

# PROJECT REPORT

---

## SCIENTIFIC CALCULATOR

### Group Members

Niraj Vidwans

Mohit Singhal

Tapish Kothari

Dimple Rathore

### INTRODUCTION

The Scientific calculator has various applications in field of science, technology and engineering.

This project is an attempt to make such a calculator using c++ programming language . The calculator performs various operations on the numbers given as input and outputs the result, such as sin, cos, tan, ln, log, etc.

Here we are trying to cover as many functions as possible to make the calculator as much user friendly as we can.

Basically, what we did in our code is that we took graphical input and stored it into a character array. Then we separated it into 2-dimensional character array. Then we processed numbers so that we could store those numbers in a different numbers array which is declared of type double.

We then detected the operators given as input to our character array and found the answer of operation using associative arrays. that is, we defined functions to deduce what operation is to be done on elements of numbers array from the element which has the index between them in the array of strings which we created.

We tried a lot of methods to carry out the parsing and calculations. Things were getting bad. The first method or move which we tried was not very efficient was getting clumsy for some operations. So, we went to our TA, who told us about the Push Pop Mechanism.

But when we switched to the Push Pop Mechanism, it was becoming too complicated to use for the different type of functions we have used in our program.

So, finally by the combined efforts of two of our team mates, namely Niraj and Mohit, we came up with a mechanism for making the correct array and conversion to 2D array. This mechanism worked for us and so, we have used this mechanism for our project.

# FUNCTIONS USED

## 1. Power Function :

This function is used to calculate the  $n$ th power of variable  $x$ .

Note that in this functions you have to use  $n$  as integer only.

For non-integral powers of  $x$  though, we have made a different function, about which we will describe later.

## 2. Factorial Function :

This function is used to calculate the factorial of any natural number  $x$ . This function uses iterative construct to find the factorial. Here  $x$  is given as the input. Note that the given  $x$  should be a natural numbers !!

## 3. sin Function :

This function is use to find the sine (one of the trigonometric operations) of a given  $x$ . In this case, we can use any value of but note that the  $x$  has to be in **Radians**. we have used taylor series to approximate sine to a fairly accurate level.

## 4. cos Function :

This function is use to find the cosine (one of the trigonometric operations) of a given  $x$ . In this case, we can use any value of but note that the  $x$  has to be in **Radians**. here also we have used taylor series.

## 5. tan Function :

This function is use to find the tangent (one of the trigonometric operations) of a given  $x$ . Note that the  $x$  has to be in **Radians**. Here we cannot use the values of  $x$  for which  $\cos$  functions gives zero value.  $\tan$  is calculated by dividing  $\sin$  by  $\cos$ , the two functions are described already above.

## 6. cot Function :

This function is use to find the cotangent (one of the trigonometric operations) of a given  $x$ . Note that the  $x$  has to be in **Radians**. Here we cannot use the values of  $x$  for which  $\sin$  functions gives zero value it is calculated similar to  $\tan$  just by changing roles of  $\sin$  and  $\cos$ .

## 7. sec Function :

This function is use to find the secant (one of the trigonometric operations) of a given  $x$ . Note that the  $x$  has to be in **Radians**. Here we cannot use the values of  $x$  for which  $\cos$  functions gives zero value. It is basically calculated as the reciprocal of the  $\cos$  function.

## 8. cosec Function :

This function is use to find the cosecant (one of the trigonometric operations) of a given  $x$ . Note that the  $x$  has to be in **Radians**. Here we cannot use the values of  $x$  for which  $\sin$  functions gives zero value. It is basically calculated as the reciprocal of the  $\sin$  function.

## 9. arcsin Function :

This function is one of the inverse trigonometric functions. It gives back the angle for which the sin will be  $x$ , the variable we have to pass as input. Note that the  $x$  should be between  $-1$  to  $+1$  since only then the inverse function of sin will be meaningful.

#### 12. $e^x$ Functions :

It is a type of the power function in which the base is an irrational number 'e' whose value is approximately 2.71828. There are no conditions on the variable  $x$ . We have used the Taylor series here also to reach the correct value of  $e^x$ .

