Assignment 2 - Slice of Pi

Chucheng Xie

CSE 13S – Spring 2023

Purpose

The purpose of this assignment is to approximate many different values, such as e and pi, using various methods. Different approximation methods include Madhava Series, Euler's Solution, and Viete's Formula. We keep adding items until the current item is less than the value of epsilon (1 * 10^-14). Other approximation methods include Newton's Method, Bailey-Borwein-Plouffe Series, and Wallis Series. There will be a test file that will run all approximate tests.

Program Design

Pseudocode:

- **e.c**

```
counter = 0
while current term >= epsilon:
        divide current term by temp
        add current term to sum
        add counter by 1
        multiply temp by counter
return the sum
```

- **madhava.c**

```
counter = 0
sum = 0
while current term >= epsilon:
        divide top term by -3
        add bottom term by 2
        add sum by top term
        add current term to sum
```

add counter by 1

return the sum

- **euler.c**

counter = 0

sum = 0

while current term >= epsilon:

add current term /= counter

add counter term to sum

multiply bottom_term by 2

add sum by current term

add counter by 1

return the sum

- **bbp.c**

k = 0

counter = 0

sum = 0

while current term >= epsilon:

current term = (4 / (8k + 1) - 2 / (8k + 4) - 1 / (8k + 5) - 1 / (8k + 6)) * 16^-k

add current term to sum

add k by 1

add counter by 1

add sum by current term

return the sum *square root of 6

- **viete.c**

counter = 0

product = 0

previous_term = 1

current_term = square root of 2

while current term >= epsilon:

multiply previous term by current_term

divide bottom term by 2

add couter by 1

pervious_term = current_term

current_term = square root of (2 + previous_term)

return 2 / sum

- **wallis.c**

iters = 0

terms = 1.0

x = 1.0

for i = 1; x >= epsilon; i++

terms = (4 * i * i) / (4 * i * i - 1.0)

iters += 1

x = ((4 * i * i) / (4 * i * i - 1.0)) - 1.0

return 2 * terms

- **mathlib-test.c**

parse through input

while there are inputs to be read:

mark which command arguments are inputted

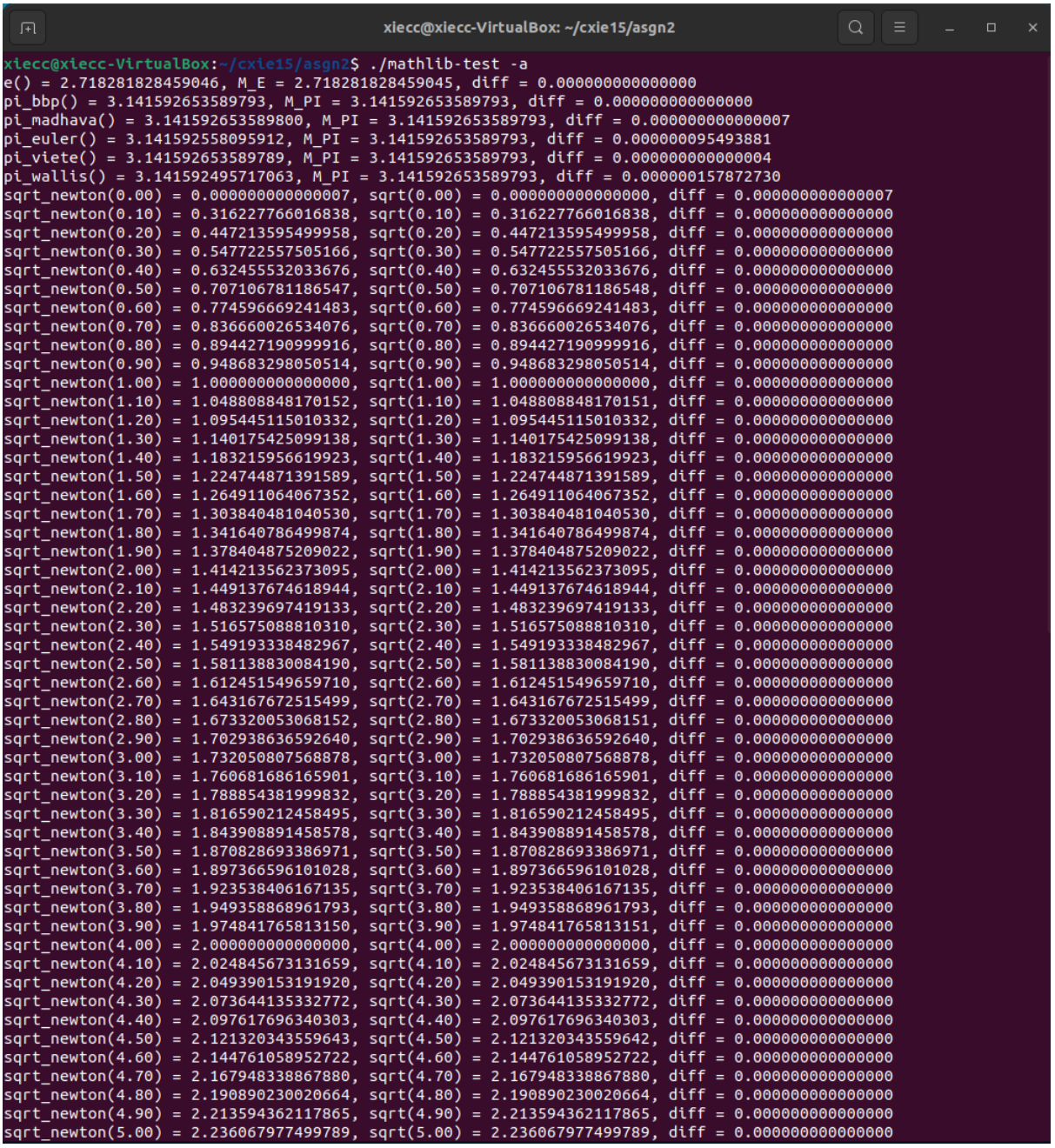
run tests that are flagged

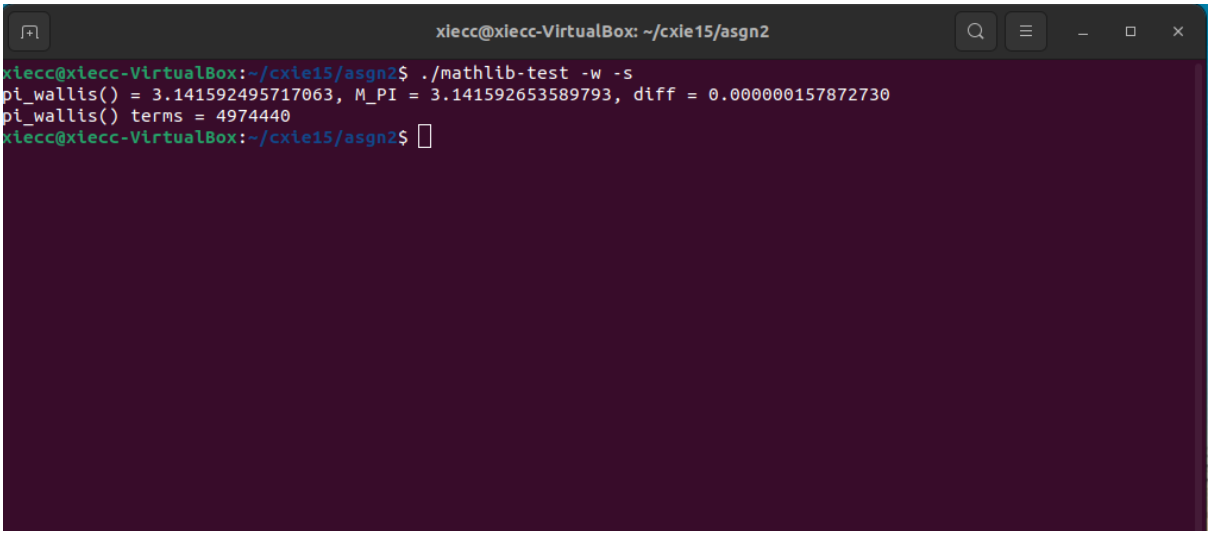
provide statistics if -s is flagged

