile app, you still need to wait for access to your money.

- **Lack of Personal Banker Relationship**

  For the most part you may be able to handle your general banking needs by yourself. Yet when problems arise if you don't have a personal relationship with a banker, it might be more difficult to get your issues resolved

- **A Limited Scope of Services**

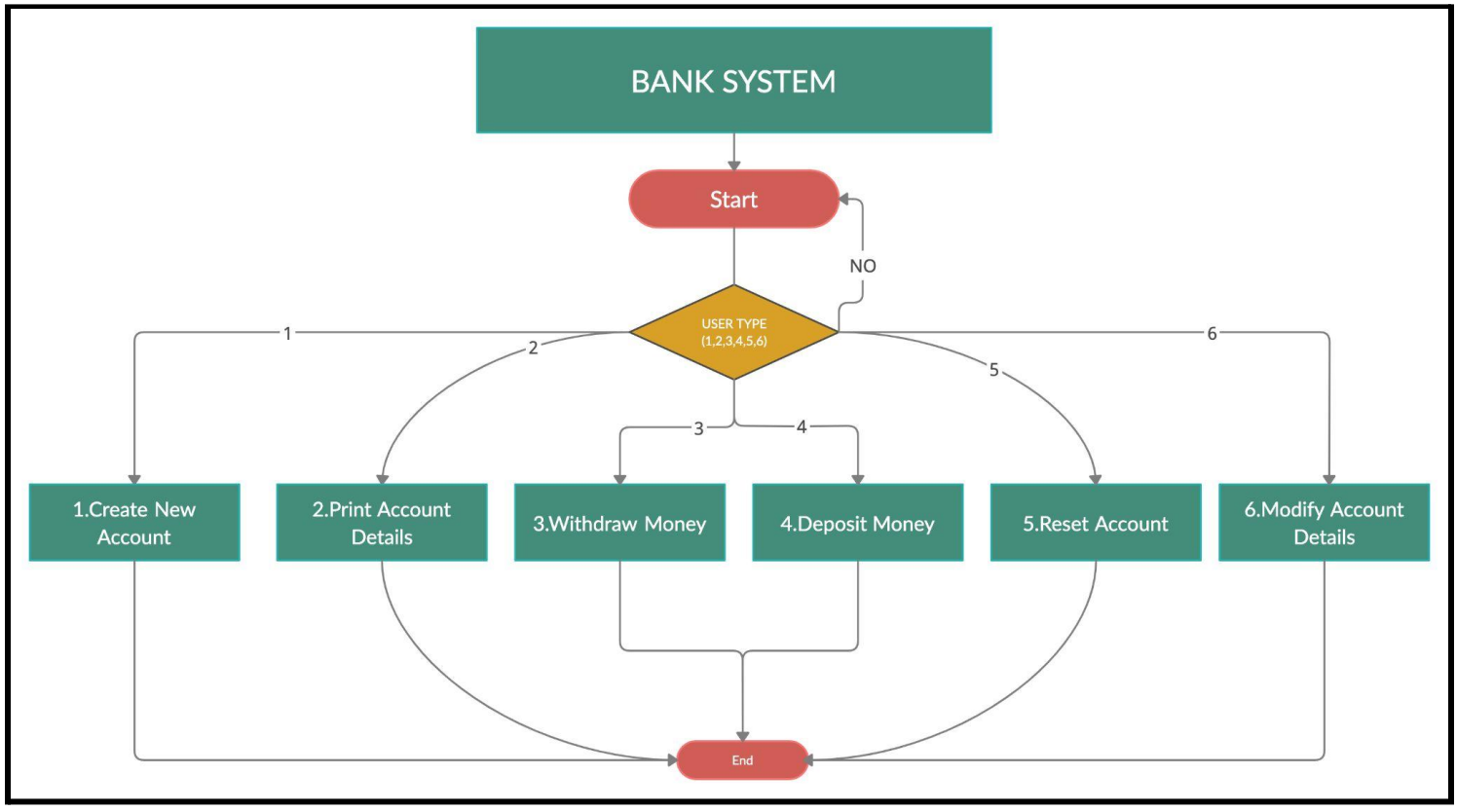
  Although you can do quite a bit with an online bank account, such as make deposits, check balances and pay bills, there are limitations to the kinds of services you can access.

PROPOSED WORK

We wish to provide an optimal solution for the current banking system's drawbacks by introducing EMU8086 as the interface. EMU8086 will help in overcoming the drawbacks of the system because of its following advantages:

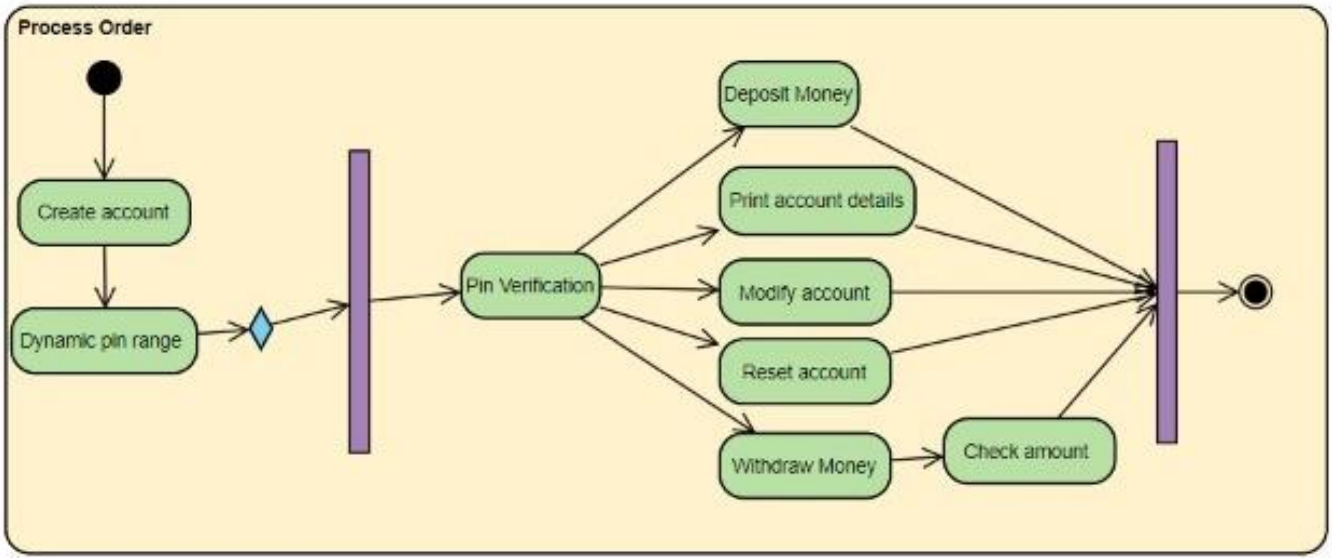
- The code can be reused and the instructions need not be written repeatedly.
- The storage space is less since it is a sub program.
- Developers can re-develop the code with ease since programming has become simple for them and the new developers can change the existing code.

