University Name: Shaheed Zulfikar Ali Bhutto Institution of Science and Technology.

Project Name: Scientific Calculator.

Group Members Names: Umair Khan (2012412), Erum Shehzadi (2012378), Sarita Batra (2012406), Chandras Ayush (2012377).

Course Name: Introduction to Computer Science

Supervisors: Sir Sikandar Khan.

**Acknowledgment.**

First and the foremost I would like to thank to my almighty for giving me courage to.

A word of thanks to my teacher, friends, and other source that gave an unending support and helping me in numerous ways from the first stage of project assignment conceived.

I would also like to thank my project group member for their whole hearted support and cooperation.

I duly acknowledge the contribution of group members for invaluable guidance for our project. Coding scientific calculator is an uphill task and would have not been possible without teaching us C language. We thank again sir Sikandar Khan.

I would also thanks to all my friends for forwarding their suggestions to make necessary modifications.

**SPECIAL THANKS TO:** Sir Sikandar Khan for his able guidance in group project assignment.

**ABSTRACT PAGE:**

scientific calculator:

features:

1. arithmetic function (addition, subtraction, multiplication, division).

2. trigonometric function (sin, co, tan, sin-1, cos-1, tan-1).

3. logarithmic function (log, in, ex).

4. number conversion (hexadecimal, octal, binary, decimal).

5. matrix (addition-any size, subtraction-any size, multiplication-only 2x2 and 3x3).

6. random number generation.

7. average of numbers.

8. X raised to the power y (xy).

9. factorial of a number.

10. mean and standard deviation.

deliverable: (working c program that can compile and run-on code block).

this project is about Scientific Calculator which preform 10 main function and this 10 main function have some sub functionin this Scientific Calculator we use libraries, function, loops, assignment operator, if else condition, array, string, switch c+ases ,etc these things help us to create a Scientific Calculator.

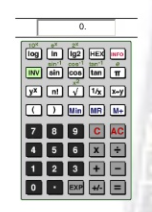
**TABLE OF CONTENTS**

* main body

1. Chapter#1 Introduction ……………………………………. 5 to 7
2. Chapter#2 Background (with algorithms and flowchart) …. 8 to 17
3. Source code ……………………………………. 18 to 27
4. Chapter#3 Working of project (with screenshot) …………. 28 to 32
5. Chapter#4 Scope of project ………………………………. 33
6. Chapter#5 Conclusion ……………………………………... 33
7. Reference ………………………………………34

**CHAPTER#1 INTRODUCTION:**

* SCIENTIFIC CALCULATOR:



the calculator was written by Rolf Howarth in early 1996.

A fully feature scientific calculator with proper operator precedence is implemented, including trigonometric functions and logarithms, factorial, 12 levels of parenthesis, logs to base 2 (a handy function for information entropists!), bitwise logical operators, hex, octal, binary and ASCII display.

* Scientific calculator features:

1. Arithmetic function (addition, subtraction, multiplication, division).
2. Trigonometric function (sin, cos, tan, sin-1, cos-1, tan-1).
3. Logarithmic function (log, in ex).
4. Number conversion (hexadecimal, octal, binary, decimal).
5. Matrix (addition-any size, subtraction-any size, multiplication-only 2x2 and 3x3).
6. Random number generation.
7. Average of numbers.
8. X raised to the power y (xy).
9. Factorial of a number.
10. Mean and standard deviation.

* ADDITION:

The addition (sum function) is used by clicking on the “+” button or using the keyboard. The function result in a+b.

* SUBTRACTION:

The subtraction (minus function) is used by clicking on the “-” button or using the keyboard. The function result in a-b.

* MULTIPLICATION:

The multiplication (times function) is used by clicking on the “x” buttonor using the keyboard “\*” key. The function result in a\*b.

* DIVISION:

The division (division function) is used by clicking on the “/” button or using the keyboard “/” key. The function result in a/b.

* SIGN:

the sign key (negative key) is used by clicking on the “(-)” button. The function result in -1\*x.

* SQUARE:

The square function is used by clicking on the “x^2” button or type “^2”. The function result in x\*x.

* SQUARE ROOT:

The square root function is used by clicking on the “x” button or type “sqrt ()”. The function represents x^.5 where the result squared is equal to x.

* RAISE TO THE POWER:

The raise to the power (y raised to the x function) is used by clicking on the “y^x” button or type “^”.

* NATURAL EXPONENTIAL:

The natural exponential (e raised to the x) is used by clicking on the “e^x” button or type “exp ()”. The result is e (2.71828…) raised to x.

* LOGARITHM:

The logarithm (LOG) is used by clicking on the “LN” button or type “LN ()”.

* NATURAL LOGARITHM:

The nature logarithm (LN) is used by clicking on the “LN” button or type “LN ()”.

* INVERSE:

Multiplication inverse (reciprocal function) is used by clicking on the “l/x” button or typing “inv ()”. This function is the same as x^-1 or dividing 1 by the number.

* FACTORIAL:

The factorial function is used by clicking the “!” button or type “!”.

* PI:

PI is a mathematical constant of the ratio of a circle’s circumference to its diameter.