#### 10. sinh Function :

It is a hyperbolic function. It will give values on giving any  $x$ . It is calculated by using  $e^x$  function described above.

#### 11. cosh Function :

It is also one of the hyperbolic functions. Here also there are no conditions on the variable  $x$ . It is calculated similar to sinh.

#### 16. tanh Function :

It is also one of the hyperbolic functions. Here also there are no conditions on the variable  $x$ .

#### 13. $\ln x$ Function :

This is the logarithmic function having base 'e'. Note that the input variable  $x$  has to be greater than zero. Here again we have used Taylor series.

#### 14. log x Function :

This function is also a logarithmic function having base 10.

Here also, the variable  $x$  should be strictly more than zero. it is calculated by dividing  $\ln x$  by  $\ln 10$ .

#### 15. Quadratic Function :

This is a traditional quadratic function. In this we will have a typical quadratic equation of the form  $ax^2+bx+c=0$ . Here we have written a code which will compute the roots of this quadratic equation. but this code runs on console screen and not the graphics screen.

#### 17. nroot Function :

It is the function made to find the  $n$ -th root (power  $1/n$ ) of any number. we have used a simple yet smart trick for calculating this function. we used the fact that

$a^b = e^{(b \ln a)}$ . this ,in fact does all the power operations needed. but this method takes little more time.

#### 18. Cube Root function :

Since the cube root is a more frequently used function, so we have made a seperate program to find the cube root of any number  $x$ .

### 19. Square Root function :

Since the square root is a more frequently used function, so we have made a separate program to find the square root of any number  $x$ . Here note that the variable  $x$  should take non-negative values.

### 20. Square function :

It becomes tedious to use the power function again and again for as frequent as square. Hence we have separated the square function so that the square of any no.  $x$  can be calculated more quickly.

### 21. Cube function :

Same as square function, we have created a separate function to calculate the cube of any number.

### 22. Integration of Sin :

For calculating the integral of  $\sin x$  we used the fact that the integral of  $\sin$  is  $\cos$ . So, we just called for input of the limits from the user and put those limits in ' $-\cos$ ' function to calculate the integral.

### 23. Integration of $\cos x$

For calculating the integral of  $\cos x$  we used the fact that the integral of  $\cos$  is  $\sin$ , so all we had to do was to call the  $\sin$  function from our program and just put the limits in that  $\sin$  function to calculate the value of our integral.

### 24. Integration of $\tan x$

We used the fact that the integral of  $\tan$  is  $\ln(|\sec x|)$ . So first we called  $\sec$  function and put the resulting value in a temporary variable whose value is equal to the expression  $\sec x$  and then call  $\ln$  function followed by putting the temporary variable in  $\ln$  function and applying the limits.

#### 25. Integration of $\sec x$

We used the fact that the integral of  $\tan$  is  $\ln(|\sec x + \tan x|)$ . So first we called  $\sec$  and  $\tan$  function and put the resulting value in a temporary variable whose value is equal to the expression  $\sec x + \tan x$  and then call  $\ln$  function followed by putting the temporary variable in  $\ln$  function and applying the limits.

#### 26. Integration of $\operatorname{cosec} x$

We used the fact that the integral of  $\tan$  is  $-\ln(|\operatorname{cosec} x + \cot x|)$ . So first we called  $\operatorname{cosec}$  and  $\cot$  function and put the resulting value in a temporary variable whose value is equal to the expression  $\operatorname{cosec} x + \cot x$  and then call  $\ln$  function followed by putting the temporary variable in  $\ln$  function and applying the limits.

#### 27. Integration of $\cot x$

We used the fact that the integral of  $\tan$  is  $\ln(|\sin x|)$ . So first we called  $\sin$  function and put the resulting value in a temporary variable whose value is equal to the expression  $\sin x$  and then call  $\ln$  function followed by putting the temporary variable in  $\ln$  function and applying the limits.

