Marcus Kielman

Babylonian Method

Description: The Babylonian Method states in a square root function, the relationship between the square root, and the radicand is such that the square root of any number lies in a range [y, v/y] (where y is any real number, and v is the radicand. If y is constantly replaced with the average of these two values, y and v/y, then the range will ultimately converge to the square root of the radicand. The following program calculates the square root of any given value using the Babylonian Method.

Code:

Note: The following code was written in Python

import random

def sqrt(value):

value = prevans = float(value)

try:

sqrtans = float(random.randint(0,10000000000000000000000000000000000000000000000000000)%value-1+value)

except:

sqrtans = 0

print('Random number is: ', sqrtans)

print('You are finding the square root of: ', value)

print('According to the Babylonian Method, each number calculated is your new guess at finding the correct answer:')

while sqrtans != prevans:

prevans = sqrtans

sqrtans = (sqrtans + (value / sqrtans)) / 2.0

print (sqrtans)

if sqrtans < 0:

return 'Cannot Compute Imaginary Numbers'

print('The square root of your answer is: ')

return sqrtans

if \_\_name\_\_ == '\_\_main\_\_':

print(sqrt(144))