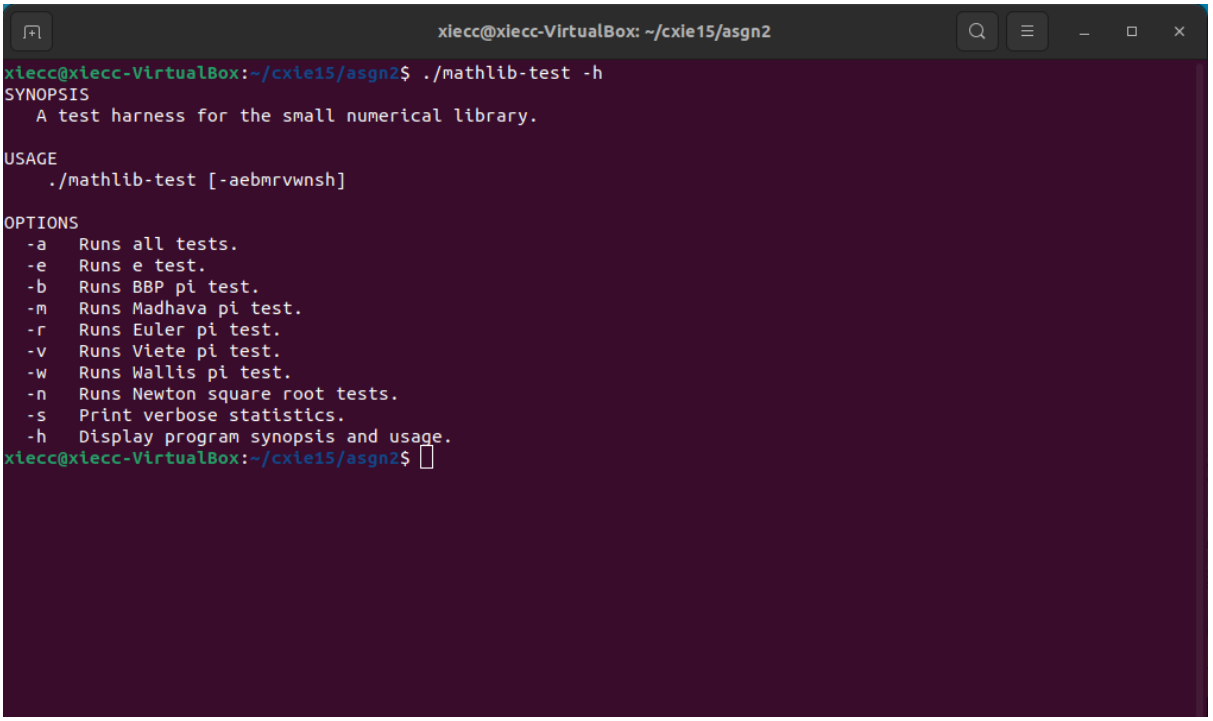
provide instructions if -h is flagged

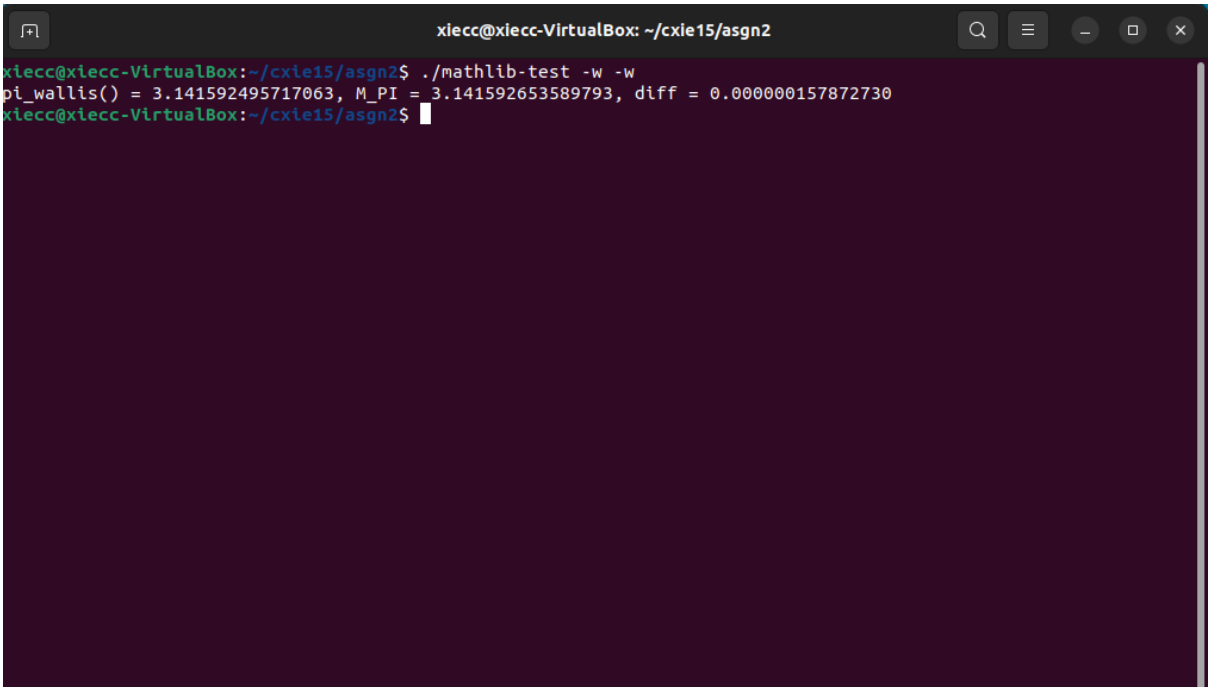
Result

Below are screenshots of program output.


