Richard Hayes Crowley

06/15/2021

CSC\_242\_Lab\_03

**SOURCE CODE:**

*# Babylonian Method*

*# Programmer: Richard Hayes Crowley*

*from* tabulate *import* tabulate

iterationArr = []

def Babylonian(*number*):

global iterationArr

*# abs value for negative numbers*

x = abs(*number*)

y = 1

count = 0

*if* x > 2:

y = x/2

*while* (True):

y = (y + x / y) / 2

count += 1

iterationArr.append([count, y])

*# rounding to within 4 points of precision*

*if*(round(y\*y, 4) == x):

*return* y

print("----- The Babylonian Method -----\n")

radicand = float(input("please enter a real number "))

radical = Babylonian(radicand)

*# conditional statement for imaginary numbers*

*if* radicand < 0:

radical = f"{radical:0.4f}i"

*else*:

radical = f"{radical:0.4f}"

print(tabulate(iterationArr, *headers*=[

"Iteration number", "Approximation"], *tablefmt*='github', *numalign*="left"))

print("")

print(f"The square root of {radicand} is {radical}")

**OUTPUT**

Text

Description automatically generated