SOEN 6011- SOFTWARE ENGINEERING PROCESSES
PROJECT DELIVERABLE 2

Hetvi Shah (40089272)

Github: https://github.com/Hetvishah1710/tanx

July 29 2019

Contents

0.1 Problem 4

0.1.1 Implementation of Tan Function

The tan value of a given number x is sin value by cos value. There are many different ways to implement tan functions such as using MaclaurinâĂŹs Series, iteration method etc. I have implemented tan function using MaclaurinâĂŹs Series by implementing sin and cos value through it and then finding tan value using its formula without using any built-in functions which is mathematically represented.

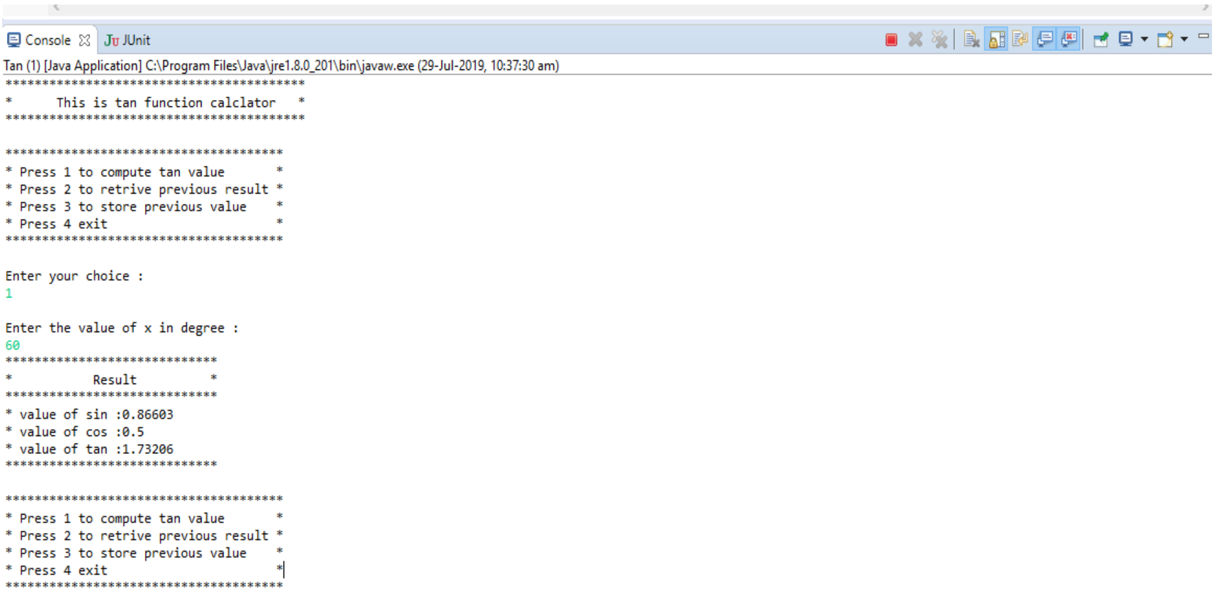
$$sin = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \ldots.$$

and cos would be ;

