**SCIENTIFIC CALCULATOR**

**PROJECT REPORT**

***Submitted by***

**SANKET SUHAGIYA (ROLL NO. 59)**

**RITHWIK VEDPATHAK (ROLL NO 66)**

**SIDDHANT VASAIKAR (ROLL NO. 65)**

*To*

*The Mumbai University*

*in partial fulfillment of the requirements for the second year of degree of*

*Bachelor of Computer Engineering*

****

**DEPARTMENT OF COMPUTER ENGINEERING**

**VIDYAVARDHINI’S COLLEGE OF ENGINEERING AND TECHNOLOGY**

**K. T. MARG, VASAI ROAD (W) DIST-THANE, PIN:401202(Affiliated to University of Mumbai )**

**2019-2020**

**VIDYAVARDHINI’S COLLEGE OF ENGINEERING AND TECHNOLOGY**

**K. T. MARG, VASAI ROAD (W.) DIST-THANE, PIN:401202**

**DEPARTMENT OF COMPUTER ENGINEERING**

****

**CERTIFICATE**

*This is to certify that the mini project report for the subject of “Data Structures” entitled* **“SCIENTIFIC CALCULATOR”** *is submitted by* **”SIDDHANT VASAIKAR(65)”,”RITHWIK VEDPATHAK(66)”,“SANKET SUHAGIYA(59)”** *in partial fulfilment of the requirements for the award to the second year of the degree of Bachelor of Engineering in Computer Engineering to the Mumbai University as a record of work done by him/her under our supervision and guidance.*



**Prof. Sweety Rupani**

**Project Guide**

**ACKNOWLEDGEMENT**

It gives us a great pleasure to express our deep sense of gratitude and indebtedness to our guide **Prof. Sweety Rupani** for her valuable support and encouraging mentality throughout the project. We are highly obliged to her for providing us this opportunity to carry out her ideas and work during our project period and helping us to gain the successful completion of our Project.

Our special thanks is going to Head of the Department of Computer Engineering of our college, **Dr. Megha Trivedi** and to all of the faculties for allowing us to come here and encouraging us to constantly work hard in this project.

We are highly grateful to the Honorable Principal of VCET, **Dr. H. V. Vankudre** for giving us this golden opportunity to be a part of this organization for this period.

**ABSTRACT:**

The main objective of this project is to develop a basic scientific calculator using JAVA programming language.

This Scientific Calculator is based on various operational levels. Those are Standard functions, Scientific functions and Statistical function. Under these three main categories all the basic functions which are related like addition, subtraction, multiplication, division, hyperbolic and circular trigonometric functions, round off, logarithmic functions, etc. Interface is designed according to a very user friendly manner with gray-tone colors.

It is a calculator designed to help you calculate science, engineering, and mathematics problems. It has way more buttons than your standard calculator that just lets you do your four basic arithmetic operations of addition, subtraction, multiplication, and division.

All of these extra buttons allow you to work with various kinds of numbers and equations. Here user can access their past written equations or obtained answers in the history tables which makes use of a queue as data structure.

For converting the equation obtained from user into actual infix form we’ve used stack implementation. For solving nested bracket equations a recursive function named “god” is used which when recognizes ‘(’ solves for the equation between ‘(’ and ‘)’ and returns the obtained answer back to the original equation.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr No** | **Topic** | **Page No** |
|  | **Problem Definition** | **2** |
|  | **Scope of Project** | **2** |
|  | **Sample Code** | **3** |
|  | **Result** | **10** |
|  | **Conclusion** | **10** |
|  | **References** | **11** |

**PROBLEM DEFINITION**

A scientific calculator is a type of [electronic](http://en.wikipedia.org/wiki/Electronics) [calculator](http://en.wikipedia.org/wiki/Calculator), usually but not always handheld, designed to calculate problems in science (especially [physics](http://en.wikipedia.org/wiki/Physics)),[engineering](http://en.wikipedia.org/wiki/Engineering), and [mathematics](http://en.wikipedia.org/wiki/Mathematics). They have almost completely replaced [slide rules](http://en.wikipedia.org/wiki/Slide_rule) in almost all traditional applications, and are widely used in both education and professional settings.

