KARNATAK LAW SOCIETY’S

GOGTE INSTITUTE OF TECHNOLOGY

UDYAMBAG, BELAGAVI-590008

(An Autonomous Institution under Visvesvaraya Technological University, Belagavi)

**(APPROVED BY AICTE, NEW DELHI)**

Department of Electronics and Communication Engineering



*Course Activity Report on*

**Bank Application**

**in**

***C/C++***

***Submitted by***

*Shubham Parasharam Marihalkar(2GI17EC134)*

*Shreyas Maruti Lalge(2GI17EC129)*

*Vrushabh Ashok Lengade(2GI17EC165)*

*Swapnil Pundalik Gudagi(2GI17EC149)*

**Guide**

Prof. Uttam Deshpande

(Assistant Professor)

**2019-20**

**Bank Application**

**Objective:**

To develop bank application and validate the account holder based on current time OTP generation.

**CODE:**

#include<stdio.h>

#include<string.h>

#include<time.h>

#define N 5//number of customers

//Function declarations

void deposit();

void withdraw();

void balance();

int otpgenerate();

//Structure to maintain accounts

struct account{

long int accno;//account number

char name[20];//name of customer

char type;//saving or current

float bal;//balance

int pin;

}custmer[N]={{101,"Shubham",'S',10000,111},

{102,"Shreyas",'S',12000,222},

{103,"Vrushabh",'S',14000,333},

{104,"Swapnil",'C',90000,444},

{105,"SwapnilC",'C',2000,555}

};

int accin;//account index in the structure

float amount;//withdraw or deposit amount

//function to generate OTP based o current time

int otpgenerate()

{

FILE \*fp;//file pointer to hold file address

void fstr(char str[100])//function to put a string into the file

{

fp=fopen("otp.txt","a");

fprintf(fp,str);

fclose(fp);

}

void fint(int a)//function to put a integer into the file

{

fp=fopen("otp.txt","a+");

fprintf(fp,"%d",a);

fclose(fp);

}

fp=fopen("otp.txt","w+");

fprintf(fp,"OTP GENERATION BASED ON CURRENT TIME\n");

fclose(fp);

int i,j,otp[7],otp1,otp2,rem;

time\_t t;//Variable of type time

time(&t);

char tim[26];//String Variable to store current time

int binary[25][7];//array to hold binary value of each character of the current time

int asci[25];//array to hold ASCI value of the each character of the current time

strncpy(tim,ctime(&t),25);//copy current time into a string

fstr(tim);//put current time into file

//Converting each character of current time to ASCI value

for(i=0;tim[i]!='\0';i++)

{

asci[i]=(int)(tim[i]);

}

//binary sequence initialization to avoid padding of zeros

for(i=0;i<25;i++)

{

for(j=0;j<6;j++)

{

binary[i][j]=0;

}

}

//binary values calculation of the ASCI values of each current time character

for(i=0;i<25;i++)

{

for(j=0;j<7;j++)

{

binary[i][j]=asci[i]%2;

asci[i]=asci[i]/2;

}

}

//add each column value in a single dimensional array index

for(i=0;i<7;i++)

{

for(j=0;j<25;j++)

{

otp[i]+=binary[j][i];

}

}

//add each member of otp in otp1

otp1=0;

for(i=0;i<7;i++)

{

otp1+=otp[i];

}

//fixing length of otp to 6

int otplen=6;

otp2=0;//to store the otp of length=otplen

for(i=0;i<otplen;i++)

{

rem=otp1%10;

otp1=otp1/10;

otp2=otp2\*10+rem;

}

fstr("\nOTP = ");

fint(otp2);//write OTP to fi

return(otp2);

}

//Function to deposit amount in the account

void deposit()

{

printf("Enter the amount = ");

scanf("%f",&amount);

printf("\n");

custmer[accin].bal+=amount;//add amount to the account

balance();

}

//Function to withdraw amount amount from account

void withdraw()

{

printf("Enter the amount = ");

scanf("%f",&amount);

printf("\n");

int otp=otpgenerate();//OTP generation to validate the withdraw transaction

int count=0,pw;

