EXP NO:-2B

SIMPLE CALCULATOR

ALGORITHM

Step 1: Start

Step 2: Declare and define functions add, subtract, multiply and divide   
Step 3: Declare variables num1, num2, ans and c.

Step 4: Read values num1, num2 and arithmetic operation.

Step 5: Read arithmetic operation.  
 case ‘+’ – call “add” function using pointers which returns (ans=num1+num2)

case ‘-’ – call “substract” function using pointers which returns (ans=num1-num2)

case ‘\*’ – call “multiply” function using pointers which returns (ans=num1\*num2)

case ‘/’ – call “divide” function using pointers which return (ans=num1/num2)

Step 6: Display result

Step 7: Stop

CODE:

#include<stdio.h>

float add(float a,float b) //Defining the addition of two numbers Function

{

{

return(a+b);

}

float sub(float a,float b) //Defining the subtraction of two numbers Function

{

return(a-b);

}

float mul(float a,float b) //Defining the multiplication of two numbers Function

{

return(a\*b);

}

float div(float a,float b) //Defining the Division of two numbers Function

{

return((float)a/b);

}

void display(char choice,float a,float b,float ans) //Defining the displaying the answer Function

{

printf("%.3f %c %.3f is : %.2f",a,choice,b,ans);

}

int main() //main function

{

float a,b,ans;

unsigned char choice;

float (\*ptr)(float,float);

void (\*ptr1)(char,float,float,float); // passing address value

ptr1=display;

printf("Enter the value of a and b :"); //taking two values from user

scanf("%f%f",&a,&b);

pass :

printf("Enter the operation'+';'-';'/' or '\*' to be performed on %.3f and %.3f :",a,b); //asking which operation to be done on those two numbers

{

getchar();

choice = getchar();

switch(choice)

{

case'+' :

ptr=add;

break;

case'-' :

ptr=sub;

break;

case'\*' :

ptr=mul;

break;

case'/' :

ptr=div;

break;

default :

printf("The choice entered is INCORRECT\nPlease enter the correct choice\n"); //printing error message for wrong option.

}

ans = (\*ptr)(a,b);

(\*ptr1)(choice,a,b,ans);

printf("\n");

goto pass;

}

**OUTPUT:-**

