TACKLING WITH CODE REVIEW
HTTPS://GITHUB.COM/BASANTISCITS/SOEN_6011

PROBLEM 5 REVIEW FUNCTION : F1: ARCCOS(X) VERSION 1.0

Basant Gera (40082433)*
Dept. of Computer Science & Software Engineering
Concordia University
basantgera29@gmail.com

Submitted to : Pankaj Kamthan* & Team*
Dept. of Computer Science & Software Engineering
Concordia University
kamthan@cse.concordia.ca

August 2, 2019

1 Steps taken into account to perform code review are as follows :

- **Scope and size** : While Reviewing the code I Knew the scope, Domain & Range of arccos(x).
- Use of Intention-Revealing Names which help you to remember what this declaration is meant for.
- Avoidance of Duplication of function would not be there.
- Creating and Destroying Objects : Avoid creating unnecessary objects.
- Don't ignore exceptions.
- Adhere to generally accepted naming conventions.
- Make sure the code formatting is applied.
- Do all the method names seem clear and intuitive?
- Are there ways in which this code could be made more efficient?
- Use Check Style to follow some format in which code should be readable and understandable to others and peers.
- Variable and function name declared are meaning full.

2 10 Set of categories to distinguish code review which are as follows:

1. *Security* : The process of auditing the source code for an application to verify that the proper security controls are present.
2. *Error Prone* : Error Prone to augment the compiler's type analysis, you can catch more mistakes before they cost you time, or end up as bugs in code.
3. *Code Style* : The purpose of Code Style is to help developers to maintain consistency between design and implementation styles across many team members.
4. *Compatibility* : Compatibility Testing is a type of Software testing to check whether your software is capable of running on different hardware, operating systems, applications, network environments or Mobile devices.
5. *Unused Code* : Unused code should not be there in the code while reviewing.
6. *Performance* : Ensure Code performance Faster and efficient.
7. *Code Convention / Name Convention* : Ensuring proper Name convention followed in the code.
8. *Commenting* : Ensure proper Commenting should be there in the code.

*Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

9. *Spacing* : Ensure proper Spacing should be there in the code.
10. *Indentation* :Ensure proper Indentation should be there in the code.

3 Computing Environment Used is as follows :

- *IDE Used* : Eclipse.
- *IDE Version Used* : Neon 4.6
- *OS used* : Windows 10.
- *Reviewing Tool used* : Codacy
- *Test Cases Check Based on Review on properties like* : Security,Error Prone,Code Style,Compatibility,Unused Code,Performance,Naming Conventions,Indentation,Comments,spacing.

4 Review based on Code Smell by Code Review Tool on Test Cases as per requirement document : Codacy

- *File Name* : CalController.java
- *Function Name* : initial()
- *Return type* : void type
- *Function Description* : It calls view and get interface from the view Which has button and has a action listener event.
- *Security* : Secured enough.✓
- *Error Prone* : No errors found.✓
- *Code Style* : followed coding style in the function.✓
- *Compatibility* : followed good view Controller Architecture and is compatible enough.✓
- *Unused Code* : No unused code listed in the function.✓
- *Code Convention / Name Convention* : Code convention done correctly and followed eclipse-java-google-style.XML.✓
- *Spacing* :Not Aligned and spacing in the function not done correctly.✗
- *Indentation* : Not aligned properly in the function.✗
- java doc mentioned above the function: No java doc mentioned over the function nor on the file.✗

. . . . . .

- *File Name* : CalController.java
- *Function Name* : calculate()
- *Return type* : void type
- *Function Description* :The calculate function has parameter strnum and controller object.Usually it takes text and have listed exceptions like NumberFormatException and RuntimeException.
- *Security* :Secured enough.✓
- *Error Prone* : No errors found.✓
- *Code Style* : followed coding style in the function.✓
- *Compatibility* : Code convention done correctly and followed eclipse-java-google-style.XML.✓
- *Unused Code* : No unused code listed in the function.✓
- *Code Convention / Name Convention* : Followed Name Conventions.✓
- *Spacing* :Aligned and spacing in the function done correctly.✓
- *Indentation*:Aligned properly in the function.✓
- java doc mentioned above the function: No java doc mentioned over the function nor on the file.✗

- *File Name* : CalController.java
- *Function Name* : arrcos()
- *Return type* : double type
- *Function Description* : arrcos() calculates the inverse of cos with the value given to the function and return back the double result
- *Security* :Secured enough.✓
- *Error Prone* : No errors found.✓
- *Code Style* : followed coding style in the function.✓
- *Compatibility* : Code convention done correctly and followed eclipse-java-google-style.XML.✓
- *Unused Code* : No unused code listed in the function.✓
- *Code Convention / Name Convention* : Followed Name Conventions.✓
- *Spacing* : Aligned and spacing in the function done correctly.✓
- *Indentation* :Aligned properly in the function.✓
- java doc mentioned above the function: No java doc mentioned over the function nor on the file.✗

- *File Name* : CalView.java
- *Function Name* : CalView(),initial(),getText().getButton(),getAnswer(),getPanel()
- *Return type* : Constructor, void(),TextFeild type,Jbutton(),JLabel,JPanel();
- *Function Description* : Function which shows the interface has button textfeild panel and label.
- *Security* :Secured enough.✓
- *Error Prone* : No errors found.✓
- *Code Style* : followed coding style in the function.✓
- *Compatibility* : Code convention done correctly and followed eclipse-java-google-style.XML.✓
- *Unused Code* : No unused code listed in the function.✓
- *Code Convention / Name Convention* : Followed Name Conventions.✓
- *Spacing* : Aligned and spacing in the function done correctly.✓
- *Indentation* :Aligned properly in the function.✓
- java doc mentioned above the function: No java doc mentioned over the function nor on the file.✗

5 Review based on Code Smell by Code Review Tool on java file calController.java : Codacy

Function Name	Actual value	Expected Value	Pass or Fail	Comments	symbol ✓or ✗
arrcos(0.2)	1.3694811973578689	1.3694811973578689	True	Correct Value	✓
arrcos(-1)	3.14159265358979	3.14159265358979	True	Correct Value	✓
arrcos(1)	0.0	0.0	True	Correct Value	✓
arrcos(2)	Value out of Domain[-1,1]	Value out of Domain[-1,1]	True	Value out of Domain[-1,1]	✓
arrcos(-2)	Value out of Domain[-1,1]	Value out of Domain[-1,1]	True	Value out of Domain[-1,1]	✓

6 Comments for code review which is done :

- As per the review only java doc & commenting over class and the function are missing.
- Apart from that code is short and new features of java has been used and made code smaller and faster.
- No errors found and exception handling done exceptionally good.
- Followed Domain and range as per requirement document no 2 and found everything correct.
- Variable name and function name can be more meaning full and can be improved.

References

[1] https://www.rapidtables.com/calc/math/Arccos_Calculator.html

[2] https://www.analyzemath.com/Calculators_2/arccos_calculator.html

[3] https://www.rapidtables.com/math/trigonometry/arccos.html

[4] https://link.springer.com/article/10.1023/A:1008121502359