//Check for balance availability

if(amount<=custmer[accin].bal)

{

pass1:

printf("\nenter OTP ");

scanf("%d",&pw);

printf("\n");

//check entered OTP matches with generated OTP

if(!(otp==pw))

{

count++;

printf("\nWrong OTP\nattempted %d times\nmaximum attempts left %d\n",count,3-count);

if(count<3)

goto pass1;

else

{

printf("maximum attempts reached\nYOUR ACCOUNT IS LOCKED\nCONTACT BANK FOR MORE DETAIS\n ");

exit(0);

}

}

else

{

custmer[accin].bal-=amount;//Withdraw Amount

balance();

}

}

else

{

printf("Insufficient Balance in your account\n");

balance();

}

}

//Function to display balance

void balance()

{

printf("Balance = %f\n",custmer[accin].bal);

}

int main()

{

time\_t t;

time(&t);

while(1){

int accnum,i,ch,f=0,pw,count=0;

start1:

printf("\*\*\*\*\*\*\*\*\*\*SPM BANK \*\*\*\*\*\*\*\*\*\*\*\t\t\t%s",ctime(&t));

printf("Enter the Account number= ");

scanf("%d",&accnum);//input account number

printf("\n");

for(i=0;i<N;i++)//check for existance of the account

{

if(accnum==custmer[i].accno)

{

accin=i;

f=1;

break;

}

}

if(f==0)

{

printf("\nInvalid Account number \n");

exit(0);

}

int otp=otpgenerate();//OTP generation for login

pass:

printf("\nenter OTP ");//Input OTP from user

scanf("%d",&pw);

printf("\n");

if(!(otp==pw))//Password validation and if entered wrong then give max of 3 tries then lock account

{

count++;

printf("\nWrong password\nattempted %d times\nmaximum attempts left %d\n",count,3-count);

if(count<3)

goto pass;

else

{

printf("maximum attempts reached\nYOUR ACCOUNT IS LOCKED\nCONTACT BANK FOR MORE DETAIS\n ");

exit(0);

}

}

//Display account details

printf("Account number= %d\nName : %s\nAccount type= %c\nBalance = %f\n",custmer[accin].accno,custmer[accin].name,custmer[accin].type,custmer[accin].bal);

while(1)

{

//Bank menu

time\_t t;

time(&t);

printf("\*\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\t\t\t%s",ctime(&t));

printf("\nEnter your choice\n1 : Balance Enquiry\n2 : Deposit Amount \n3 : Withdraw Amount\n4 :Sign Out\n5 :Exit\n");

scanf("%d",&ch);//input choice of operation

switch(ch)

{

case 1:balance();break;//to display balance

case 2:deposit();break;//to deposit amount

case 3:withdraw();break;//to withdraw amount

case 4: system("cls");//sign out

goto start1;

case 5:exit(0);//close application

default:printf("Enter the correct choice");break;

}

}

}

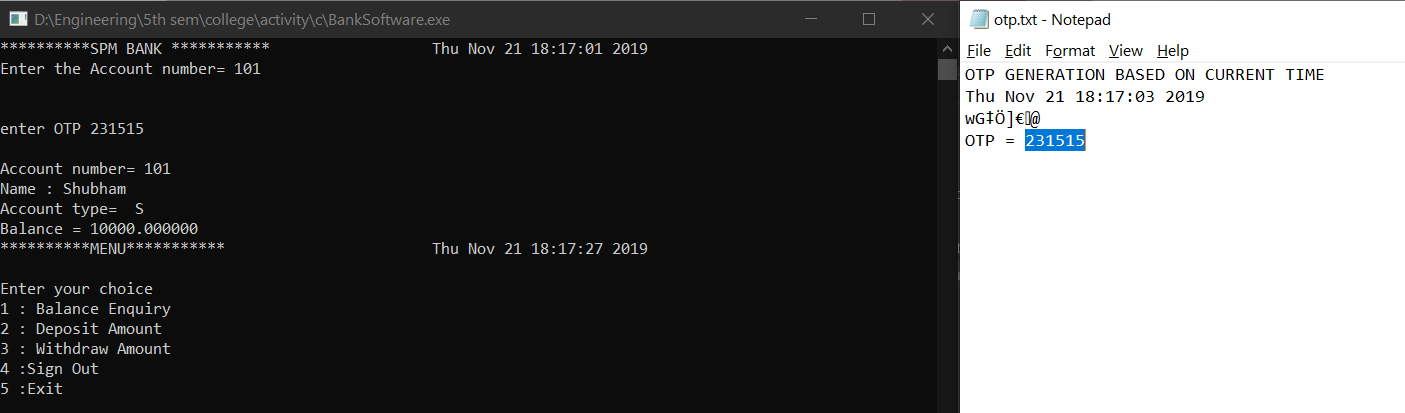
return 0;