FLOW CHART



Flow chart link:- https://app.creately.com/diagram/UGav8cmiWAz/edit

ACTIVITY DIAGRAM



IMPLEMENTATION

;1. Create Account
;2. Deposit Money
;3. Widthdraw Money
;4. Print Account Details
;5. Modify Account
;6. Reset Account
;7. Currerncy

;7. Dynamic Pin Range
;8. Pin Verification
;9. Checks if account is created before performing functions
;10. When withdrawing, checks whether if u have enough money in account

```
.model small
.stack 100h
.data
```

```
dmsg1 db '      _   _                    _          _ __  __        $'
dmsg2 db '     |  | |   | |\   | |  /    |      \  / |      |  |    |\  /|$'
dmsg3 db '     |_| |_| | \  | | /     |_     \/  |_  |  |__  | \/ |$'
dmsg4 db '     |  | |   | |  \ | | \        |  /       |  |  |    |    |$'
dmsg5 db '     |_| |   | |   \| |  \    _|  /    __|  |   |__ |    |$'
dmsg6 db '                                                          $'

op3mmsg1 db '            ___       __    ___                    $'
op3mmsg2 db ' |      | |    |  |    |     \  |    \     /\   |      |$'
op3mmsg3 db ' |  /\  | |    |  |_|      | |_/    /  \  |  /\  |$'
op3mmsg4 db ' | /  \ | |    |  |    |      | |\_      /__\ | /  \ |$'
op3mmsg5 db ' |/    \| |    |  |    |__/  |  \__ /      \|/    \|$'
op3mmsg6 db ' $'

op5mmsg1 db '  _      _  __ __   _ __ _____ $'
op5mmsg2 db ' |  \    /  || _ |  _ \ |  |  _/|  __  |$'
op5mmsg3 db ' | \ \  / / |||   || | | ||  | |_   \ \_/ /$'
op5mmsg4 db ' | |\ \/ /| |||   || | | ||  |  _/    \   /$'
op5mmsg5 db ' | |      | |||_|| |_| ||  | |       | |$'
op5mmsg6 db ' ||      |||___|_/ |_||       |_|$'


op2mmsg1 db  ' _   _ __          __  $'
op2mmsg2 db '  |   | |      |      /\      |   |$'
op2mmsg3 db '  |   | |_    |     /_\     |   |$'
op2mmsg4 db '  |   | |      |    /    \    |   |$'
op2mmsg5 db '  |_| |_   |   /      \ _|_ |__$'


op4mmsg1 db '  __   _  _  __  __  _ ___$'
op4mmsg2 db ' |  _ \ | _|| _ \| __ ||    ||  |_   _|$'
op4mmsg3 db ' | | | |||_  ||||||   ||| __||  |  | |$'
op4mmsg4 db ' | | | || | | |  / ||   ||(__ )|  |  | |$'
op4mmsg5 db ' | | | |||_ | |  ||__|||    ||  |  | |$'
op4mmsg6 db ' |_/ | _|||  |_||_||_|  ||$'


op1mmsg1 db '                  _       _                _$'
op1mmsg2 db '   __    __    | _|     / \             | __|$'
op1mmsg3 db '  |  _| |    |  ||_      /   \      __  ||$'
op1mmsg4 db '  | |     |    |  | |    / / \ \   |   _| | _|$'
op1mmsg5 db '  | |     | |\ \  ||_   / /   \ \    | |   ||_$'
op1mmsg6 db '  | |_  || \| |_| //     \\   ||   |__|$'
op1mmsg7 db '  |___|$'


op7mmsg1 db '  _         _   _   _          _$'
op7mmsg2 db ' |     |   | |   | |   | |    |\   | |     \  /$'
```

```
op7mmsg3 db ' |     |  | | _| | _| |__  | \ | |      \/$'
op7mmsg4 db ' |     |  | | \  | \  |    |  \ | |      /$'
op7mmsg5 db ' |_  |_| | \  | \  |_ |   \| |_  /$'



opmsg1 db '1. Create new Account$'
opmsg2 db '2. Print Account Details$'
opmsg3 db '3. Withdraw Money $'
opmsg4 db '4. Deposit Money $'
opmsg5 db '5. Reset Account $'
opmsg6 db '6. Modify Account Details$'
opmsg7 db '7. Currency$'

opmsg8 db 'Press Enter To Return to Main Menu $'

imsg db 'How can we help you today? : $'
inputCode db ?

;Account details

accountName db 100 dup('$')
accountPIN db 100 dup('$')
accountPINcount dw 0       ;This keeps track how many digit a pin is
totalAmount dw 0
inputAmountOption db ?


;Option 1 (Create Account) Messages
op1msg1 db '1. Enter Your Account Name: $'
op1msg2 db '2. Enter Your Account Pin: $'
op1msg3 db 'Congratulations!!! You have created New Account Successfully!!! $'

;Option 2 <Print details> Messages
op2msg1 db 'Account Name: $'
op2msg2 db 'Account PIN: $'
op2msg3 db 'No such existing account!$'
op2msg4 db 'Your Account Balance is: $'
op2msg5 db '0 $'

;Option 4 <Money> Messages
op4msg1 db '1. 1000$'
op4msg2 db '2. 2000$'
op4msg3 db '3. 5000$'
op4msg4 db '4. 10000$'
op4msg5 db 'Enter Code: $'
op4msg6 db 'Amount Exceeded!$'

;Option 5 <Reset> Messages
```

```
    op5msg1 db 'Account Has been reset successfully$'

    ;Option 6 <Modify Account> Messages
    op6msg0   db 'Account Details Successfully Changed !$'
    op6msg1_1 db '1. New Account Name ( old: $'
    op6msg1_2 db ' ) : $'
    op6msg2_1 db '2. New Account Pin ( old: $'
    op6msg2_2 db ' ) : $'

    ;Option 7<Currency Converter> Messages
    op7msg0 db 'Cuurencies available for transaction:$'
    op7msg1 db '1. US Dollar$'
    op7msg2 db '2. Australian Dollar$'
    op7msg3 db '3. UK Pound$'
    op7msg4 db '4. UAE Dirham$'
    op7msg5 db '5. Indian Rupee$'
    op7msg6 db 'Enter the currency you wish to proceed with: $'

    ;PIN Protection
    pinop_msg1 db 'Enter PIN: $'
    pinop_msg2 db 'Account NOT created ... $'

.code

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                                  ;
;                              U T I L S                           ;
;                                                                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Enter to Continue
proc etc
   etcin:
      mov ah,1
      int 21h
      cmp al,13
      je mainloop
      jmp etcin
   ret
etc endp

;This checks whether the account has been created or not using the pin Count
checkAccountCreated proc
  cmp accountPINcount,0
  je accountNotCreated
  ret

  accountNotCreated:
    call clearScreen
```

```
    printString pinop_msg2
    call etc

checkAccountCreated endp

;just mov number to ax and call this proc
printNumber PROC
    ;initilize count
    mov cx,0
    mov dx,0
    label1:
        ; if ax is zero
        cmp ax,0
        je print1

        ;initilize bx to 10
        mov bx,10

        ; extract the last digit
        div bx

        ;push it in the stack
        push dx

        ;increment the count
        inc cx

        ;set dx to 0
        xor dx,dx
        jmp label1

    print1:
        ;check if count
        ;is greater than zero
        cmp cx,0
        je exitprint

        ;pop the top of stack
        pop dx

        ;add 48 so that it
        ;represents the ASCII
        ;value of digits
        add dx,48

        ;interuppt to print a
        ;character
        mov ah,02h
        int 21h
```

```
        ;decrease the count
        dec cx
        jmp print1
exitprint:
ret
printNumber ENDP

clearScreen proc near
    call newLine
    call newLine
    ret
clearScreen endp

newLine proc near
    mov ah,2
    mov dl,10
    int 21h
    mov dl,13
    int 21h
    ret

newLine endp

macro printString str
  mov ah,9
  lea dx,str
  int 21h
endm

;Ask for user pin here
getPinInput proc
  call clearScreen

  printString pinop_msg1

  mov si,offset accountPIN
  mov cx,accountPINcount     ;Search n amount of times the pin Count
  getinput:

    mov ah,7
    int 21h

    cmp al,[si]

    mov dl,'*'
    mov ah,2
    int 21h
```

```
    jne mainloop

    inc si
  loop getinput

  ret
getPinInput endp


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                                    ;
;                      M E N U   SYSTEM                              ;
;                                                                    ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

DisplayMenu proc near
    printString dmsg1
    call newLine
    printString dmsg2
    call newLine
    printString dmsg3
    call newLine
    printString dmsg4
    call newLine
    printString dmsg5
    call newLine
    printString dmsg6
    call newLine

    call newLine
    printString opmsg1
    call newLine
    printString opmsg2
    call newLine
    printString opmsg3
    call newLine
    printString opmsg4
    call newLine
    printString opmsg5
    call newLine
    printString opmsg6
    call newLine
    printString opmsg7
    call newLine
    ret
DisplayMenu endp

GetInputMenuSystem proc near
    call newLine
```

```
    printString imsg
    mov ah,1
    int 21h
    mov inputCode,al
    ret
GetInputMenuSystem endp

.....................................................................
'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