#### 28. Integration of $\sec^2 x$



For calculating the integral of  $\tan x$  we used the fact that the integral of  $\sec^2$  is  $\tan$ , so all we had to do was to call the  $\tan$  function from our program and just put the limits in that  $\tan$  function to calculate the value of our integral.

## 29. Integration of $x^p$

This function calculates the integral of the  $x$  to the power  $n$ . In this use have two conditions :

a) When  $n = -1$ ;

In this case, we know that the integral equals  $\ln x$ , so we just put the limits in the  $\ln x$  function(which we call during calculation) to compute the answer.

b) When  $n \neq -1$ ;

This case is simple. We know that the integral equals  $x^{(n+1)}/(n+1)$ , so we just call the power function to compute  $x^{(n+1)}$  and divide it by  $n+1$ ; followed by putting the limits.

## 30. Integration of $e^x$

This function calculates the integral of  $e^x$ . In this we use the fact that the integral of  $e^x$  is itself. So all we do is that we call the function  $e^x$  and put the limits in that function

## 31. Integration Of $\ln x$

We simply use the integral of  $\ln x$ , which we already know to be  $x \ln x - x$ , in the function to calculate the integral. All we did was to call the function 'ln' and then apply the limits.

# Creating the calculator

First of all we had made a sketch of calculator ,how the calculator will look like and what are the positions of different functions. most important what will be the coordinates of the rectangles used for creating the buttons . then we created our buttons for different functions used or performed by our calculator. then we entered the text related to each block in the buttons. like the button which is used for calculating sin ,we have entered sin in that. then doing all that we made a separate box where we can expression or input will be shown and then a separate box for the answer one.

we entered on the input part ,like when the button will be clicked it will show the clicked part. so we done it with the help of `getClick()` option .but then one new problem came that as soon as we press the new button the previous one will be deleted and so we think about to solve the problem ,then we found the answer and we stored it in a array named `ARRAY` and at last we print the complete array in the input box.

then thats how our input taking problem was solved .

side by side our functions was ready and we were checking them that they are working well . but the main problem now arises in front of us that how to give the input to compiler

through the canvas or the calculator .then we asked our TA for help .he suggested us a method for parsing . his method was good but difficult . he suggested that first convert an expression to a post fix expression . the problem was that it was very difficult to convert an expression to a post expression and then there was a method to evaluate thte answer from that expression .then for simplicity we assumed that the expression is given in a postfix form and then we put them one by one in a stack and then calculated as the operators were coming .

the main problem with the idea was that

1. no bodmass was applied,
- 2.difficult to get a postfix expression.
- 3.method only applicable to single digit values.

then we were working on this problem and an idea came again that we used files handling in reading the values ,then we succeeded in that and it was giving us the simple calculations .

but it was not at all user friendly and it was facing problem with the typical calculations but still we were bounded as we were not having any choice in our hand.

then two days of hard work gave us a method .

in this we first store our input in the array and then check our complete input one by one. it checks the first element and according to it ,it stores it in a new 2 dimensional array 'array1' which is like in one row either one number is stored or an operator.

Then the numbers in char array are converted to double numbers using sscanf and stored in array of double variables.

These variables are having same indices as their char counterparts in array1.these are then processed by the operators with indices between them.

## CONTRIBUTIONS by individual MEMBERS

### Niraj

He created the functions for conversion, quadratic equation, nth power, nth root, when n is any number, solving system of linear equation.

He also made rudimentary mechanism for input output

Which was not very useful but his idea of converting char array into double number was used in final implementation.

### MOHIT SINGHAL:

He created the complete graphics part of the calculator including the design, created functions for the project like arcsin, ln x, arcos, combination, power,.

He made an good contribution in taking input from user through calculator, solving the input taken as string to make it understand to the compiler through the idea of array conversion, included the BODMAS part,

### TAPISH KOTHARI

He wrote functions for all trigonometric functions, inverse

Trigonometric functions, integrations, ln x, log x, etc.

**Dimpal**

She wrote function for cube square, and tried to make  $e^x$ .