**CHAPTER# 2 BACKGROUND (WITH ALGORITHM AND FLOWCHARTS):**

* Algorithm:

1.Start

2.Menu for calculator

3.Arithematic trigonometry log number conversion matrix random number average x raised to power y factorial mean and standard deviation

4.Select which you want to perform

5.Arithematic enter 1

6.Select operator on +-/\*

7.Trigonometry enter 2 select operation sin cos tan sin inverse cos inverse tan inverse

8.Log enter 3 select operation 1 log, 2ln, e^x

9.Number conversion enter 4 select operation 1, 2 binaries

10.Matrix enter 5 select operation 1 add ,2 sub, 3multi

11.Range for random enter 6 to take random number

12. Average enter 7 enter total number, add/total

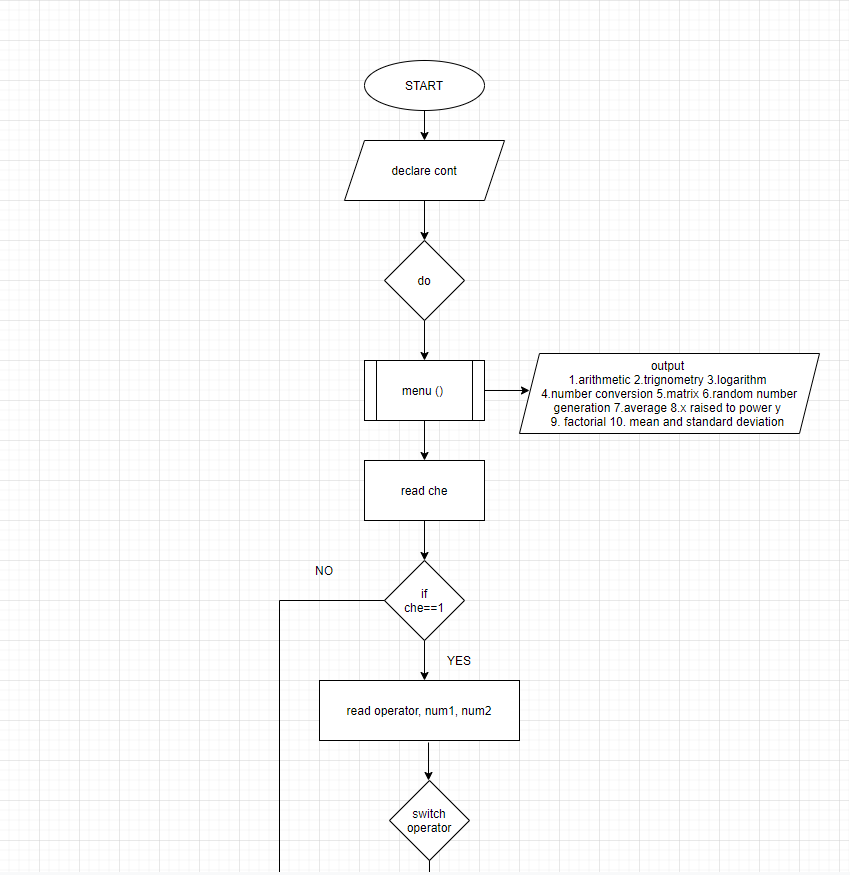
13.X raised to power y enter 8 for base and power

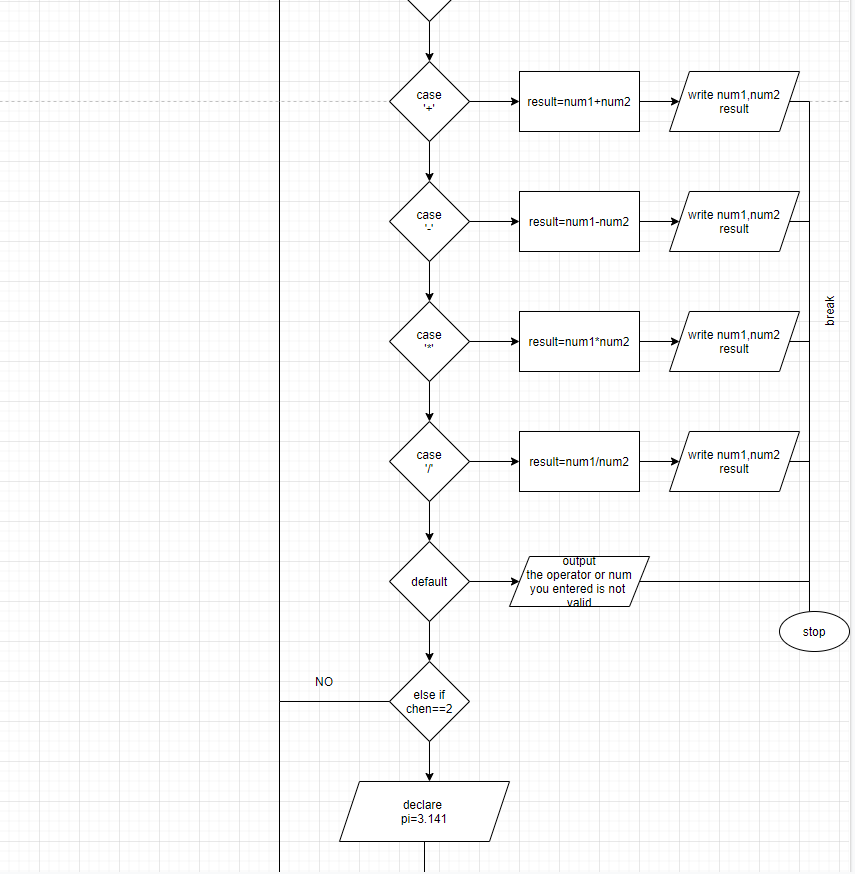
14.Factorial enter 9 for any number factorial

