Arccosine Function

Himansi Maheshkumar Patel ID: 40072262

Gina Cody School of Engineering and Computer Science, Concordia University

Introduction

The arccosine is the inverse trigonometric function.The arccosine indicates the angle whose cosine is x. The arccosine of x is defined as the inverse cosine function of x when -1 ≤ x ≤ 1.
When the cosine of y is equal to x : $cosy = x$. Then the arccosine of x is equal to the inverse cosine function of x, which is equal to y : $\arccos x = \cos^{-1} x = y$.
The Taylor's series for arccosine function is :

- The domain of $\arccos x$ is -1 ≤ x ≤ 1.
- The range of $\arccos x$ is 0 ≤ y ≤ π in radians or 0°≤ y ≤180° in degrees.
- The arccosine function is a reflection of the cosine function about the line $y = x$.
- The arccosine function is continuous on open interval (-1,1)

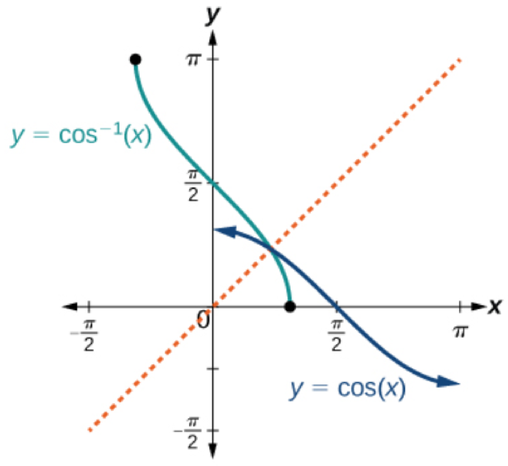

Fig. 1: arccosine Function

- It is also useful in application of engineering, physics and others.

Problem 1 - Function Description

- Finding precise information about arccosine function especially its unique characteristics was quite hard. I needed to search into various sources and find a source with the right information.
- Learned about arccosine function, its domain & range, and its applications about which I was not completely aware of!
- Learned a new language called 'LaTeX' for writing a document in very short time.

Problem 2 - Function Requirements

- Decided functional and non-functional requirements for arccosine function to see whether some of them would work or not in the future?
- After researching on ISO/IEC/IEEE 29148 Standards, I got to know about standard ways by which we can define functional and non-functional requirements in context of software development.

Problem 3 - Algorithm

- Brainstormed with my team to decide an identical pseudocode format which can be useful for team members when reviewing code.
- After exploring various possible approaches to solve arccosine function, using Taylor's series for its evaluation seemed most appropriate.

$$\arccos x = \frac{\pi}{2} - \sum_{n=0}^{\infty} \frac{(2n)!}{2^{2n}(n!)^2} \frac{x^{2n+1}}{(2n+1)}, |x| < 1$$

- In order to arrive at an optimal solution, from a performance perspective, I tried implementing and further comparing an iterative version of the algorithm against a recursive one and decided to use iterative approach.
- Learned about implementing an algorithm using a Taylor series.

Problem 4 - Implementation

- Brainstormed with my team to choose a common programming style which makes it easier to review codes for others.
- Generally, it seems more logical to go for a GUI when the system offers a variety of options to an End-User for interacting with the application. Considering our application requirements which primarily involved limited user interaction only in form of taking user inputs I found using a textual interface better than a complex GUI.
- Learned about implementing the arccosine function from a scratch since I didn't have access to any in-built library.
- While implementing arccosine function, I learned about error handling mechanism and how useful an error message can be for the users.
- Learned about ways to implement a program which makes it correct, efficient, usable and robust.
- I had never used checkstyle before. I came to know the use of checkstyle in project.
- Learned about agile principles.

Problem 5 - Code Review

- When I implemented a program, I didn't know that I could use static method for the utility function. But, after getting review from teammate, I came to know about static method. I then searched on internet that in which context I can make method static. Then I realized that I could have used static method for my function as well.
- Got to know the importance of working in a team that how reviewing team member's code can be useful in sharing knowledge, finding early bugs, team cohesion and maintaining consistent code style in a group.
- Learned about manual and automatic code review approach.
- Used **PMD** tool to check the quality of source code which was a new lesson for me.

Problem 6 - Testing

- Decided to make testcase methods based on arccosine function's requirements as it finds early bugs, reduces the cost of bug fixes, and stops program from failing. For implementing this, I used standard unit testing framework known as 'Junit".
- Arccosine function was not able to provide accurate result up to 4 decimal points for some domain values so I had to make a decision to check arccosine function's results up to 2 decimal points only.
- Learned how to write testcase methods which can cover all possible the testcases.

Problem 7 - Test Case Analysis

- From testcase analysis approach, I came to know about different steps of functional testing, how to perform it, and how to report it.
- Learned about importance of test case analysis report to the developer as it helps them in finding out failing testcases.

GitHub Link

https://github.com/Himansipatel/SOEN-6011-Team-H-Himansi