CSE 13s
DESIGN for Assignment 2
October 8, 2021

0.1 Taylor Series

$$\frac{x}{k}$$

To calculate the approximation of e:

- halter = epsilon
- two functions: function e (taylor series) and (taylor series) e terms (return the number of computed terms)
  1. taylor series function is to approximate the square root
  2. taylor series terms is to tell us how many times it ran
- create variables: k (k will start @ 1), current term, previous term
- create while loop: while current term > epsilon current term = 1/k e += previous term x current term
  previous term = current term (set up for the next loop)
- return the approximation of e

0.2 Bailey-Borwein-Plouffe

$$\sum_{k=0}^{n} 16^{-k}\left(\frac{4}{(8k+1)} - \frac{2}{(8k+4)} - \frac{1}{(8k+5)} - \frac{1}{(8k+6)}\right) \tag{1}$$

- two functions: function pi bbp and pi bbp terms (return the number of computed terms)
  1. bpp's function is to approximate the square root
  2. bbp terms is to tell us how many times it ran
- create variables: k (k will start @ 0) and term
- create while loop: w/ epsilon pi += (4 / 8 * k + 1) pi -= (2 / 8 * k + 4)
  pi -= (1 / 8 * k + 5)
  pi -= (1/ 8 * k + 6) * term
  term /= 16
- return the approximation of pi

0.3 Newton's Method of Square Root

To calculate the a square root:

- root formula = 0.5 * (x + (y/x)) x can be assumed to be y or x
- two functions: newton's function and newton iters
  1. square root's function is to approximate the square root
  2. square root terms is to tell us how many times it ran
- create variables: x (set to as an input), sqrt, temp
- create while loop: loop only if abs(sqrt - temp) > epsilon temp = sqrt
  sqrt = 0.5 * (temp + x / temp)
- return approximation of the square root

0.4 Euler's Solution

Finding the approximation of pi:

$$p(n) = \sqrt{6 \sum_{k=0}^{\infty} \frac{1}{k^2}} \tag{2}$$

- two functions: euler's function and euler terms
  1. euler's function is to approximate pi
  2. euler terms is to tell us how many times it ran
- create variables: n, k, pi
- create while loop: loop only if n > epsilon n = (1/k * 1/k) k += 1
  pi += n
  use newton's square root to determine 6 * pi
- return approximation of pi

DESIGN for Assignment 2

0.5 Madhava

1. Finding the approximation of pi:

$$\sum_{k=0}^{\infty} \frac{(-3)^{-k}}{2k+1} \tag{3}$$

- two functions: madhava's function and madhava terms
  (a) madhava's function is to approximate pi
  (b) madhava terms is to tell us how many times it ran
- create variables: x, pi (x = 1), k
- create for loop (where k reaches close to 0)
- it will alternate between adding and subtracting because there is a (-1/3k)
- return the approximation of pi
-

0.6 Viete

1. Finding the approximation of pi:

$$\prod_{k=1}^{\infty} \frac{a}{2} \tag{4}$$

- two functions: viete function and viete terms
  (a) viete function is to approximate pi
  (b) viete terms is to tell us how many times it ran
- create variables: k, pi
- create for loop (where k reaches close to 0)

0.7 MAIN Additional Diagrams

:

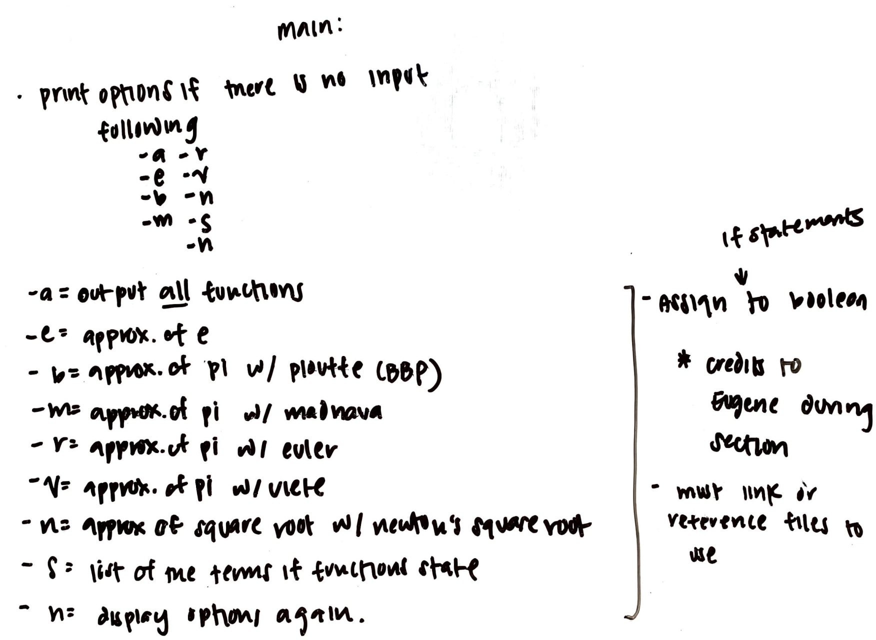

Figure 1: Diagram for the main

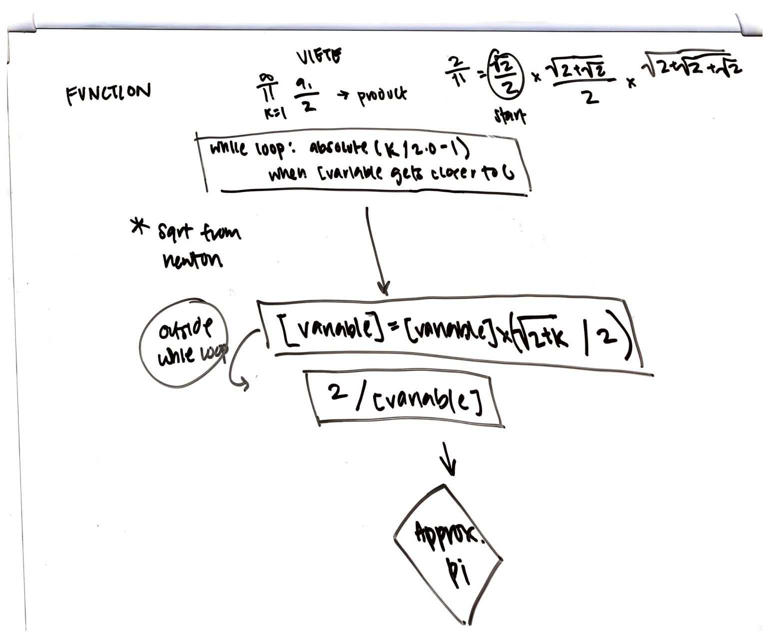

Figure 2: Diagram for approximating pi with viete

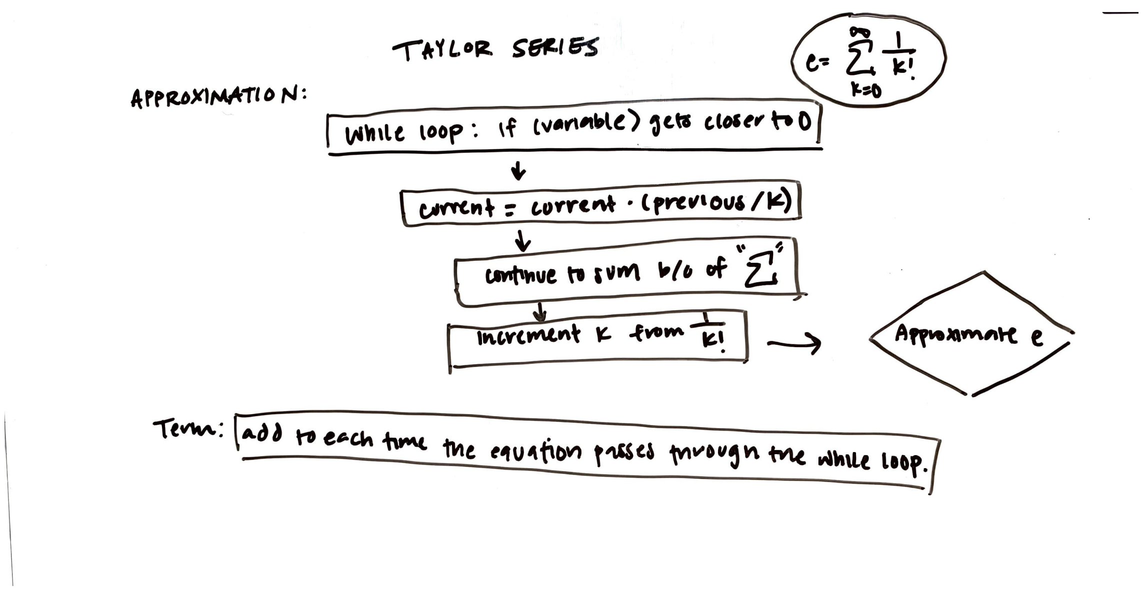