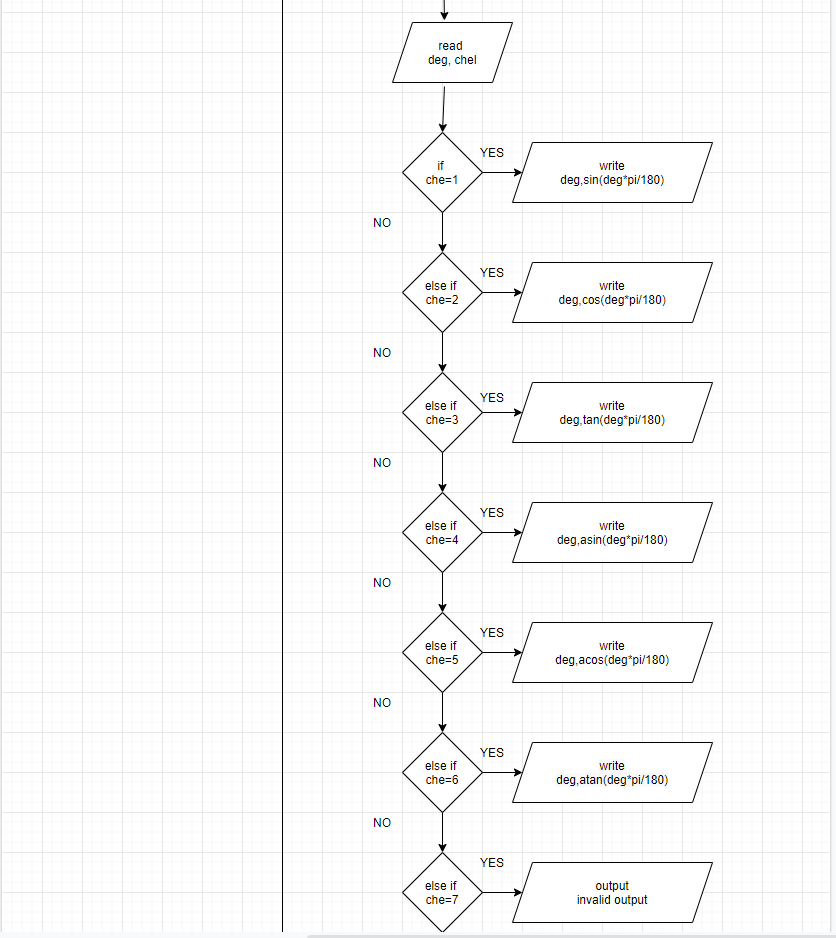
15.Mean and standard deviation enter 10 input total number