;                                                                   ;
;             O P T I O N  1   => CREATE ACCOUNT                    ;
;                                                                   ;
.....................................................................
'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

macro ISop11 str
    mov si,offset str
    input:
        mov ah,1
        int 21h
        cmp al,13
        je labelop1_1
        mov [si],al
        inc si
        jmp input

    exitMac:
        ret

endm

macro ISop12 str
    mov si,offset str
    input2:
        mov ah,1
        int 21h
        cmp al,13
        je labelop1_2
        inc accountPINcount
        mov [si],al
        inc si
        jmp input2

    exitMac2:
        ret

endm

proc etcop1
   etcop1in:
      mov ah,1
```

```
      int 21h
      cmp al,13
      je mainloop
      jmp etcop1in
   ret
etcop1 endp

op1 proc

    call clearScreen

    printString op1mmsg1
    call newLine
    printString op1mmsg2
    call newLine
    printString op1mmsg3
    call newLine
    printString op1mmsg4
    call newLine
    printString op1mmsg5
    call newLine
    printString op1mmsg6
    call newLine
    printString op1mmsg7
    call newLine
    call newLine
    call newLine

    printString op1msg1
    ISop11 accountName

    labelop1_1:
      call newLine
      printString op1msg2
      ISop12 accountPIN

    labelop1_2:

        call newLine
        call newLine
        printString op1msg3
        call etcop1

    ret
op1 endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                              ;
;              O P T I O N  1   => PRINT DETAILS                    ;
```

```
;                                                    ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

proc etcop2
   call newLine
   printString opmsg8
   etcop2in:
      mov ah,1
      int 21h
      cmp al,13
      je mainloop
      jmp etcop2in
   ret
etcop2 endp

op2 proc

  call checkAccountCreated  ;check whether the account has been created or not
  call getPinInput  ;gets the pin input for verification
  call clearScreen

  printString op2mmsg1
  call newLine
  printString op2mmsg2
  call newLine
  printString op2mmsg3
  call newLine
  printString op2mmsg4
  call newLine
  printString op2mmsg5
  call newLine
  call newLine
  call newLine

  printString op2msg1
  printString accountName
  call newLine

  printString op2msg2
  printString accountPIN
  call newLine

  printString op2msg4
  mov ax,totalAmount
  cmp ax,0
  je noMoneyError
  call printNumber
  call newLine
```

```
  call etcop2

  noMoneyError:
    printString op2msg5
    call newLine
    call etcop2

  ret

op2 endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                                  ;
;              O P T I O N  3   => WIDTHDRAW MONEY                 ;
;                                                                  ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

op3 proc

  call checkAccountCreated  ;check whether the account has been created or not
  call getPinInput  ;gets the pin input
  call clearScreen


  printString op3mmsg1
  call newLine
  printString op3mmsg2
  call newLine
  printString op3mmsg3
  call newLine
  printString op3mmsg4
  call newLine
  printString op3mmsg5
  call newLine
  printString op3mmsg6
  call newLine
  call newLine
  call newLine

  printString op4msg1
  call newLine
  printString op4msg2
  call newLine
  printString op4msg3
  call newLine
  printString op4msg4
  call newLine

  call inputAmountCode
```

```
cmp inputAmountOption,'1'
je wcop1

cmp inputAmountOption,'2'
je wcop2

cmp inputAmountOption,'3'
je wcop3

cmp inputAmountOption,'4'
je wcop4

;check if widraw amount <= totalAmount in acc
wcop1:
  mov bx,totalAmount
  cmp bx,1000
  jl nowaybro
  sub totalAmount,1000
  jmp mainloop
wcop2:
  mov bx,totalAmount
  cmp bx,2000
  jl nowaybro
  sub totalAmount,2000
  jmp mainloop
wcop3:
  mov bx,totalAmount
  cmp bx,5000
  jl nowaybro
  sub totalAmount,5000
  jmp mainloop
wcop4:
  mov bx,totalAmount
  cmp bx,10000
  jl nowaybro
  sub totalAmount,10000
  jmp mainloop

;error message for widthdrawing too much
nowaybro:

  call newLine
  call newLine
  printString op4msg6
  call etcop4

ret
```

```
op3 endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                    ;
;           O P T I O N  4   => DEPOSIT MONEY                 ;
;                                                    ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

proc etcop4
   call newLine
   ;printString opmsg8
   etcop4in:
      mov ah,1
      int 21h
      cmp al,13
      je mainloop
      jmp etcop4in
   ret
etcop4 endp

proc inputAmountCode
  call newLine
  printString op4msg5
  mov ah,1
  int 21h
  mov inputAmountOption,al
  ret
inputAmountCode endp

op4 proc

  call checkAccountCreated  ;check whether the account has been created or not
  call getPinInput  ;gets the pin input
  call clearScreen

  printString op4mmsg1
  call newLine
  printString op4mmsg2
  call newLine
  printString op4mmsg3
  call newLine
  printString op4mmsg4
  call newLine
  printString op4mmsg5
  call newLine
  printString op4mmsg6
  call newLine
  call newLine
  call newLine
```

```
  printString op4msg1
  call newLine
  printString op4msg2
  call newLine
  printString op4msg3
  call newLine
  printString op4msg4
  call newLine

  call inputAmountCode

  cmp inputAmountOption,'1'
  je dcop1

  cmp inputAmountOption,'2'
  je dcop2

  cmp inputAmountOption,'3'
  je dcop3

  cmp inputAmountOption,'4'
  je dcop4

  dcop1:
   add totalAmount,1000
   jmp mainloop
  dcop2:
   add totalAmount,2000
   jmp mainloop
  dcop3:
   add totalAmount,5000
   jmp mainloop
  dcop4:
   add totalAmount,10000
   jmp mainloop

  ret

op4 endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                          ;
;              O P T I O N  5   => RESET ACCOUNT                   ;
;                                                          ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

proc etcop5
   call newLine
```

```
   ;printString opmsg8
   etcop5in:
      mov ah,1
      int 21h
      cmp al,13
      je mainloop
      jmp etcop5in
   ret
etcop5 endp

op5 proc

  call checkAccountCreated  ;check whether the account has been created or not
  call getPinInput  ;gets the pin input

  ;Do the rest of the work .. display the data
  call clearScreen

  mov si,offset accountName
  mov cx,30
  l1:
    mov [si],' '
    inc si
  loop l1

  mov cx,30
  mov si,offset accountPIN
  l2:
    mov [si],' '
    inc si
  loop l2

  mov totalAmount,0
  mov accountPINcount,0 ;reset pin count

  printString op5msg1
  call etcop5
  ret
op5 endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                                   ;
;               O P T I O N  6   => MODIFY ACCOUNT DETAILS          ;
;                                                                   ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

proc etcop6
   call newLine
   ;printString opmsg8
```

```
   etcop6in:
      mov ah,1
      int 21h
      cmp al,13
      je mainloop
      jmp etcop6in
   ret
etcop6 endp

macro ISop6 str
 mov si,offset str
    ISop6input:
        mov ah,1
        int 21h
        cmp al,13
        je labelop6_1
        mov [si],al
        inc si
        jmp ISop6input
endm

macro ISop6_2 str
 mov si,offset str
 mov accountPINcount,0 ;reset pin count
    ISop6_2input:
        mov ah,1
        int 21h
        cmp al,13
        je labelop6_2
        inc accountPINcount ;increment pin account again
        mov [si],al
        inc si
        jmp ISop6_2input
endm

op6 proc

  call checkAccountCreated  ;check whether the account has been created or not
  call getPinInput  ;gets the pin
  call clearScreen

  printString op5mmsg1
  call newLine
  printString op5mmsg2
  call newLine
  printString op5mmsg3
  call newLine
  printString op5mmsg4
  call newLine
```

