EXP NO:-3

BANK APLLICATION

#include<stdio.h>

#include<stdlib.h>

typedef struct

{

int acc\_no;

char acc\_type;

char name[20];

float balance;

}account;

account customer[10];

account customer[]={{100,'s',"amy",1000.90},

{101,'r',"gail",3000.50},

{102,'c',"marc",5000},

{103,'s',"garry",4000.80},

{104,'s',"cathy",6000.3}

};

float amt;

int arin;

void balance()

{

printf("\nbalance()\n");

printf("\n\nyour account balance is %0.2f",customer[arin].balance);

}

void deposit()

{

printf("\ndeposit()\n");

printf("please enter the amount to be deposited:\n");

scanf("%f",&amt);

customer[arin].balance+=amt;

printf("\nrs.%0.2f is deposited in your account\n",amt);

printf("\n\nthe current balance is %0.2f",customer[arin].balance);

}

void withdraw()

{

printf("\nwithdraw()\n");

printf("\nenter the amount to be withdrawn:");

scanf("%f",&amt);

if(amt>customer[arin].balance)

printf("\nno sufficient balance:\n");

else

{

customer[arin].balance-=amt;

printf("\nplease collect your amount.\n\nThe current balance is %0.2f\n",customer[arin].balance);

}

}

int main()

{

int accnum,choice;

int i,flag=0;

printf("\n------------");

printf("\nwelcome to the KLS-GIT bank");

printf("\n------------");

printf("\nplease enter your 3 digit acount number\n");

scanf("%d",&accnum);

for(i=0;i<5;i++)

{

if (customer[i].acc\_no==accnum)

{

flag=1;

break;

}

else

{

continue;

}

}

if(flag==0)

{

printf("\nsorry !!!!invalid account number\n");

exit(0);

}

arin=accnum%100;

printf("\naccount number:%d\taccount type:%c\t\tname:%s",customer[arin].acc\_no,customer[arin].acc\_type,customer[arin].name);

do{

printf("\n\nplease enter your choice:");

printf("\n\n1.balance enquiry 2.deposit 3.withdraw 4.exit :");

scanf("%d",&choice);

switch(choice)

{

case 1:balance();

break;

case 2:deposit();

break;

case 3:withdraw();

break;

default:exit(0);

};

}while(1);

return 0;

}

**ALGORITHM**

Step 1: Start

Step 2: Declare and define structure “account” with variables “acc\_no”, “acc\_type”, “name[20]” and “balance”

Step 3: Declare variables “amt”, “arin”, “balance” and define array “customer []” with predefined data.

Step 4: Read account number from the user.

Step 5: Check if account number is valid. If not valid print “Invalid Account Number” GOTO 8.

Step 6: Check if account number is valid. Print Account Number, Account Type and Name.

Step 7: Read choice from the user.  
 Case ‘1’ – call “balance” function which prints account balance.

Case ‘2’ – call “deposits” function which deposits funds to current balance.

Case ‘3’ – call “withdraw” function which withdraws funds from current balance

Case ‘default’ – GOTO 8

Step 8: STOP

OUTPUT:-