xiecc@xiecc-VirtualBox: ~/cxie15/asgn2
xiecc@xiecc-VirtualBox:~/cxie15/asgn2$ ./mathlib-test -a
e() = 2.718281828459046, M_E = 2.718281828459045, diff = 0.000000000000000
pi_bbp() = 3.141592653589793, M_PI = 3.141592653589793, diff = 0.000000000000000
pi_madhava() = 3.141592653589800, M_PI = 3.141592653589793, diff = 0.000000000000007
pi_euler() = 3.141592558095912, M_PI = 3.141592653589793, diff = 0.000000095493881
pi_viete() = 3.141592653589789, M_PI = 3.141592653589793, diff = 0.000000000000004
pi_wallis() = 3.141592495717063, M_PI = 3.141592653589793, diff = 0.000000157872730
sqrt_newton(0.00) = 0.000000000000007, sqrt(0.00) = 0.000000000000000, diff = 0.000000000000007
sqrt_newton(0.10) = 0.316227766016838, sqrt(0.10) = 0.316227766016838, diff = 0.000000000000000
sqrt_newton(0.20) = 0.447213595499958, sqrt(0.20) = 0.447213595499958, diff = 0.000000000000000
sqrt_newton(0.30) = 0.547722557505166, sqrt(0.30) = 0.547722557505166, diff = 0.000000000000000
sqrt_newton(0.40) = 0.632455532033676, sqrt(0.40) = 0.632455532033676, diff = 0.000000000000000
sqrt_newton(0.50) = 0.707106781186547, sqrt(0.50) = 0.707106781186548, diff = 0.000000000000000
sqrt_newton(0.60) = 0.774596669241483, sqrt(0.60) = 0.774596669241483, diff = 0.000000000000000
sqrt_newton(0.70) = 0.836660026534076, sqrt(0.70) = 0.836660026534076, diff = 0.000000000000000
sqrt_newton(0.80) = 0.894427190999916, sqrt(0.80) = 0.894427190999916, diff = 0.000000000000000
sqrt_newton(0.90) = 0.948683298050514, sqrt(0.90) = 0.948683298050514, diff = 0.000000000000000
sqrt_newton(1.00) = 1.000000000000000, sqrt(1.00) = 1.000000000000000, diff = 0.000000000000000
sqrt_newton(1.10) = 1.048808848170152, sqrt(1.10) = 1.048808848170151, diff = 0.000000000000000
sqrt_newton(1.20) = 1.095445115010332, sqrt(1.20) = 1.095445115010332, diff = 0.000000000000000
sqrt_newton(1.30) = 1.140175425099138, sqrt(1.30) = 1.140175425099138, diff = 0.000000000000000
sqrt_newton(1.40) = 1.183215956619923, sqrt(1.40) = 1.183215956619923, diff = 0.000000000000000
sqrt_newton(1.50) = 1.224744871391589, sqrt(1.50) = 1.224744871391589, diff = 0.000000000000000
sqrt_newton(1.60) = 1.264911064067352, sqrt(1.60) = 1.264911064067352, diff = 0.000000000000000
sqrt_newton(1.70) = 1.303840481040530, sqrt(1.70) = 1.303840481040530, diff = 0.000000000000000
sqrt_newton(1.80) = 1.341640786499874, sqrt(1.80) = 1.341640786499874, diff = 0.000000000000000
sqrt_newton(1.90) = 1.378404875209022, sqrt(1.90) = 1.378404875209022, diff = 0.000000000000000
sqrt_newton(2.00) = 1.414213562373095, sqrt(2.00) = 1.414213562373095, diff = 0.000000000000000
sqrt_newton(2.10) = 1.449137674618944, sqrt(2.10) = 1.449137674618944, diff = 0.000000000000000
sqrt_newton(2.20) = 1.483239697419133, sqrt(2.20) = 1.483239697419133, diff = 0.000000000000000
sqrt_newton(2.30) = 1.516575088810310, sqrt(2.30) = 1.516575088810310, diff = 0.000000000000000
sqrt_newton(2.40) = 1.549193338482967, sqrt(2.40) = 1.549193338482967, diff = 0.000000000000000
sqrt_newton(2.50) = 1.581138830084190, sqrt(2.50) = 1.581138830084190, diff = 0.000000000000000
sqrt_newton(2.60) = 1.612451549659710, sqrt(2.60) = 1.612451549659710, diff = 0.000000000000000
sqrt_newton(2.70) = 1.643167672515499, sqrt(2.70) = 1.643167672515499, diff = 0.000000000000000
sqrt_newton(2.80) = 1.673320053068152, sqrt(2.80) = 1.673320053068151, diff = 0.000000000000000
sqrt_newton(2.90) = 1.702938636592640, sqrt(2.90) = 1.702938636592640, diff = 0.000000000000000
sqrt_newton(3.00) = 1.732050807568878, sqrt(3.00) = 1.732050807568878, diff = 0.000000000000000
sqrt_newton(3.10) = 1.760681686165901, sqrt(3.10) = 1.760681686165901, diff = 0.000000000000000
sqrt_newton(3.20) = 1.788854381999832, sqrt(3.20) = 1.788854381999832, diff = 0.000000000000000
sqrt_newton(3.30) = 1.816590212458495, sqrt(3.30) = 1.816590212458495, diff = 0.000000000000000
sqrt_newton(3.40) = 1.843908891458578, sqrt(3.40) = 1.843908891458578, diff = 0.000000000000000
sqrt_newton(3.50) = 1.870828693386971, sqrt(3.50) = 1.870828693386971, diff = 0.000000000000000
sqrt_newton(3.60) = 1.897366596101028, sqrt(3.60) = 1.897366596101028, diff = 0.000000000000000
sqrt_newton(3.70) = 1.923538406167135, sqrt(3.70) = 1.923538406167135, diff = 0.000000000000000
sqrt_newton(3.80) = 1.949358868961793, sqrt(3.80) = 1.949358868961793, diff = 0.000000000000000
sqrt_newton(3.90) = 1.974841765813150, sqrt(3.90) = 1.974841765813151, diff = 0.000000000000000
sqrt_newton(4.00) = 2.000000000000000, sqrt(4.00) = 2.000000000000000, diff = 0.000000000000000
sqrt_newton(4.10) = 2.024845673131659, sqrt(4.10) = 2.024845673131659, diff = 0.000000000000000
sqrt_newton(4.20) = 2.049390153191920, sqrt(4.20) = 2.049390153191920, diff = 0.000000000000000
sqrt_newton(4.30) = 2.073644135332772, sqrt(4.30) = 2.073644135332772, diff = 0.000000000000000
sqrt_newton(4.40) = 2.097617696340303, sqrt(4.40) = 2.097617696340303, diff = 0.000000000000000
sqrt_newton(4.50) = 2.121320343559643, sqrt(4.50) = 2.121320343559642, diff = 0.000000000000000
sqrt_newton(4.60) = 2.144761058952722, sqrt(4.60) = 2.144761058952722, diff = 0.000000000000000
sqrt_newton(4.70) = 2.167948338867880, sqrt(4.70) = 2.167948338867880, diff = 0.000000000000000
sqrt_newton(4.80) = 2.190890230020664, sqrt(4.80) = 2.190890230020664, diff = 0.000000000000000
sqrt_newton(4.90) = 2.213594362117865, sqrt(4.90) = 2.213594362117865, diff = 0.000000000000000
sqrt_newton(5.00) = 2.236067977499789, sqrt(5.00) = 2.236067977499789, diff = 0.000000000000000