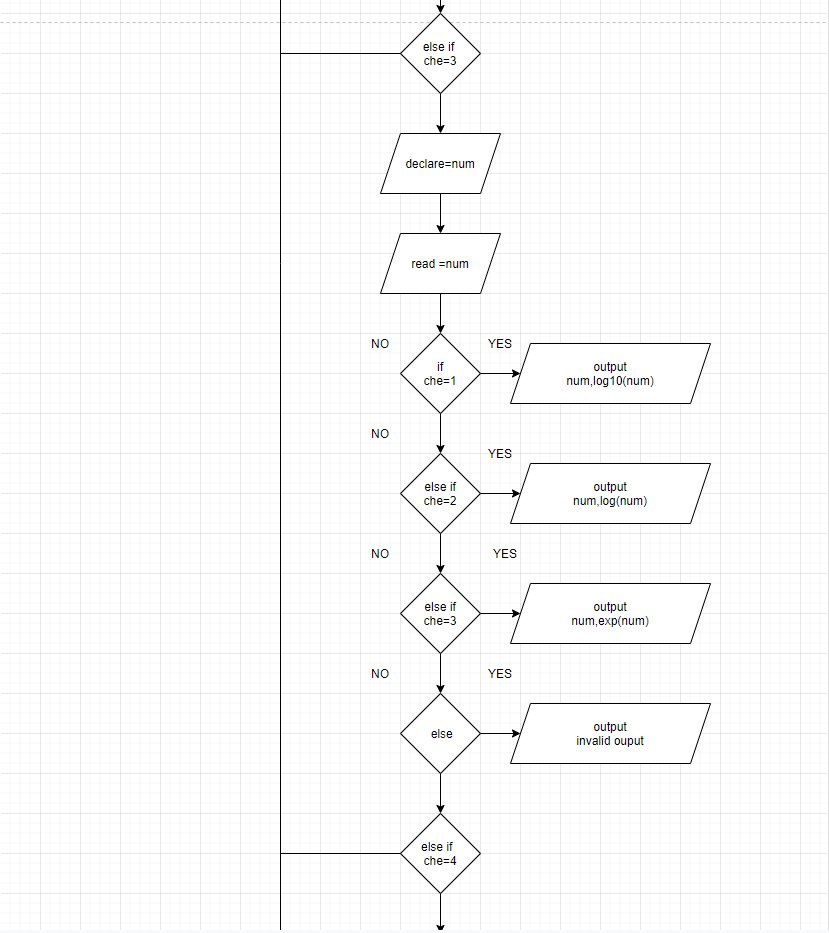
16.Select 1 yes ,2 no to continue

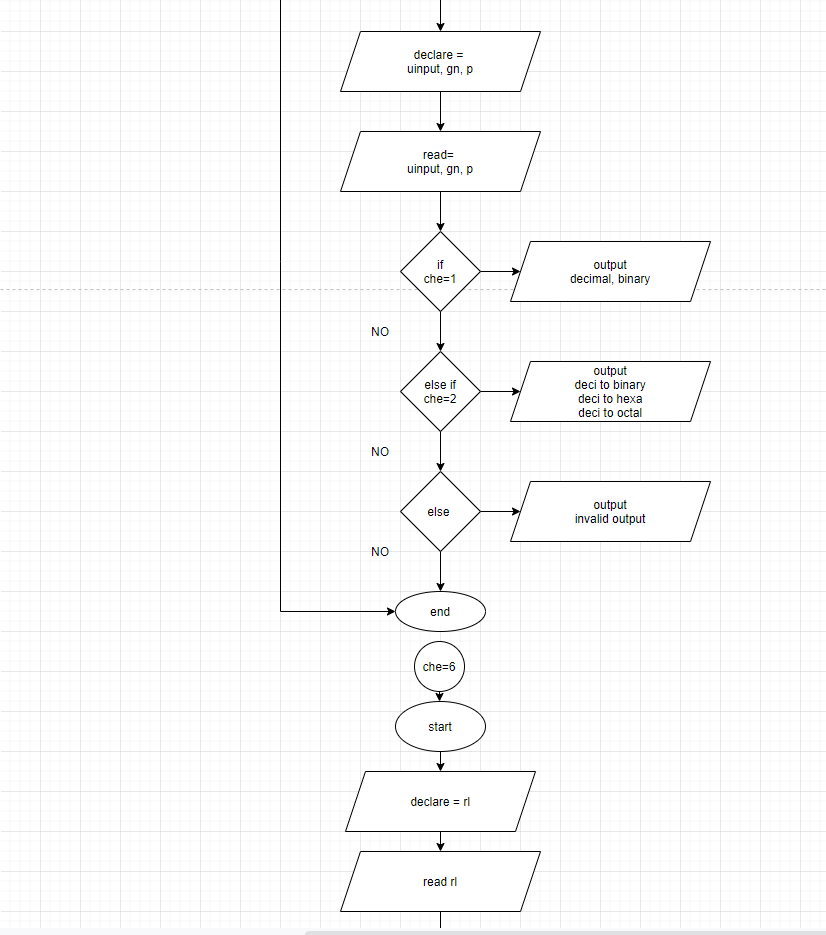
17.stop

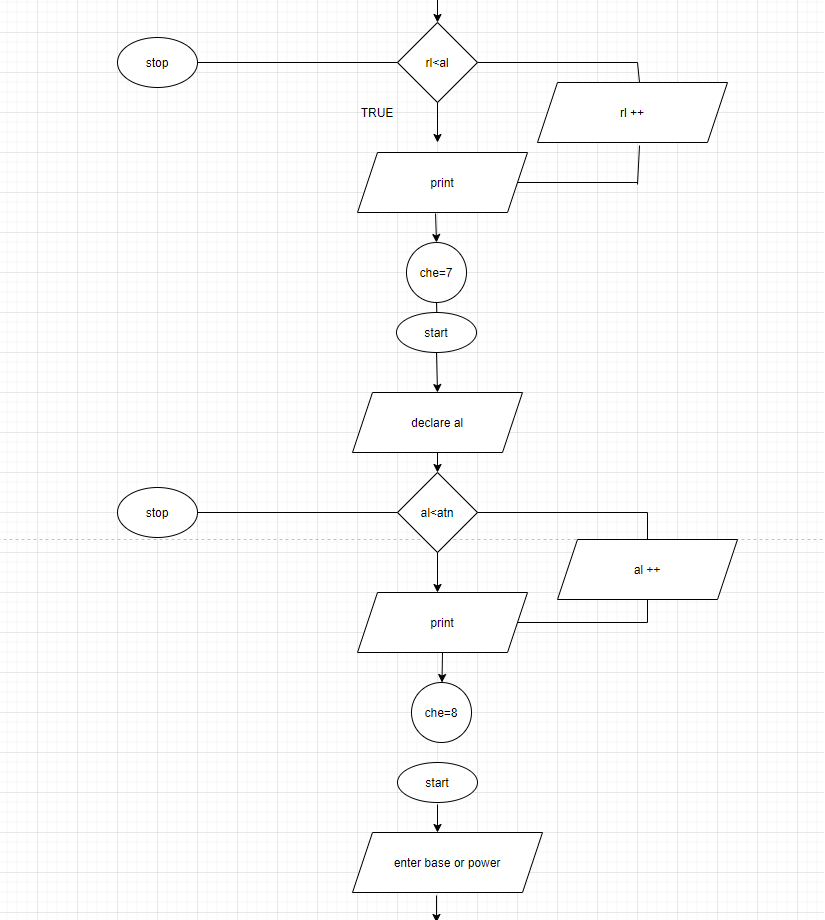


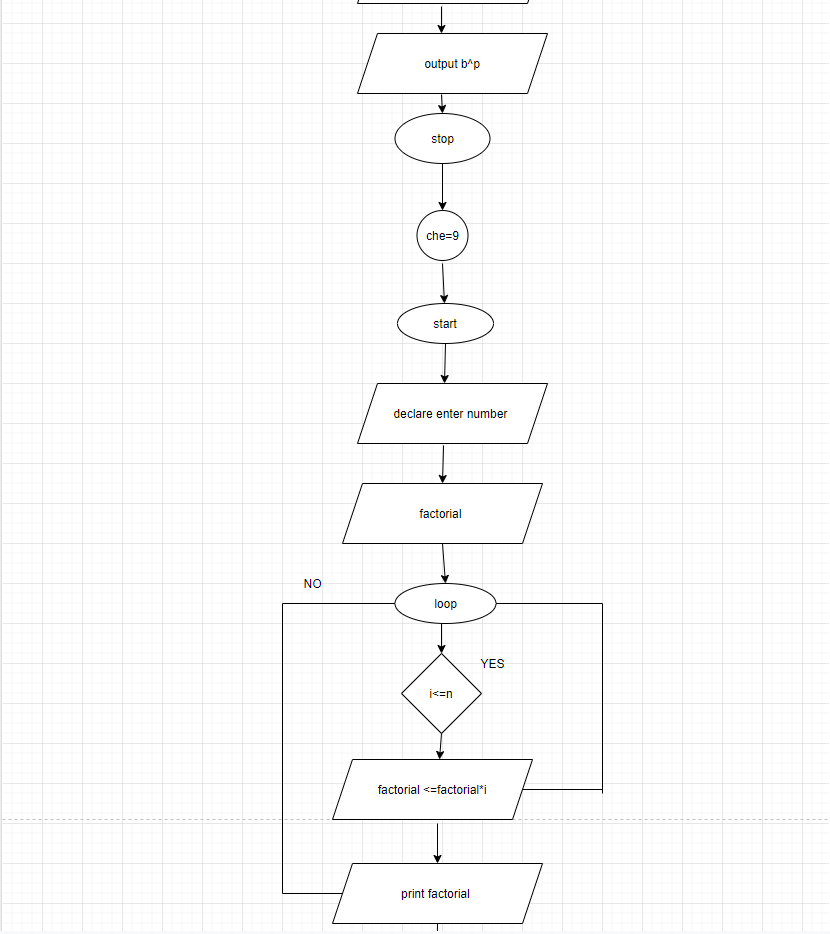


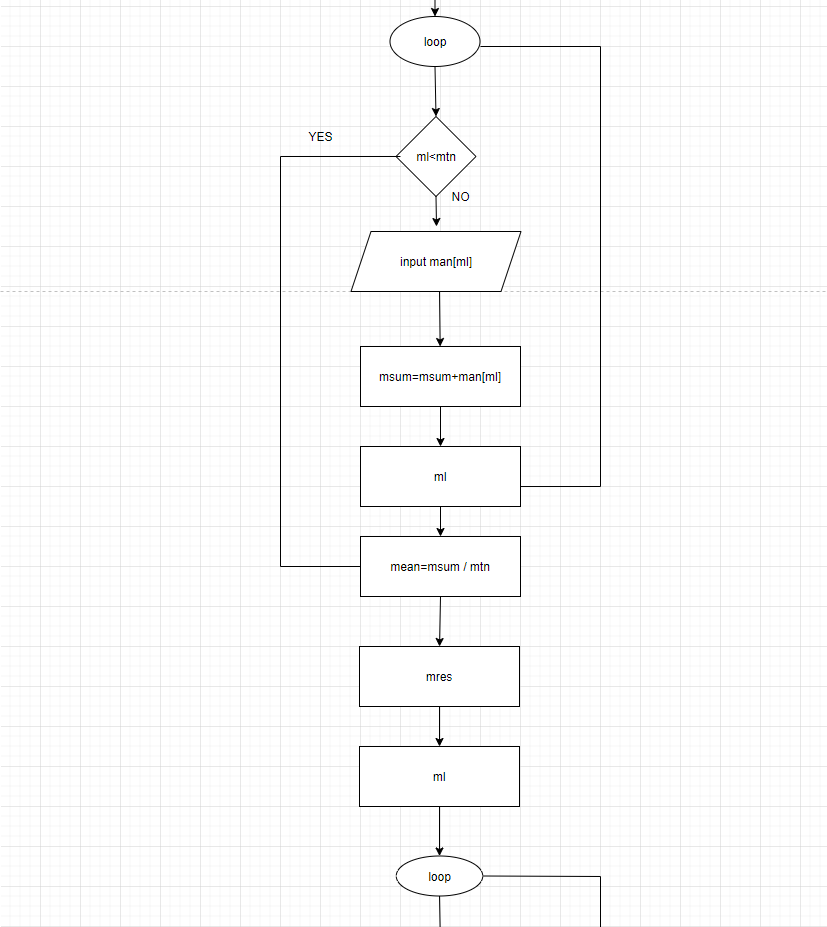


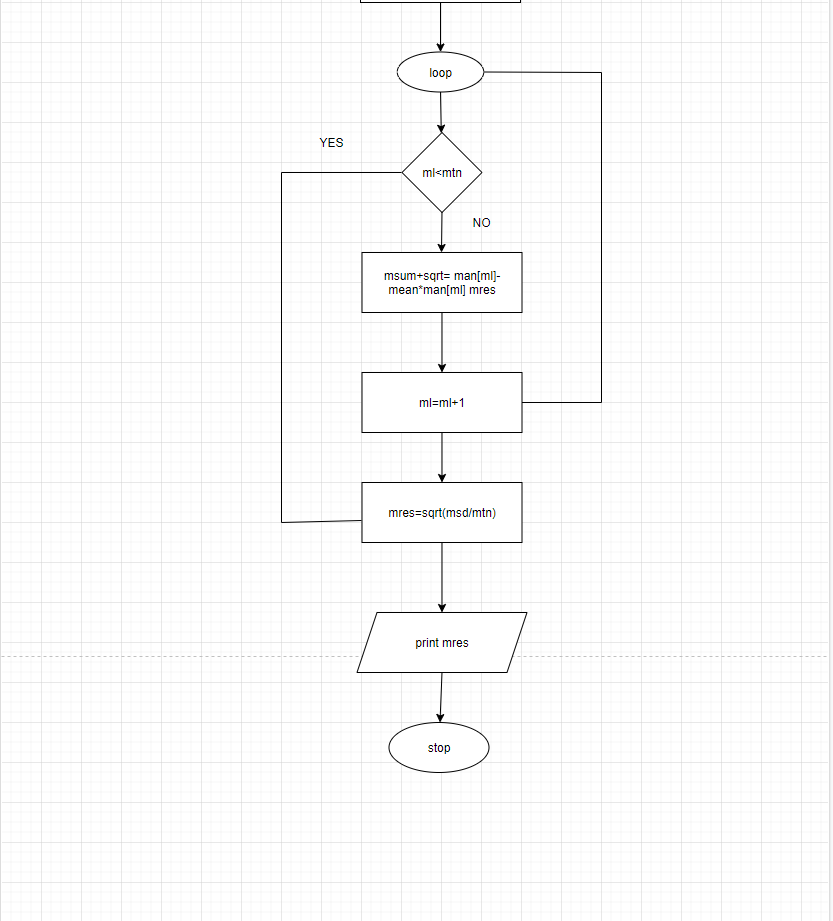












**SOURCE CODE:**

#include<stdio.h>

#include<math.h>

#include<stdlib.h>

#include<time.h>

voidmenu (){

printf("\n\n1) Arithematic\n2) Trignometry\n3) Logarithm function\n4) Number conversion\n5) Matrix\n6) Random