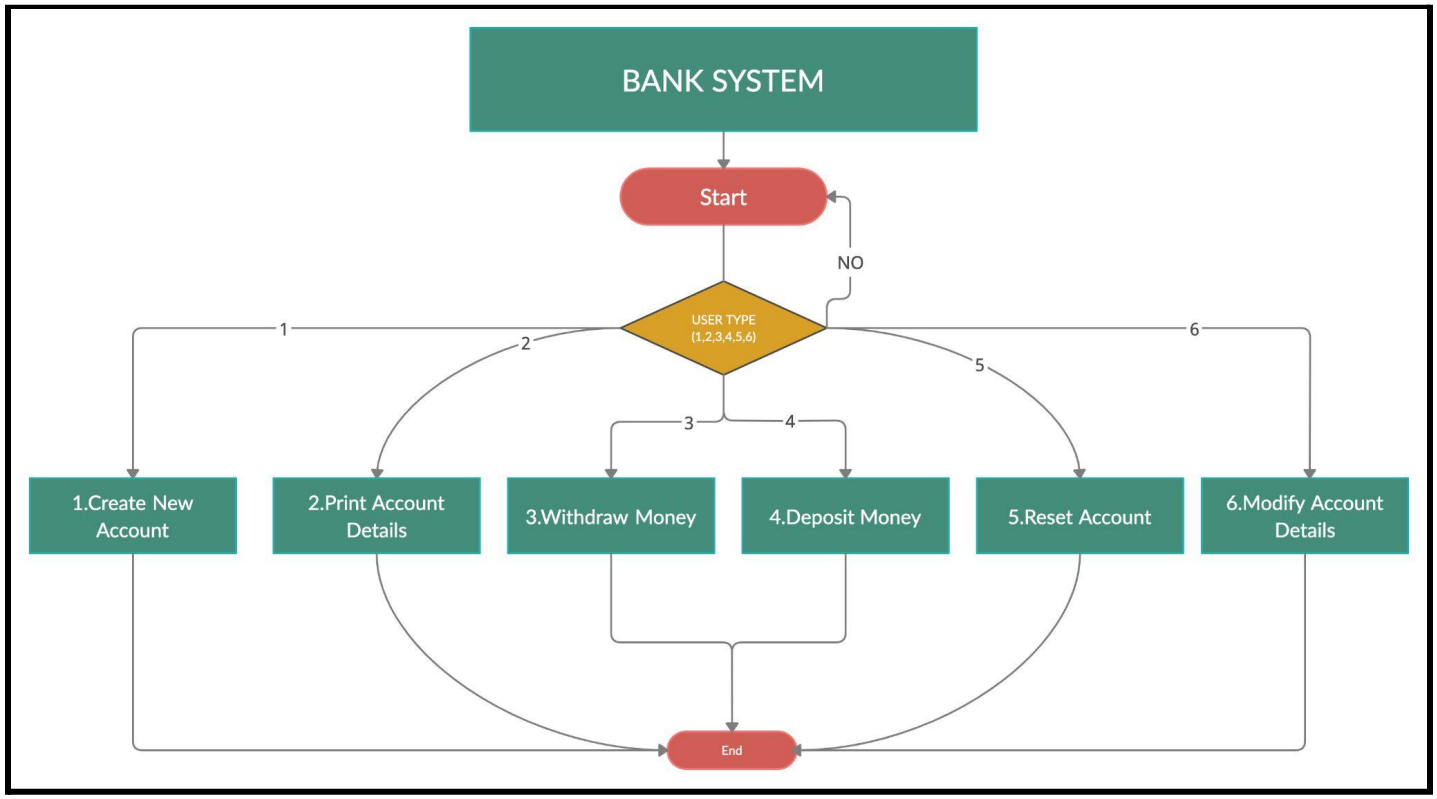
```
  printString op5mmsg5
  call newLine
  printString op5mmsg6
  call newLine
  call newLine
  call newLine

  ;;account name
  printString op6msg1_1
  printString accountName
  printString op6msg1_2

  ISop6 accountName ;input accountName

  labelop6_1:

    call newLine
    printString op6msg2_1
    printString accountPIN
    printString op6msg2_2
    ISop6_2 accountPIN

  labelop6_2:

    ;Finished MSG
    call newLine
    call newLine
    printString op6msg0
    call etcop6


  ret
op6 endp

.....................................................................
'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
;                                                                   ;
;                         CURRENCY                                  ;
;                                                                   ;
.....................................................................
'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

proc etcop7
   call newLine
   ;printString opmsg8
   etcop7in:
      mov ah,1
      int 21h
      cmp al,13
      je mainloop
      jmp etcop7in
```