xiecc@xiecc-VirtualBox: ~/cxie15/asgn2
xiecc@xiecc-VirtualBox:~/cxie15/asgn2$ ./mathlib-test -w -s
pi_wallis() = 3.141592495717063, M_PI = 3.141592653589793, diff = 0.000000157872730
pi_wallis() terms = 4974440
xiecc@xiecc-VirtualBox:~/cxie15/asgn2$


xiecc@xiecc-VirtualBox: ~/cxie15/asgn2
xiecc@xiecc-VirtualBox:~/cxie15/asgn2$ ./mathlib-test -h
SYNOPSIS
A test harness for the small numerical library.
USAGE
./mathlib-test [-aebmrvwnsh]
OPTIONS
-a Runs all tests.
-e Runs e test.
-b Runs BBP pi test.
-m Runs Madhava pi test.
-r Runs Euler pi test.
-v Runs Viete pi test.
-w Runs Wallis pi test.
-n Runs Newton square root tests.
-s Print verbose statistics.
-h Display program synopsis and usage.
xiecc@xiecc-VirtualBox:~/cxie15/asgn2$


xiecc@xiecc-VirtualBox: ~/cxie15/asgn2
xiecc@xiecc-VirtualBox:~/cxie15/asgn2$ ./mathlib-test -w -w
pi_wallis() = 3.141592495717063, M_PI = 3.141592653589793, diff = 0.000000157872730
xiecc@xiecc-VirtualBox:~/cxie15/asgn2$