number generation\n7) Average\n8) x raised to power y \n9) Foctorial\n10)Mean and standard deviation\n");

}

voidarithmeticmenu(){

printf("\na) Addition\nb) Substraction\nc) Multiplication \nd) Division\n");

}

voiddecitobinary(int no){

int a[100],i;

for(i=0;no>0;i++){

a[i]=no%2;

no=no/2;

}

printf("\n\nBinary form of given no is:\t");

for(i=i-1;i>=0;i--){

printf("%d",a[i]);

}

printf("\n");

}

voiddecitohexa(int nu)

{

intremainder,quotient;

inti=1,j,temp;

charhexanumber[100];

quotient = nu;

while(quotient!=0) {

temp = quotient % 16;

if( temp < 10){

temp =temp + 48;

}

else{

temp = temp + 55;

}

hexanumber[i++]= temp;

quotient = quotient / 16;

}

printf("\nHexadecimal form of given number is : ");

for (j=i-1;j>0;j--)

{

printf("%c",hexanumber[j]);

}

printf("\n");

}

voiddecitocta(int no)

{

intquot,remain;

intoctn[100], i = 1, j;

quot=no;

while(quot != 0)

{

octn[i++]=quot % 8;

quot=quot/8;

}

printf("\nOctal form of given number is :\t");

for (j=i-1;j>0;j--)

{

printf("%d", octn[j]);

}

printf("\n");