$$cos = x - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \ldots.$$

So tan can be derived as,

$$tan = \frac{sin}{cos}$$

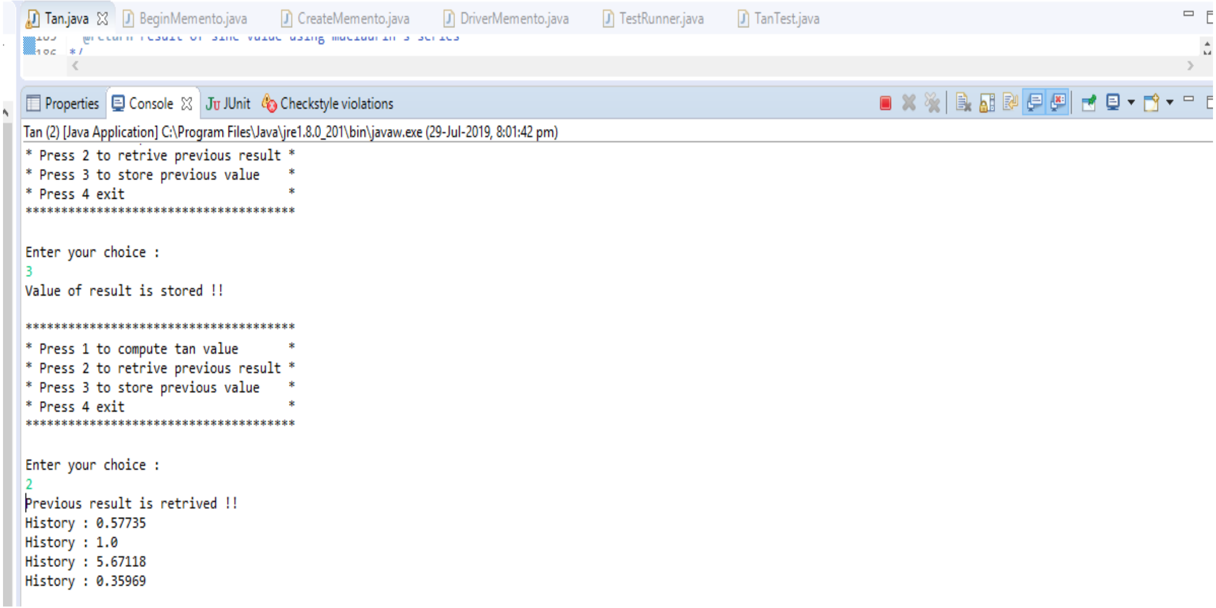


```
Tan (1) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (29-Jul-2019, 10:37:30 am)
*****************************************
*      This is tan function calclator   *
*****************************************

**************************************
* Press 1 to compute tan value       *
* Press 2 to retrive previous result *
* Press 3 to store previous value    *
* Press 4 exit                       *
**************************************

Enter your choice :
1

Enter the value of x in degree :
60
*****************************
*           Result          *
*****************************
* value of sin :0.86603
* value of cos :0.5
* value of tan :1.73206
*****************************

**************************************
* Press 1 to compute tan value       *
* Press 2 to retrive previous result *
* Press 3 to store previous value    *
* Press 4 exit                       *
**************************************
```

Figure 1: textual User interface of implementation

Memento Pattern

I have used Memento software design pattern which provides an ability to restore an object to its previous state.Below figure shows the History for the values of tan functions calculated before.



```
Tan (2) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (29-Jul-2019, 8:01:42 pm)
* Press 2 to retrive previous result *
* Press 3 to store previous value    *
* Press 4 exit                       *
**************************************

Enter your choice :
3
Value of result is stored !!

**************************************
* Press 1 to compute tan value       *
* Press 2 to retrive previous result *
* Press 3 to store previous value    *
* Press 4 exit                       *
**************************************

Enter your choice :
2
Previous result is retrived !!
History : 0.57735
History : 1.0
History : 5.67118
History : 0.35969
```

Figure 2: example of memento pattern

0.1.2 Eclipse Debugger

Introduction : Debugger used in java application which is designed on theoretical level to find simple bugs , insufficiences and weakness of a program.They bind to provide access to information that is still in the run-time stack.There are so many debuggers used depending upon the language and application like jdb , jBixBie etc .Below are some advantage and disadvantage of integrated debugging tool in ecllipse .

Advantages

- Reduce the problem and convert it to the automated test.
- Helps to inspect your codes , values and evaluating symbols.
- Single step through the code.
- Stop execution at a given point to investigate where it goes and what the values are.
- Attach to an already running program.

Disadvantages

- Not running real-time, so may not expose all problems
- Going through all breakpoints is done manually.

0.1.3 Quality Attributes

Program fulfills all the quality attributes and is efficient , also provides test cases which helps in traceability .Program is accurate for the values within its domain.

- **Correct**
  My Program is meant to be correct as the procedures are followed and values for the particular function are correct and within its range and gives accuracy and I have tested it with testing unit framework.

- **Efficient**
  My Program is efficient as it gives result within seconds and takes less response time. Also it is memory efficient.

- **Maintainablity**
  Program is maintainable as the interdependency is high cohesion and low coupling so if any errors occur program can be easily changeable and therfore it is used again also.

