Arccos Function

Prabhpreet Singh

SOEN 6011: Software Engineering Principles

Introduction

- The Arccos function is the inverse of the cosine function. It takes input in the range of -1 to 1 and gives output in the range of 0 to π.

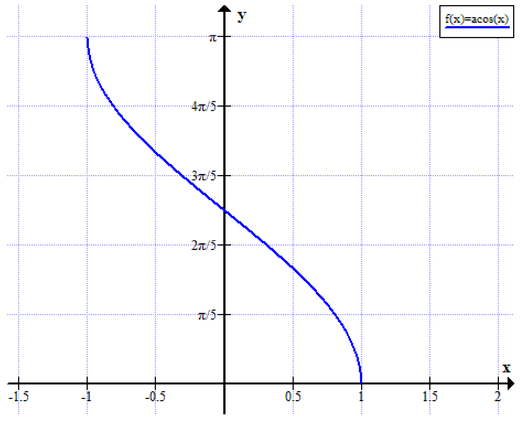

Fig. 1: Arccos Graph

- The value of arccos function is π at -1. The value decreases to $\pi/2$ at 0 and later on it becomes 0 at input value 1. It is one-to-one function.

Requirements

- The value to be entered by the user should be a real number between -1 to 1.
- The response time for the function should be less than 2 seconds.
- The function should be able to handle the exception when the entered input is outside the domain of the function.
- The user should have knowledge of the function and its domain and co-domain.

Implementation

- The Arccos(x) function is solved by using the Taylor Series Expansions of Inverse Cosine Function. The series is an infinite series with Geometric Progression.
- The equation for Arccos function is given below:
  $cos^{-1}x = 3.1415926535/2 - x + 1/2 * x^3/3 + 1/2 * 3/4 * x^5/5 + 1/2 * 3/4 * 5/6 * x^7/7 + ... \; where - 1 \leq x \leq 1$
- Recursive approach has been used for the implementation of the function.
- The summation $1/2 * 3/4 * x^5/5 + 1/2 * 3/4 * 5/6 * x^7/7 + ...$ is taken separately. We find that in the numerator it is odd factorial such as 1, $1 * 3$, $1 * 3 * 5$.
- The denominator is having Even Factorial such as $2, 2 * 4, 2 * 4 * 6$. The numerator is multiplied by X to power of odd number starting from X^3 , X^5, X^7.
- The denominator is multiplied by odd number that is same as that on power of X, such as 3, 5 ,7.So we perform operations separately.
- As odd number here starts from 3, we take a counter starting from 3 which is incremented by 2 after the operations. The summation continues in the for loop. The loop terminates the loop just before the summation becomes infinite.

Arccos Function GUI

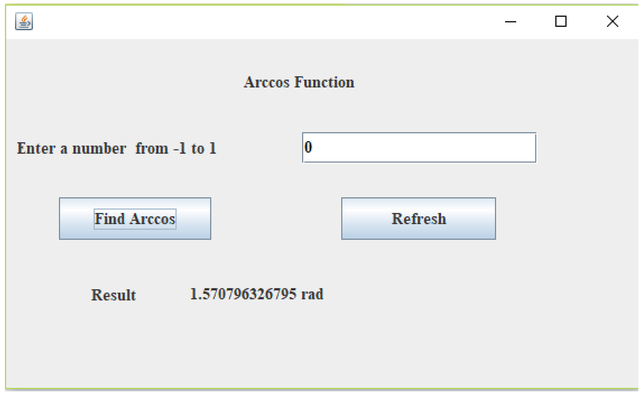

Fig. 2: Arccos GUI

Results

- The results of the Arccos function implementation are close to the actual results.
- For the mid range values, it gives correct values. For the input value 0, the output comes to be 1.570796 radian which is correct.
- As it goes to the extreme values, the values does infinite summations and hence we need to terminate the loop earlier than that point. So the values deviates a little from the correct value.
- The value of arccos(1) = 0.046078 which is close to the expected value that is 0. The absolute error comes out to be 0.046078.

Critical Decisions

Critical decisions are the decisions made during the software development that have an impact on operational, financial or product quality of a software. There were many critical decisions taken during the development of the project.

- Various algorithms were designed to implement the function. The algorithm to separate parts of the term and then performing summation recursively was implemented as it provided with solution nearest to the actual result. This decision is critical as the calculator should be able to provide accurate results for a given input. This was the motive for choosing recursive approach.
- The decision to terminate the loop was very critical because after that point, the summation turns to infinity and before that point, the absolute error increases. So I had to try different possible points in order to achieve that point. The user should not get infinity as the output which is not correct.
- The decision for the User Interface to settle on was very significant. The interaction with the user plays a main role in a software project. The Interface should direct users how to work with the software. On the other hand, the Interface should not be verbose. The interface should mention about the domain within which user has to enter the value and units which are used in the product.
- The project structure is very important for the software design. The product is designed such that it can be maintained or updated according to the new environment. The amount of code and the resources are controlled that enhances the efficiency of the software.The google programming style is used for improving the readability of the code. The various invalid inputs user can input are sought and they are handled by giving proper error messages that user can understand. These are critical decisions as they are important for the operation of the software and also for future scope of the project.

Lessons Learnt

- The function could be implemented with other logical algorithm that would give more accurate results even at the extreme values such as near -1 and 1. The point where the loop terminates just before the summation turns infinity can be optimized.
- The calculator could be designed to perform multiple operations such as user could perform addition, subtraction, multiplication and division on the input number and then perform Arccos function on the final number.
- Fields should be declared at the top of the class before any method declaration, initializers or inner class. For example, private JTextField inputField, int counter statements should be declared at the top of the class.
- Use explicit scoping instead of the default package private level. The variables or methods should be declared private to enforce encapsulation.
- The comments for the implementation are properly written. But they are missing in some important areas where they are required.
- There should be a separate error message when no input is entered by the user. But the same error message is given as for the input outside the domain of the function. It will assist the user more in understanding the error.
- The project taught me that summation of $1 * 10^{-9}$ and ahead can cause big change in the result. A these values were turning to infinity so these could not be taken which resulted in error in the output result.
- The project taught that at some places we have to act as team. For example, if even one team member does not follow the same programming style, then the team has to suffer.
- The project restricts us to use any in-built functions for the implementation. The project was built from scratch which gave us in-depth knowledge about developing the code. We had to construct in-built operations ourselves. It was great way that we understand the project and its requirements and develop it according to the requirements.

References

- Inverse Cosine,
  `http://mathworld.wolfram.com/InverseCosine.html`
- Taylor Series Expansions of Inverse Trigonometric Functions,
  `https://www.efunda.com/math/taylor_series/inverse_trig.cfm`
- Michael A. Cusumano and Alan MacCormack,Chris F. Kemerer, William Crandall, *Critical Decisions in Software Development*
- Critical decision-making skills for project managers
  `https://www.pmi.org/learning/library/critical-decision-making-skil`
- Latex Posters
  `https://www.overleaf.com/learn/latex/Posters`
- Latex Bibliography
  `https://www.overleaf.com/learn/latex/Bibliography_management_with_bibtex`