}

voidbinarytodesi(intnum){

intrem,sum=0,i=0;

while(num>0) {

rem=num%10;

sum=sum+rem\*(pow(2,i));

num=num/10;

i++;

}

printf("\n\nDecimal value of given number is: %d",sum);

}

longintbinarytohexa(long intnum){

longintremain,i=1,sum=0;

while(num!=0){

remain=num%10;

sum=sum+ remain\*i;

i=i\*2;

num=num/10;

}

return sum;

}

voidbinarytooctal(long intnum )

{

long int octal = 0, j = 1, remain;

while (num != 0)

{

remain = num % 10;

octal = octal + remain \* j;

j = j \* 2;

num = num / 10;

}

printf("\n\nOctal form of given number is : %lo\n", octal);

}

main(){

float cont;

intche;

printf(" \*\*\*\*\*\*\*\*\n WELCOME TO MY CALCULATOR\n \*\*\*\*\*\*\*\*\*\n");

do{

printf("\n MENU\n");

menu();

printf("\nEnter your choice:\t");

scanf("%d",&che);

if(che==1){

float num1, num2, result;

char op;

printf("Enter the operation: ");

scanf(" %c", &op);

printf("\nEnter the first number: ");

scanf("%f", &num1);

printf("Enter the second number: ");

scanf("%f", &num2);

switch(op)

{

case '+':

result = num1+num2;

printf("\nThe sum of %f and %f is: %f",num1,num2,result);

break;

case '-':

result = num1-num2;

printf("\nThe difference between %f and %f is: %f",num1,num2,result);

break;

case '\*':

result = num1\*num2;

printf("\nThe product of %f and %f is: %f",num1,num2,result);

break;

case '/':

result = num1/num2;

printf("\n%f divided by %f is: %f",num1,num2,result);

break;

default :

printf("The operator or number you entered is not valid");

}

}

else if(che==2){

floatdeg,pi=3.141592654;

int che1;

printf("\nEnter your number in degrees\n");

scanf("%f",&deg);

printf("\nSelect Operation\n1) sin\n2) cos\n3) tan\n4) sin-1\n5) cos-1\n6) tan-1\n");

scanf("%d",&che1);

if(che1==1){

printf("The sin of %f is : %f\n",deg,sin(deg\*pi/180));

}

else if(che1==2){

printf("The cos of %f is : %f\n",deg,cos(deg\*pi/180));

}

else if(che1==3){

printf("The tan of %f is : %f\n",deg,tan(deg\*pi/180));

}

else if(che1==4){

printf("The sin inverse of %f is : %f\n",deg,asin(deg)\*(180/pi));

}

else if(che1==5){

printf("The cos inverse of %f is : %f\n",deg,acos(deg)\*(180/pi));

}

else if(che1==6){

printf("The tan inverse of %f is : %f\n",deg,atan(deg)\*(180/pi));

}

else{

printf("Invalid function or operation");

}

}

else if(che==3){

float num;

int che2;

printf("\nENTER A NUMBER\t");

scanf("%f",&num);

printf("\nPlease,select operation:\n1) log\n2) ln\n3) e^x\n");

scanf("%d",&che2);

if(che2==1){

printf("\nlog of %f is %f",num,log10(num));

}

else if(che2==2){

printf("\nln of %f is %f",num,log(num));

}

else if(che2==3){

printf("\nlog of %f is %f",num,exp(num));

}

else{

printf("Invalid number or operation");

}

}

else if(che==4){

intuinput,gn,p;

printf("\nPlease enter the type of input!!!\n1) Decimal\n2) Binary\n");

scanf("%d",&uinput);

if(uinput==1){

printf("\nPlease enter a number you want to convert:\t");

scanf("%d",&gn);

printf("\nEntered value is:\t%d",gn);

decitobinary(gn);

decitohexa(gn);

decitocta(gn);

}

else if(uinput==2){

printf("\nPlease enter a number you want to convert:\t");

scanf("%d",&gn);

printf("\n\nEntered value is:\t%d",gn);

binarytodesi(gn);

printf("\n\nHexadecimal form of given number is: %1X",binarytohexa(gn));

binarytooctal(gn);

}

else{

printf("\n\nGiven instruction is invalid");

}

}

else if(che==5){

int che4;

printf("\nSelect operation\n1) Addition\n2) Substraction\n3) Multiplication\n");

scanf("%d",&che4);

if(che4==1){

int i,j,r,c,m1[10][10],m2[10][10],m3[10][10];

printf("\nEnter no of rows:\t");

scanf("%d",&r);

printf("Enter no of columns:\t");

scanf("%d",&c);

printf("\nEnter elements of first matrix:\t");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("\na%d%d :\t",i,j);

scanf("%d",&m1[i][j]);

}

}

printf("Enter values of second matrix\n");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("\nb%d%d:\t",i,j);

scanf("%d",&m2[i][j]);

}

}

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

m3[i][j] = m1[i][j]+m2[i][j];

}

}

printf("\nThe sum of given matrices is:\n");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("%d ",m3[i][j]);

}

printf("\n");