Figure 3: Diagram for approximating e with taylor series

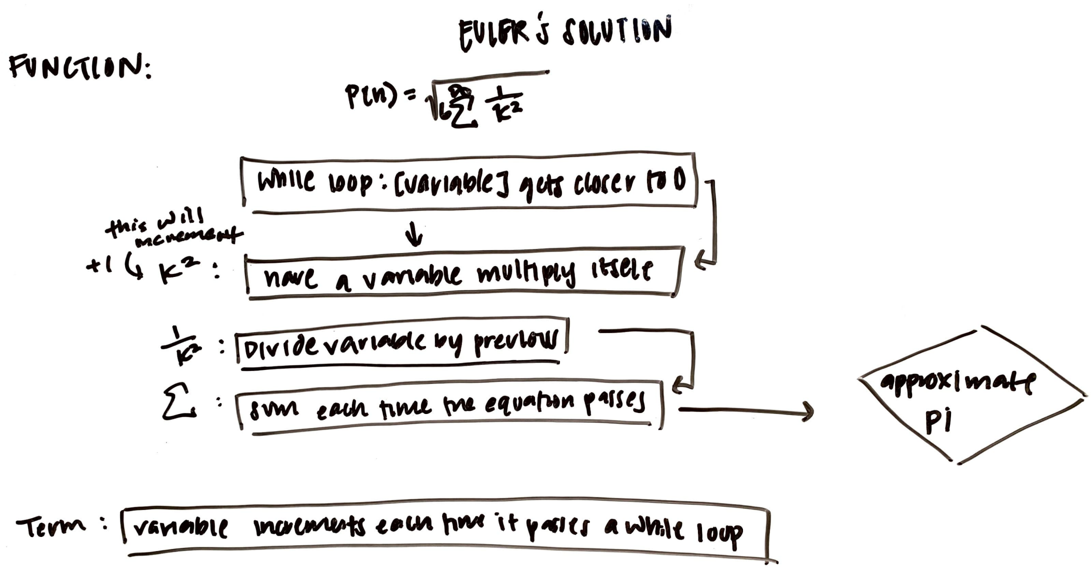

Figure 4: Diagram for approximating pi with euler

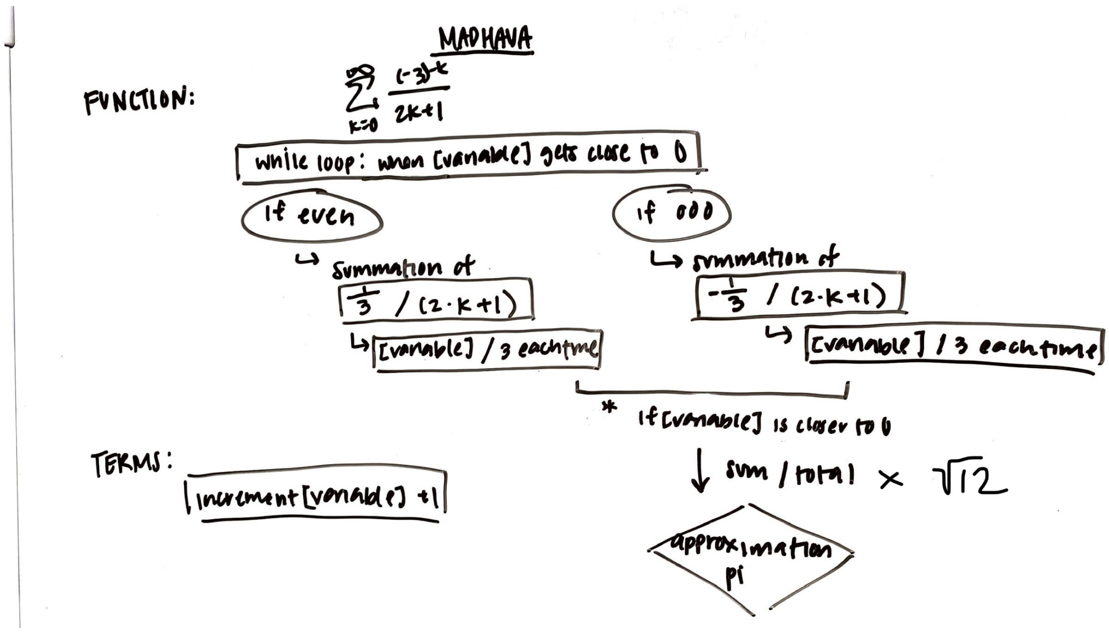