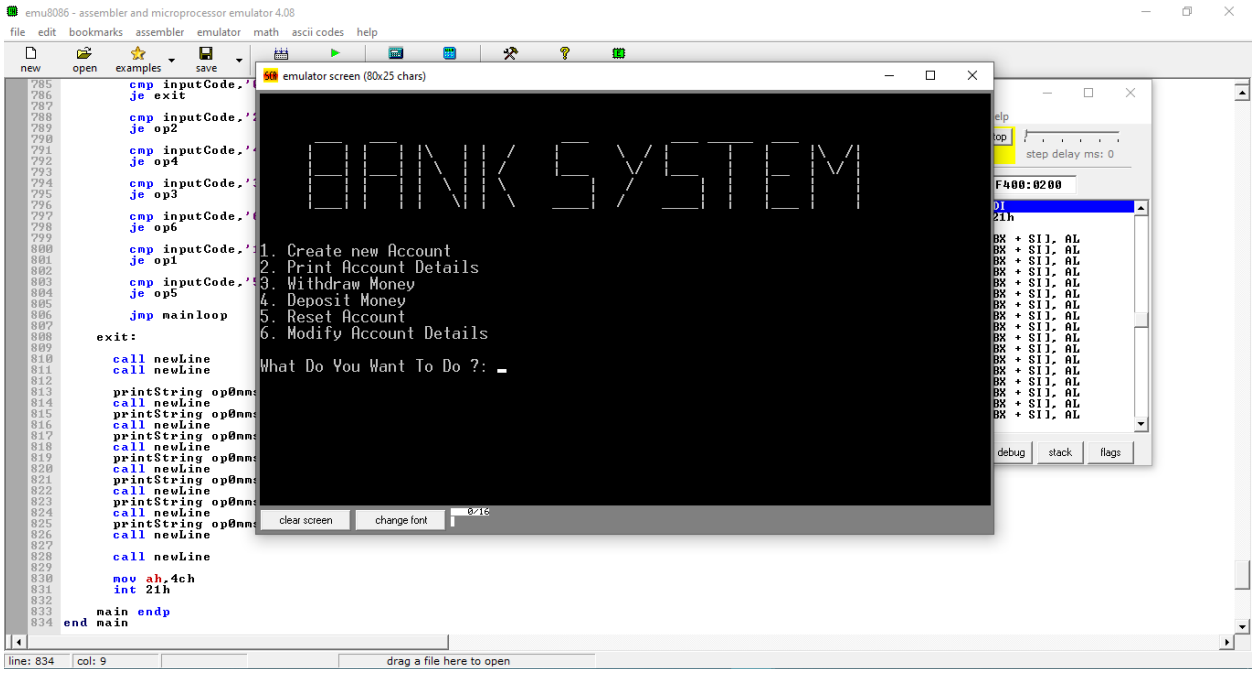
```
   ret
etcop7 endp

proc inputCurrency
  call newLine
  mov ah,1
  int 21h
  mov inputAmountOption,al
  ret
inputCurrency endp

op7 proc

  call checkAccountCreated  ;check whether the account has been created or not
  call getPinInput  ;gets the pin input
  call clearScreen

  printString op7mmsg1
  call newLine
  printString op7mmsg2
  call newLine
  printString op7mmsg3
  call newLine
  printString op7mmsg4
  call newLine
  printString op7mmsg5
  call newLine
  call newLine
  call newLine

  printString op7msg0
  call newLine
  printString op7msg1
  call newLine
  printString op7msg2
  call newLine
  printString op7msg3
  call newLine
  printString op7msg4
  call newLine
  printString op7msg5
  call newLine
  printString op7msg6
  call newLine
  call newLine

  call inputCurrency

  cmp inputAmountOption,'1'
```

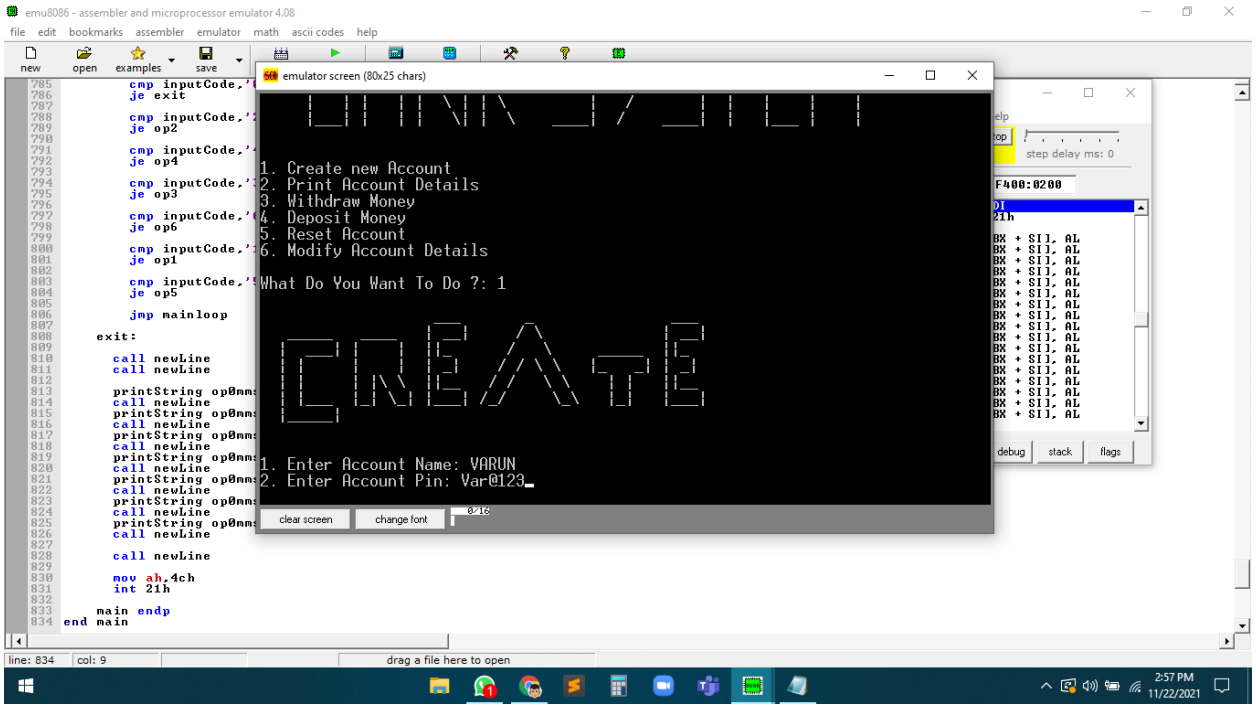
```
  je ccop1

  cmp inputAmountOption,'2'
  je ccop2

  cmp inputAmountOption,'3'
  je ccop3

  cmp inputAmountOption,'4'
  je ccop4

  cmp inputAmountOption,'5'
  je ccop5

  ccop1:
   imul totalAmount,75
   jmp mainloop
  ccop2:
   imul totalAmount,54
   jmp mainloop
  ccop3:
   imul totalAmount,100
   jmp mainloop
  ccop4:
   imul totalAmount,20
   jmp mainloop

  ccop5:
   imul totalAmount,1
   jmp mainloop

  ret

op7 endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                                     ;
;                 E N T R Y   P O I N T               ;
;                                                     ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

Main proc

   mov ax,@data
   mov ds,ax

   mainloop:

       call clearScreen
```