}

}

else if(che4==2){

int i,j,r,c,m1[10][10],m2[10][10],m3[10][10];

printf("\nEnter no of rows:\t");

scanf("%d",&r);

printf("Enter no of columns:\t");

scanf("%d",&c);

printf("\nEnter elements of first matrix:\t");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("\na%d%d :\t",i,j);

scanf("%d",&m1[i][j]);

}

}

printf("Enter values of second matrix\n");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("\nb%d%d:\t",i,j);

scanf("%d",&m2[i][j]);

}

}

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

m3[i][j] = m1[i][j]-m2[i][j];

}

}

printf("\nThe difference of given matrices is:\n");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("%d ",m3[i][j]);

}

printf("\n");

}

}

else if(che4==3){

int i,k,j,r,c,m1[10][10],m2[10][10],m3[10][10];

printf("\nEnter no of rows:\t");

scanf("%d",&r);

printf("Enter no of columns:\t");

scanf("%d",&c);

printf("\nEnter elements of first matrix:\t");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("\na%d%d :\t",i,j);

scanf("%d",&m1[i][j]);

}

}

printf("Enter values of second matrix\n");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("\nb%d%d:\t",i,j);

scanf("%d",&m2[i][j]);

}

}

for(i=1;i<=r;i++){

for(j=1;j<=c;j++)

{

m3[i][j]=0;

}

}

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

for(k=1;k<=c;k++)

m3[i][j] += m1[i][k]\*m2[k][j];

}

}

printf("\nThe multiplication of given matrices is:\n");

for(i=1;i<=r;i++){

for(j=1;j<=c;j++){

printf("%d ",m3[i][j]);

}

printf("\n");

}

}

else{

printf("Invalid number or operation");

}

}

else if(che==6){

int no1,no2,no3,i,ran;

printf("\nEnter two numbers as a range for random number \t");

scanf("%d %d",&no1,&no2);

printf("\nHow many random numbers you want\t");

scanf("%d",&no3);

for(i=1;i<=no3;i++){

ran=(rand()%(no2-no1+1))+no1;

printf("\n%d",ran);

}

}

else if(che==7){

float sum=0,avg[30];

inttno,i;

printf("\nEnter total number\t");

scanf("%d",&tno);

for(i=1;i<=tno;i++){

printf("\nEnter %d value \t",i);

scanf("%f",&avg[i]);

sum=sum+avg[i];

}

printf("\nThe average of given numbers is %f",sum/tno);

}

else if(che==8){

inti,bse,pow,ans=1;

printf("\nEnter value base\t");

scanf("%d",&bse);

printf("\nEnter power\t");

scanf("%d",&pow);

for(i=1;i<=pow;i++){

ans=ans\*bse;

}

printf("%d raise to %d is :%d",bse,pow,ans);

}

else if(che==9){

intb,i,fact=1;

printf("\nEnter a numbrer of which you you have to find factorial\t");

scanf("%d",&b);

for(i=1;i<=b;i++){

fact=fact\*i;

}

printf("\nFactoroal of %d is :%d",b,fact);

}

else if(che==10){

float mean[20],sum=0.0,mn,sum1=0,sd;

inti,c;

printf("\nEnter total number:\t");

scanf("%d",&c);

for(i=1;i<=c;i++){

printf("\nEnter %d value:\t",i);

scanf("%f",&mean[i]);

sum=sum+mean[i];

}

mn=sum/c;

for(i=1;i<=c;i++){

sum1=sum1+((mean[i]-mn)\*(mean[i]-mn));

}

sd=sqrt(sum1/c);

printf("\nMean of given data is %f",mn);

printf("\nStandard deviation of given data if %f",sd);

}

else {

printf("\n'Invalid choice'");

}

printf("\n\n Do You Want To Continue....\n1) Yes\n2) No\n");

scanf("%f",&cont);

}

while(cont==1);

printf("\n THANKS FOR USING MY CALCULATOR!!!!!");

}

**CHAPTER#3 WORKING OF PROJECT (WITH SCREENSHOTS):**



















**CHAPTER#4 FUTURE SCOPE OF PROJECT:**

Our project will be able to implement in future after making some changes and modifications as we make our project at a very low level.

So, the modification that can be done in our project are:

To make it screen touch so no need to touch key buttons and one more change which can we made is to add snaps of the person who use it.

**Chap 5: Conclusion:**

We can use 10 function in Scientific calculator in console.

1.Arithmetic function (addition, subtraction, multiplication, division).

2.Trigonometric function (sin, cos, tan, sin-1, cos-1, tan-1).

3.Logarithmic function (log, in ex).

4.Number conversion (hexadecimal, octal, binary, decimal).

5.Matrix (addition-any size, subtraction-any size, multiplication-only 2x2 and 3x3).

6.Random number generation.

7.Average of numbers.

8.X raised to the power y (x y).

9.Factorial of a number.

10.Mean and standard deviation.

These functions run separately when the user select any operator in console than the compiler show the answer to user in console.

Scientific calculator is very useful for students who need to do long calculations thus saving their time. The advantage of using scientific calculator is:

1. Time to save.
2. Easy to use.
3. Helps to do long calculations.

**Reference:**

* Let us C. (Yashwant kanetkar).