In certain contexts such as higher education, scientific calculators have been superseded by [graphing calculators](http://en.wikipedia.org/wiki/Graphing_calculator), which offer a superset of scientific calculator functionality along with the ability to graph input data and write and store programs for the device. There is also some overlap with the financial calculator market.

**SCOPE OF PROJECT**

Scientific calculators are used widely in any situation where quick access to certain mathematical functions is needed, especially those such as trigonometric functions that were once traditionally looked up in tables; they are also used in situations requiring back-of-the-envelope calculations of very large numbers, as in some aspects of [astronomy](http://en.wikipedia.org/wiki/Astronomy), [physics](http://en.wikipedia.org/wiki/Physics), and [chemistry](http://en.wikipedia.org/wiki/Chemistry).

They are very often required for math classes from junior high school level through college, and are generally either permitted or required on many [standardized tests](http://en.wikipedia.org/wiki/Standardized_test) covering math and science subjects; as a result, many are sold into educational markets to cover this demand, and some high-end models include features making it easier to translate the problem on a textbook page into calculator input, from allowing explicit [operator precedence](http://en.wikipedia.org/wiki/Operator_precedence) using parentheses to providing a method for the user to enter an entire problem in as it is written on the page using simple formatting tools.

**SAMPLE CODE**

class god {

String god(String s) {

final int MAX = 16;

final int MIN = -6;

String temp = "";

int l = s.length();

String infix[]=new String[100]; // infix string

int j=0,br=0;

Boolean flag = true; //sends the things in brackets to temp for calling god

Boolean ignore = true;//to ignore brackets "(" and ")"

Boolean ignore2 = true;//to accept negative inputs based on few assumptions

//boolean ignore3 = true;

//converts input to infix

for(int i=0 ; i < l ; i++ , ignore = true, ignore = true) {

char ch = s.charAt(i);

//infix space ready

if(infix[j] == null) {

infix[j] = "";

}

//brackets detect

if(ch=='(') {

int initial = br;

br++;

flag = false;

if(initial == 0 && br == 1) {//toggles from 0 to 1

ignore = false;

temp = "";

}

}

else if(ch == ')') {

br--;

}

// unnary negative

if(s.charAt(i) == '-') {

if(i == 0 || (s.charAt(i-1) == '+' || s.charAt(i-1) == '-' || s.charAt(i-1) == '\*' || s.charAt(i-1) == '/' || s.charAt(i-1) == '^'))

ignore2 = false;

}

//differentiates opperator and operand

if((flag) && (ignore2) && (s.charAt(i) == '+' || s.charAt(i) == '-' || s.charAt(i) == '\*' || s.charAt(i) == '/' || s.charAt(i) == '^')) {

j++;

infix[j] = "";

infix[j] += s.charAt(i);

j++;

}

else {

//calls god on bracket closure

if(br == 0 && !flag) {

System.out.println(temp);

temp = god(temp);

infix[j] += temp;

ignore = false;

flag = true;

}

if(ignore) {

temp += ch;

}

if(flag && ignore) {

infix[j] += ch;

}

}

}

//prints infix in [] brackets

int i = 0;

while(infix[i] != null) {

System.out.print("[" + infix[i] + "]");

//if(infix[i].equals(pi))

i++;

}

System.out.println("\n");

//infix to postfix

i = 0;

Stack o = new Stack();

o.push("$");

Postfix post = new Postfix();

String[] pf= new String[100];

int k = 0;

int inst, incm;

while(infix[k] != null) {

boolean a = infix[k].equals(Character.toString('+'));

boolean b = infix[k].equals(Character.toString('-'));

boolean c = infix[k].equals(Character.toString('\*'));

boolean d = infix[k].equals(Character.toString('/'));

boolean e = infix[k].equals(Character.toString('^'));

//checks if operator present

if(a || b || c || d || e){

inst = post.choice((String) o.peek());

incm = post.choice(infix[k]);

//System.out.print(inst +" "+ incm);

while(incm <= inst) {

pf[i] = (String) o.pop();

inst = post.choice((String) o.peek());

i++;

}

o.push(infix[k]);

}

else {

pf[i] = (infix[k]);

i++;