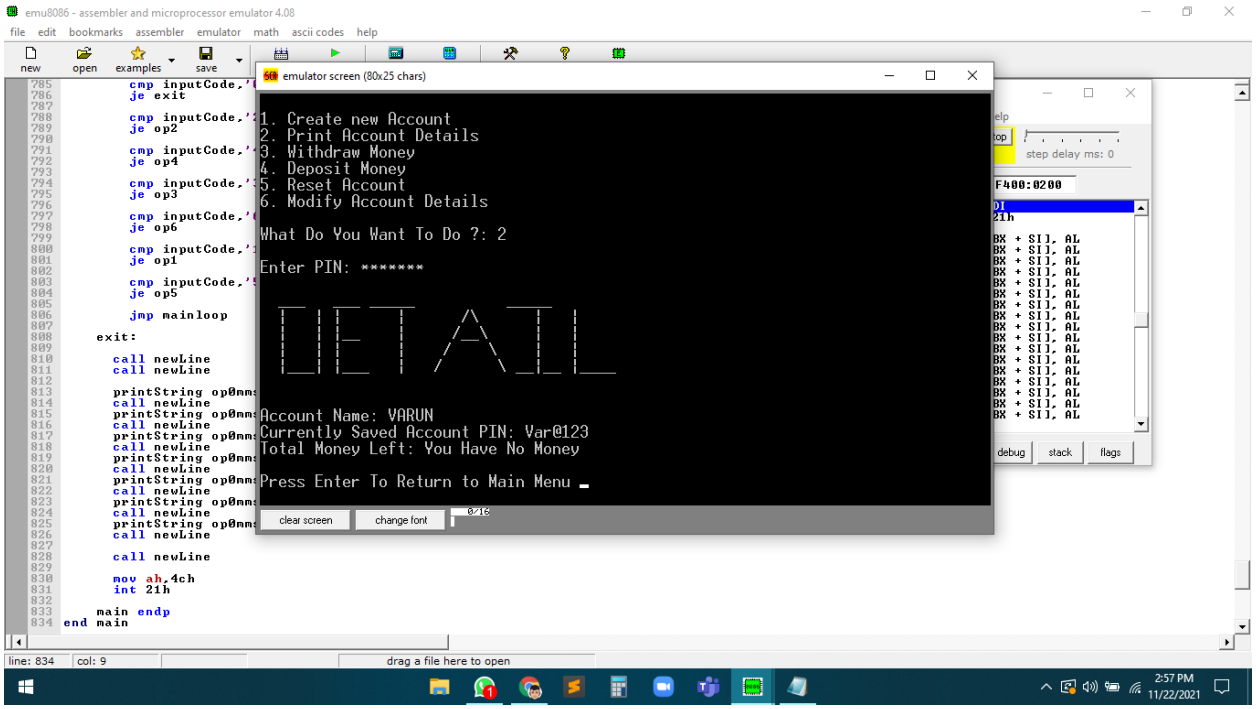
```
        call DisplayMenu
        call GetInputMenuSystem

        cmp inputCode,'0'
        je exit

        cmp inputCode,'2'
        je op2

        cmp inputCode,'4'
        je op4

        cmp inputCode,'3'
        je op3

        cmp inputCode,'6'
        je op6

        cmp inputCode,'1'
        je op1

        cmp inputCode,'5'
        je op5

        cmp inputCode,'7'
        je op7

        jmp mainloop

    exit:

      call newLine
      call newLine

      call newLine

      mov ah,4ch
      int 21h

    main endp
end main
```