}

**INPUT/OUTPUT:**

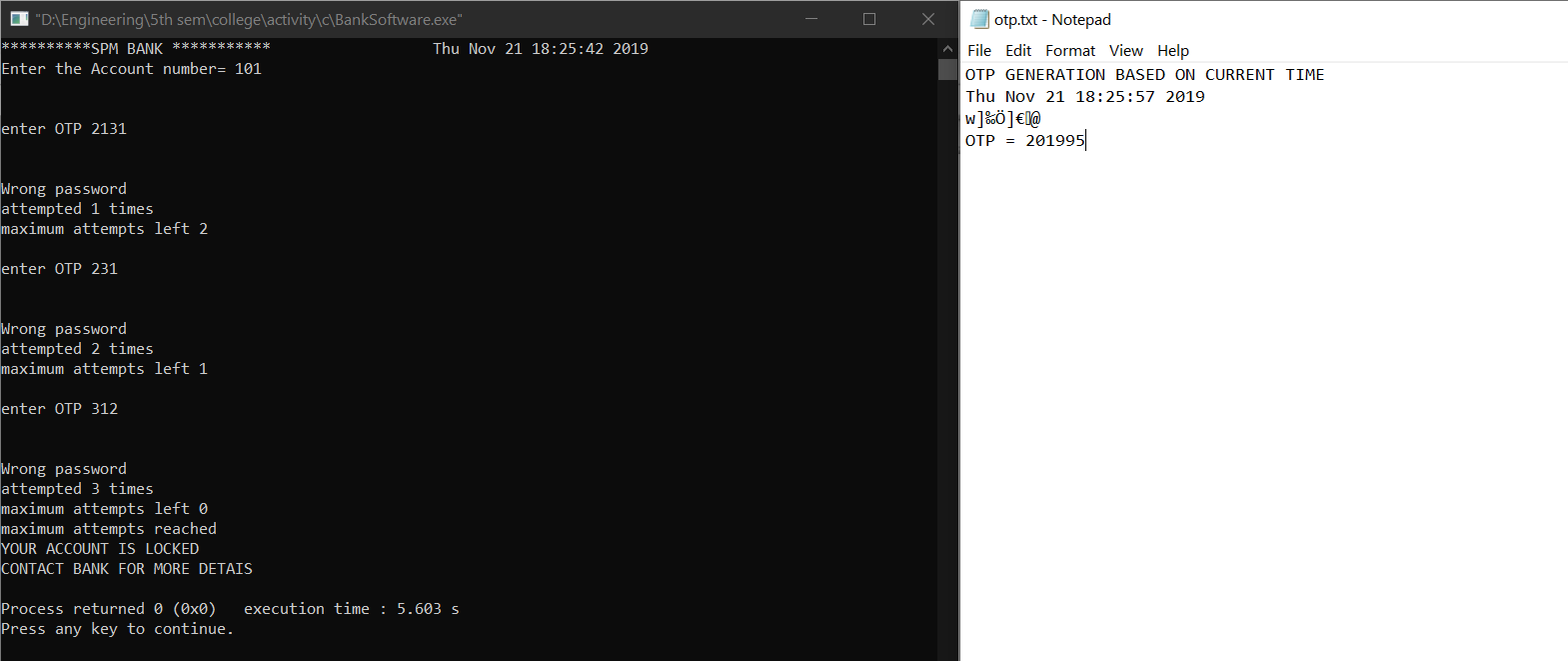
Sample Input/output 1:

Login using OTP : -

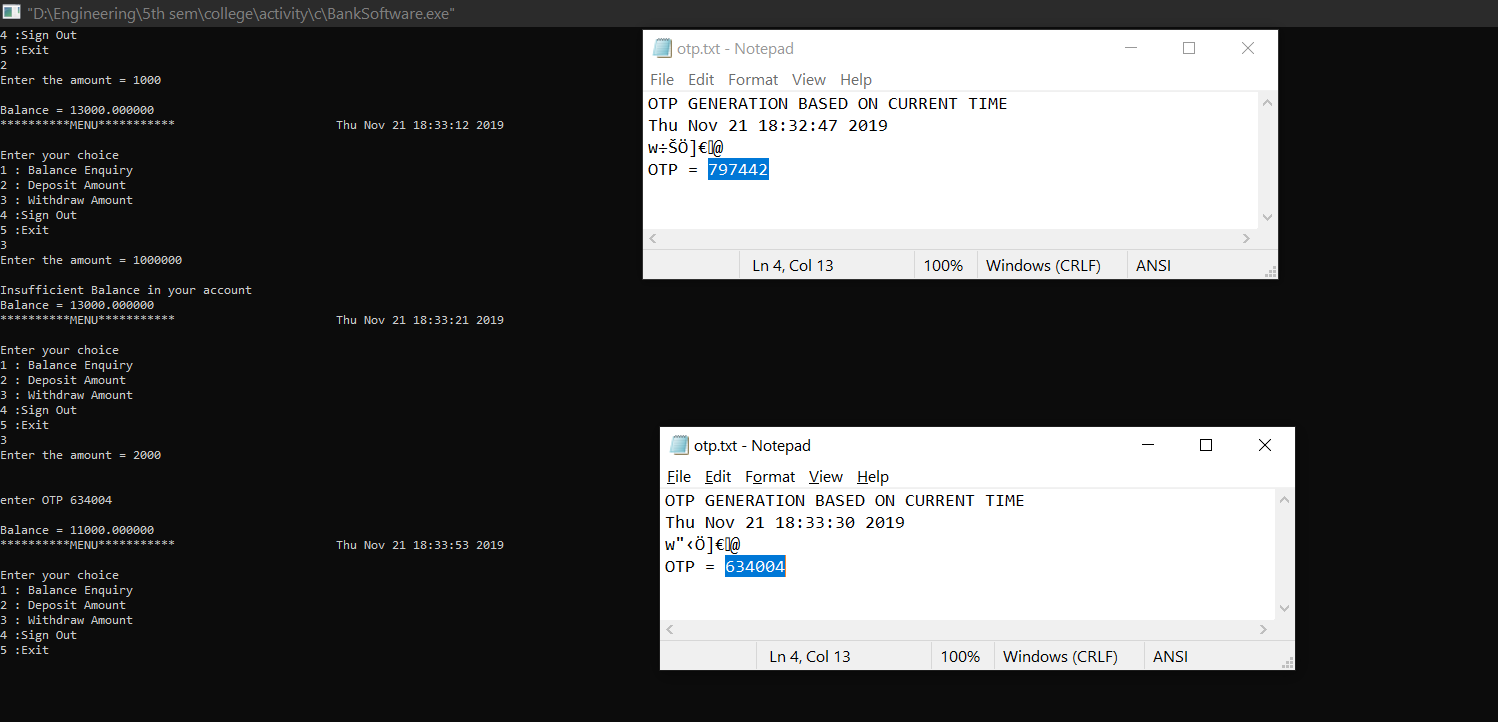


**Sample Input/output 2:**

Invalid login attempts :-



**Sample Input/output 3:**



**Limitations:**

1) The Bank application is restricted to stored account details only.

2) Sometimes OTP length gets reduced due to the addition of numbers while generating OTP because of digit 0 at the end of the hash value of OTP.

**Improvements:**

1) New account opening function can be included.

2) If Balance is not sufficient while withdrawing, Loan function can be provided.

3) Passbook for account holder .

4) Updating information of existing account.