- **Robust**
  Program has ability to cope up with different types of errors due to unexpected values by providing facility of exceptions and validations.

- **Usable**
  My Program is understandable and provides textual user interface conventions easy to learn.It takes care of users point of view , usability testing methods are used to achieve results and its Error tolerant.I have also used Memento design pattern to store the state of an object as a history to see it later.

0.1.4 Checkstyle

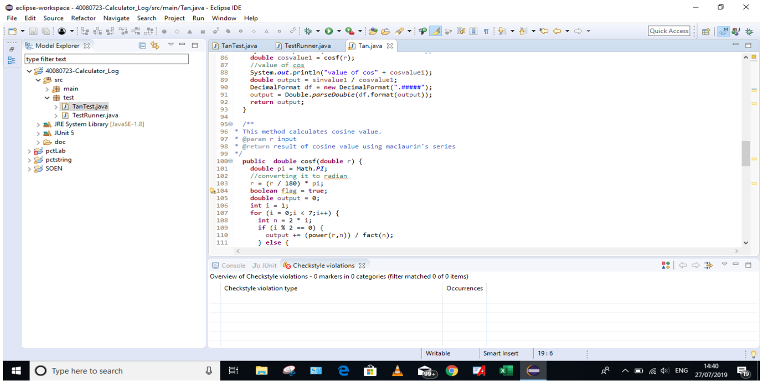
Introduction : The Eclipse Checkstyle Plugin integrates the static source code analyzer into the Eclipse IDE. Checkstyle is a Open Source development tool which adheres to a set of coding standards.Checkstyle helps to define and easily apply those common rules.

Advantages

- They allow for incredibly accurate estimates of this function.
- It is better external tooling. It's much easier to integrate checkstyle with our external tools since it is really designed as a standalone framework.
- Ability of creating our own rules. Eclipse defines a large set of styles, but checkstyle has more, and we can add our own custom rules.

Disadvantages

- Java code should be written with ASCII characters only, no UTF-8 support.
- To get valid violations, code have to be compilable, in other case you can get not easy to understand parse errors.
- You cannot see the content of other files. You have content of one file only during all Checks execution. All files are processed one by one.
- You cannot determine the full inheritance hierarchy of type.



```
    double cosvalue1 = cosf(r);
    //value of cos
    System.out.println("value of cos" + cosvalue1);
    double output = sinvalue1 / cosvalue1;
    DecimalFormat df = new DecimalFormat(".#####");
    output = Double.parseDouble(df.format(output));
    return output;
  }

  /**
* This method calculates cosine value.
* @param r input
* @return result of cosine value using maclaurin's series
*/
  public  double cosf(double r) {
    double pi = Math.PI;
    //converting it to radian
    r = (r / 180) * pi;
    boolean flag = true;
    double output = 0;
    int i = 1;
    for (i = 0;i < 7;i++) {
      int n = 2 * i;
      if (i % 2 == 0) {
        output += (power(r,n)) / fact(n);
      } else {
```

Overview of Checkstyle violations - 0 markers in 0 categories (filter matched 0 of 0 items)

Checkstyle violation type	Occurrences

Figure 3: Checkstyle with zero violations

0.2 Problem 6

0.2.1 Unit test

JUnit

Unit testing in java means testing the smaller units of your application, like classes and methods. The reason to test your code is to prove to yourself, and perhaps to the users / clients / customers, that your code works. Unit tests are typically automated, means once they are implemented, you can run them again and again. The goal of unit testing is to isolate important parts (i.e. units) of the program and show that the individual parts are free of certain faults. I have used standard unit testing framework known as JUnit by which we can easily and incrementally build a test suite that will help us measure our progress.Here,i have used Test Driven Development approach for testing.I have used different testing methods for particular functions in the program and also I have used validation testing method for the same to verify the values.

Bibliography

[1] https://genome.sph.umich.edu/wiki/Debuggers

[2] http://tutorials.jenkov.com/java-unit-testing/index.html

[3] https://stackoverflow.com/questions/13644624/advantage-of-using-checkstyle-rather-than-using-eclipse-built-in-code-formatter

[4] https://checkstyle.org/eclipse-cs/#!/

[5] https://www.wqusability.com/articles/more-than-ease-of-use.html