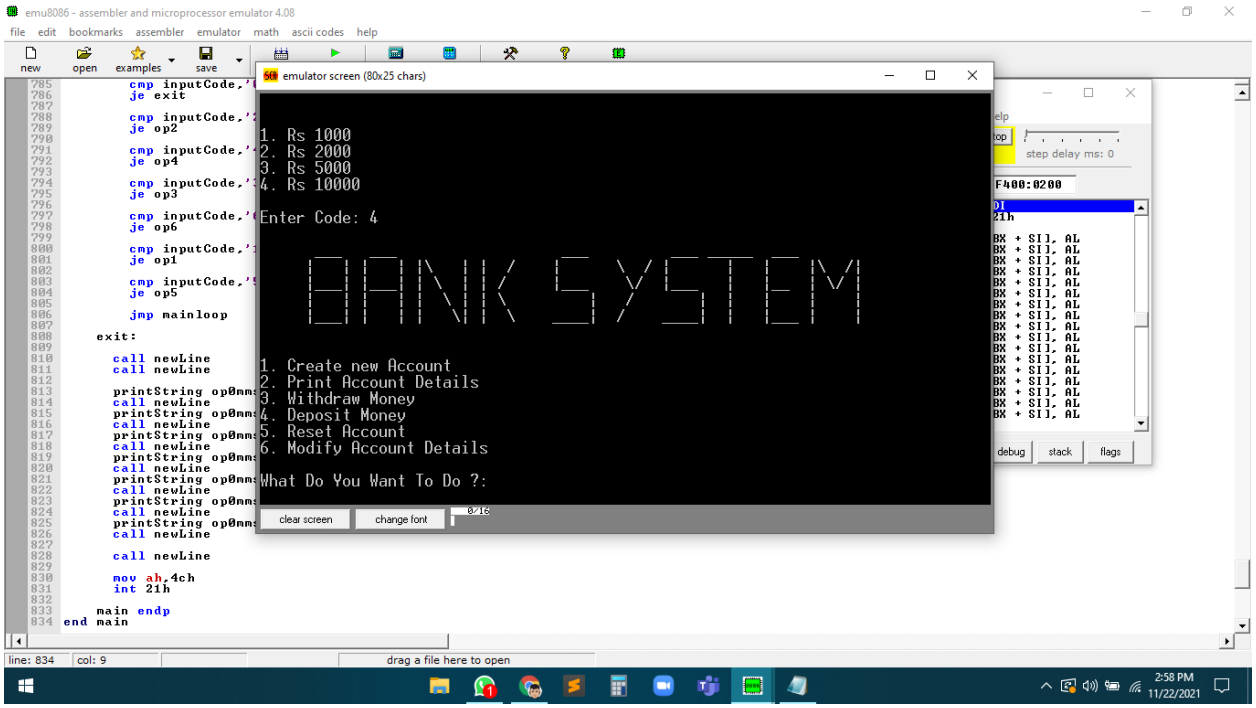
RESULTS

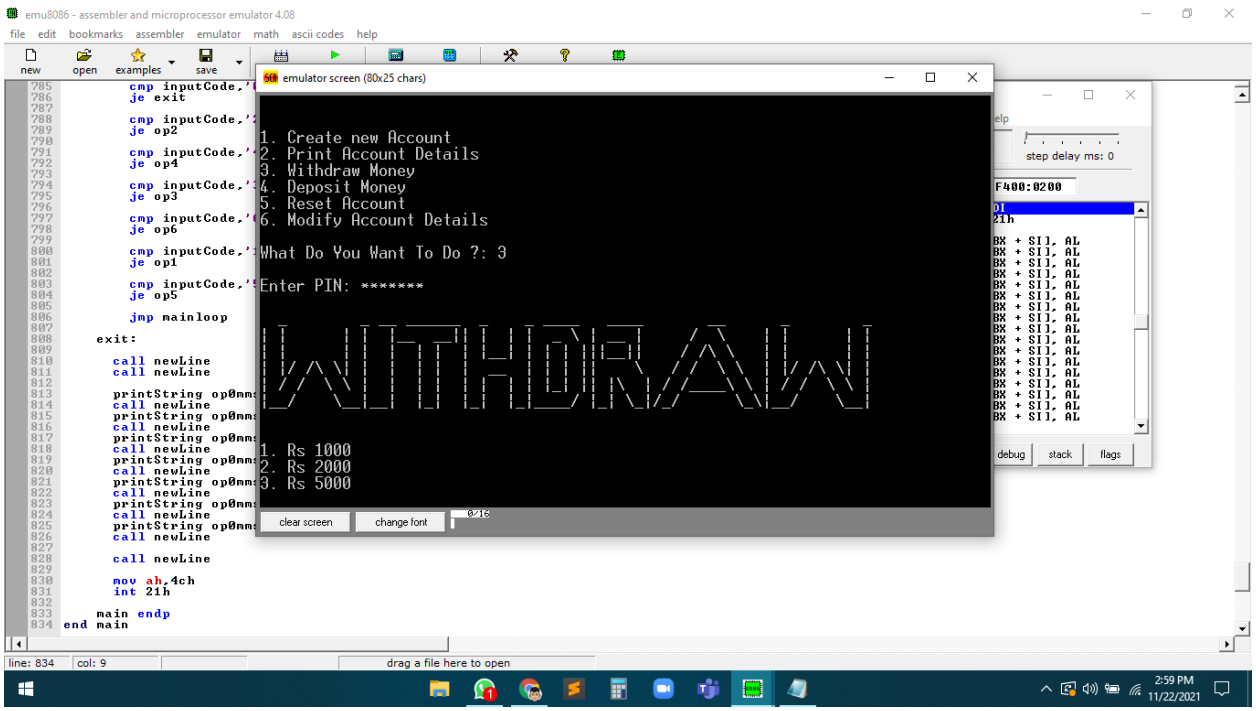

BANK SYSTEM
Start
NO
USER TYPE
(1,2,3,4,5,6)
1
2
3
4
5
6
1.Create New Account
2.Print Account Details
3.Withdraw Money
4.Deposit Money
5.Reset Account
6.Modify Account Details
End


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save
emulator screen (80x25 chars)
1. Create new Account
2. Print Account Details
3. Withdraw Money
4. Deposit Money
5. Reset Account
6. Modify Account Details
What Do You Want To Do ?:
clear screen
change font
step delay ms: 0
F400:0200
debug stack flags
line: 834 col: 9
drag a file here to open

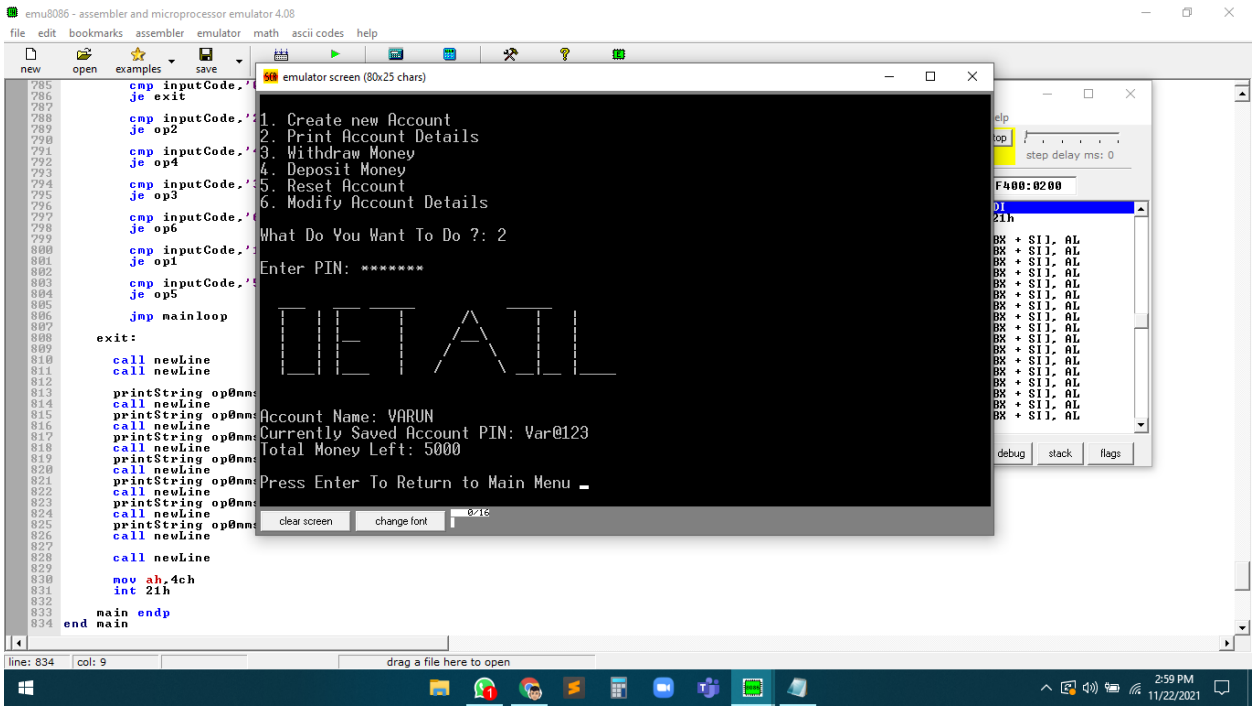

emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save
emulator screen (80x25 chars)
1. Create new Account
2. Print Account Details
3. Withdraw Money
4. Deposit Money
5. Reset Account
6. Modify Account Details
What Do You Want To Do ?: 1
1. Enter Account Name: VARUN
2. Enter Account Pin: Var@123
clear screen
change font
line: 834 col: 9
drag a file here to open
2:57 PM
11/22/2021