}

k++;

}

//completes postfix by making opertor stack empty

inst = post.choice((String) o.peek());

while(inst != 0) {

pf[i] = (String) o.pop();

inst = post.choice((String) o.peek());

i++;

}

//prints infix

i = 0;

System.out.print("Infix: ");

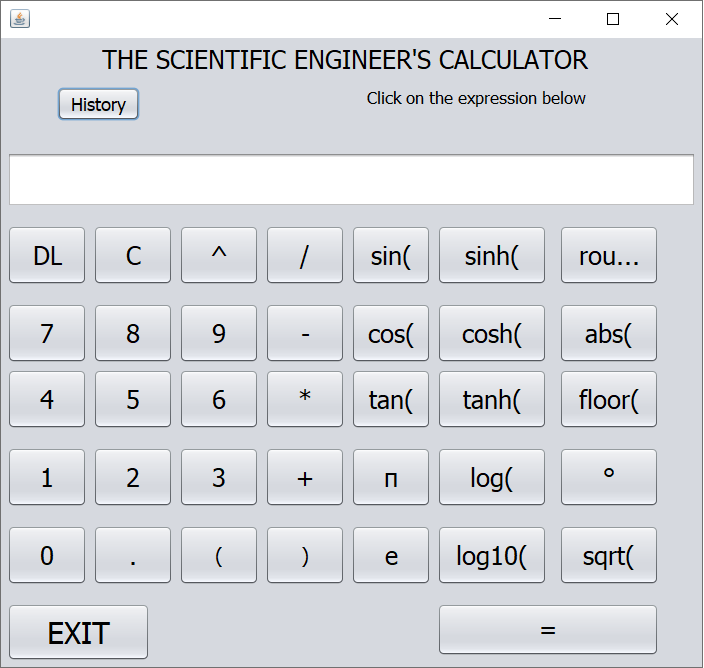
while(infix[i] != null) {

System.out.print("(" + infix[i] + ")");

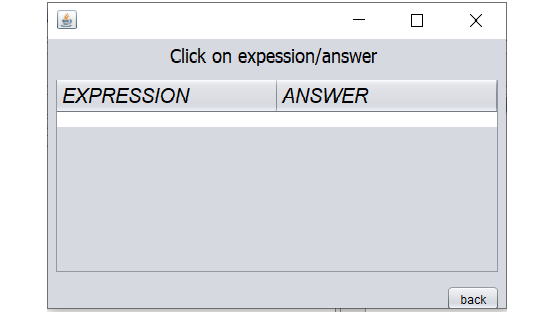
i++;

}

// end of infix to postfix

****

**CALCULATOR INTERFACE**

****

**HISTORY TABLE**

**RESULT**

This calculator offers hassle free and time-efficient way to calculate for each subject over traditional paper-based method. This is very helpful for people of all age to enter the desired information through so much simplicity. You can also view history by clicking in history button which displays previous answers and equations you’ve used.

The project takes basic idea of a scientific calculator turning it into software implementation. The front end of the project makes use of java swing and awt whereas its back end makes use of various data structures like stack and queue.

The flow of the project is easy to understand. The user must type their equation get answer and if needed to recover their previous answers and equations ...they can do so by going through history table included. Just by clicking on the equation or answer user can get it on the main text field.

Thus project fulfills all the requirements needed to successfully maintain attendance records of the students of the college, in an efficient, user-friendly and secure manner.

**CONCLUSION**

It has been a great pleasure for us to work on this project. This project provided us with practical knowledge of creating our own Calculator. It also provided us with an experience of handling all the procedures related to ‘Scientific Calculator’.

While working on the project, we gained knowledge about various software used for web development. The project helped to nurture our mindset what thinking is required to develop a project in a real sense which has an application in the real world. We also got an idea about what problems or hindrances can occur during project development and discovered ways to tackle them and resolve them. The project also gave an opportunity to us for working together as a team, thus providing us with a useful experience in working in a project management team.

This project will provide better opportunities and guidance in future in developing projects independently.

**REFERENCES**

1.[w3schools.com](http://w3)

2.[javatpoint.com](https://www.javatpoint.com/java-tutorial)