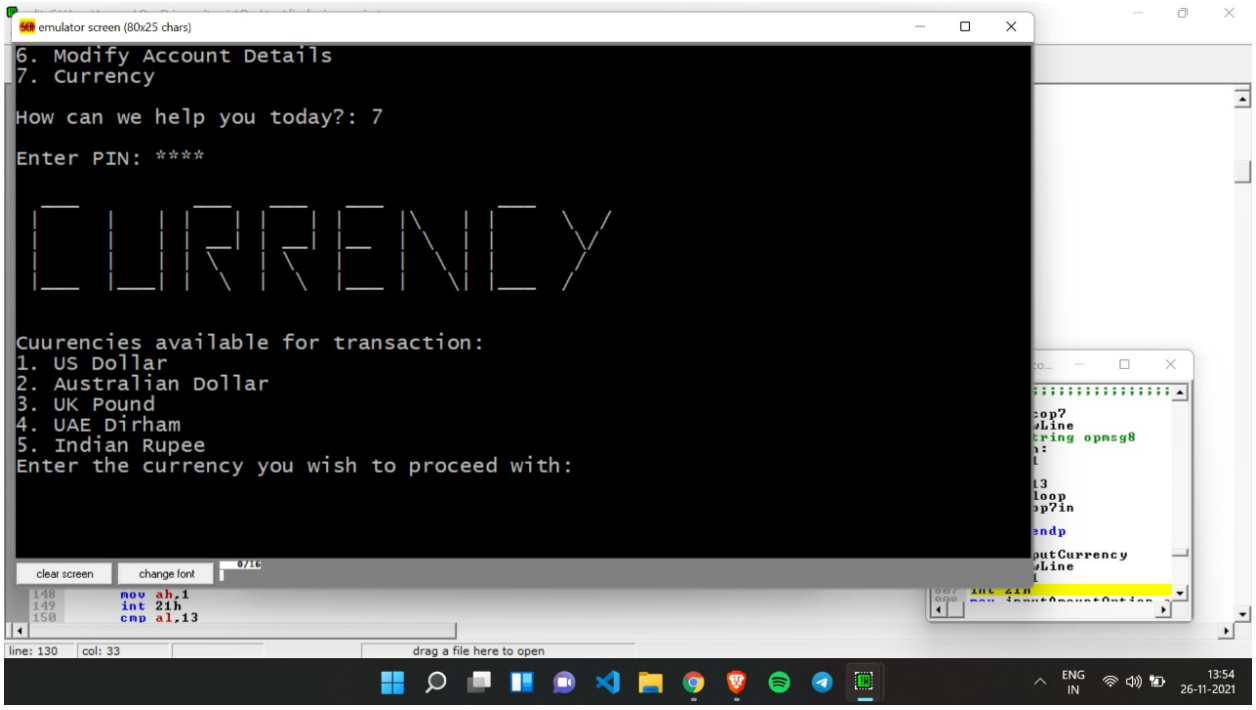

emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save
emulator screen (80x25 chars)
1. Create new Account
2. Print Account Details
3. Withdraw Money
4. Deposit Money
5. Reset Account
6. Modify Account Details
What Do You Want To Do ?: 2
Enter PIN: *******
Account Name: VARUN
Currently Saved Account PIN: Var@123
Total Money Left: You Have No Money
Press Enter To Return to Main Menu
clear screen
change font
line: 834 col: 9
drag a file here to open
2:57 PM
11/22/2021


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save
emulator screen (80x25 chars)
1. Rs 1000
2. Rs 2000
3. Rs 5000
4. Rs 10000
Enter Code: 4
BANK SYSTEM
1. Create new Account
2. Print Account Details
3. Withdraw Money
4. Deposit Money
5. Reset Account
6. Modify Account Details
What Do You Want To Do ?:
clear screen
change font
step delay ms: 0
F400:0200
debug
stack
flags
line: 834
col: 9
drag a file here to open
2:58 PM
11/22/2021


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save
emulator screen (80x25 chars)
1. Create new Account
2. Print Account Details
3. Withdraw Money
4. Deposit Money
5. Reset Account
6. Modify Account Details
What Do You Want To Do ?: 3
Enter PIN: *******
WITHDRAW
1. Rs 1000
2. Rs 2000
3. Rs 5000
clear screen
change font
step delay ms: 0
F400:0200
debug
stack
flags
line: 834
col: 9
drag a file here to open
2:59 PM
11/22/2021

After Withdrawing Rs 5000 My Account Details:-





Thus, We have successfully completed the implementation of the Bank system using emu8086.

CONCLUSION

With regard to our busy lives, online banking comes in as a boon to us . And our research has made us realize that there are quite a few benefits of doing online banking:

1. You don't need to wait in line.
2. You don't have to worry about the bank's hours and plan your day according to the bank closing time.
3. You can check your balance anywhere and wherever, not just when you get a statement.
4. It is fast, secure, convenient, and free.
5. It is quite simple and quick
6. The online banking system has Improved data security which restricts unauthorized access.

This assignment is evolved to nurture the wishes of a consumer in a banking region with the aid of embedding all of the duties of transactions taking place in a financial institution. Future model of this assignment will nonetheless be an awful lot more suitable than the present day model. Writing and depositing exams are possibly the most essential approaches to transport cash inside and out of a bank account, however improvements in the era have delivered ATM and debit card transactions. All banks have policies about how lengthy it takes to get right of entry to your deposits, what number of debit card transactions you are allowed in a day, and what sort of coins you may withdraw from an ATM. Access to the stability for your bank account also can be restricted with the aid of using companies that vicinity holds for your funds. Banks are supplying net banking offerings additionally in order that the clients may be attracted. By asking the financial institution employees we got here to realize that most of the net financial institution account holders are teenagers and enterprise men. Online banking is a revolutionary device that is rapidly turning into a necessity. It is a successful strategic weapon for banks to stay worthwhile in a risky and aggressive market of today. If right schooling need to receive to consumer with the aid of using the financial institution employs to open an account can be useful secondly the internet site need to be made friendlier from in which the primary time clients can

without delay make and get right of entry to their accounts. Thus the Bank Management System it's miles evolved and finished successfully.

In result, we have thoroughly analyzed and studied the Banking management system and succeeded in implementing the banking features via emu8086.

REFERENCES

- Zuo, D., Cheng, W., & Luo, J. (2021, May). A Practice Approach to Teach Computer Organization and Interface Technology. In *2021 2nd Information Communication Technologies Conference (ICTC)* (pp. 338-342). IEEE.
- Deepa, M., Reba, P., Santhanamari, G., & Susithra, N. (2021). Enriched Blended Learning through Virtual Experience in Microprocessors and Microcontrollers Course. *Journal of Engineering Education Transformations*, *34*, 642-650.
- Kale, P. H., & Jajulwar, K. K. (2019, November). Design of Embedded Based Dual Identification ATM Card Security System. In *2019 9th International Conference on Emerging Trends in Engineering and Technology-Signal and Information Processing (ICETET-SIP-19)* (pp. 1-5). IEEE.
- Risqi, P. M., & Handayani, Y. I. (2021). The Effect of Information Technology Management on Customer Satisfaction in E-Bank Services. *E-PROCEEDING STIE MANDALA*, 227-234.
- Haralayya, B. (2021). Core Banking Technology and Its Top 6 Implementation Challenges. *Journal of Advanced Research in Operational and Marketing Management*, *4*(1), 25-27.
- Bhasin, N. K., & Rajesh, A. (2022). The Role of Emerging Banking Technologies for Risk Management and Mitigation to Reduce Non-Performing Assets and Bank Frauds in the Indian Banking System. *International Journal of e-Collaboration (IJeC)*, *18*(1), 1-25.
- https://scholar.google.com/
- https://www.philadelphia.edu.jo/academics/qhamarsheh/uploads/emu8086.pdf
- http://www.gabrielececchetti.it/Teaching/CalcolatoriElettronici/Docs/i8086_instruction_set.pdf
- http://www1.frm.utn.edu.ar/arquitectura/t86.pdf
- https://www.youtube.com/