Figure 5: Diagram for approximating pi with madhava series

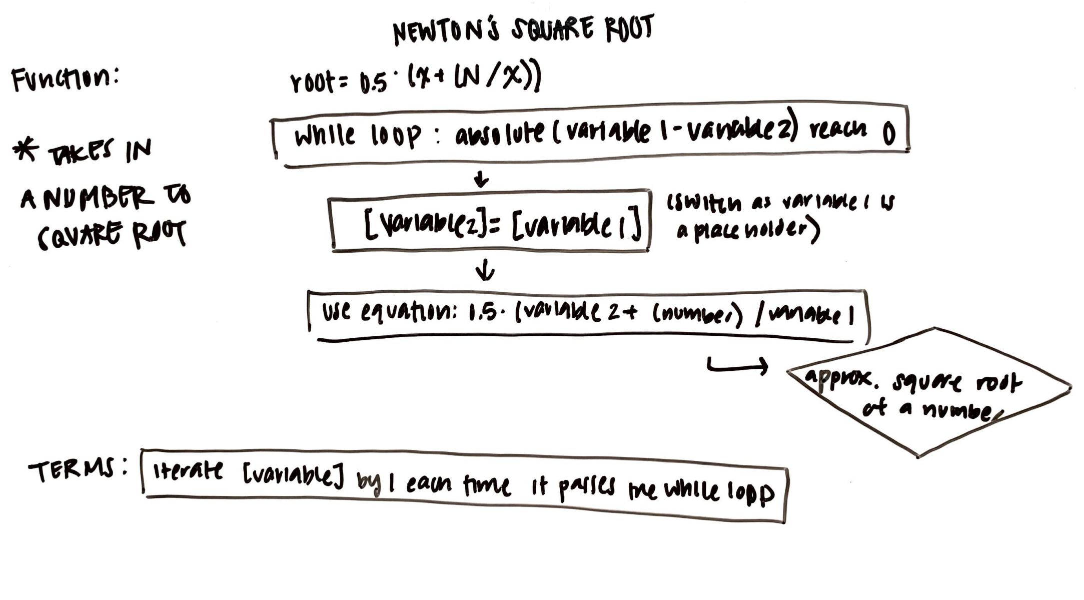

Figure 6: Diagram for approximating square root with newton's square root

DESIGN for Assignment 2

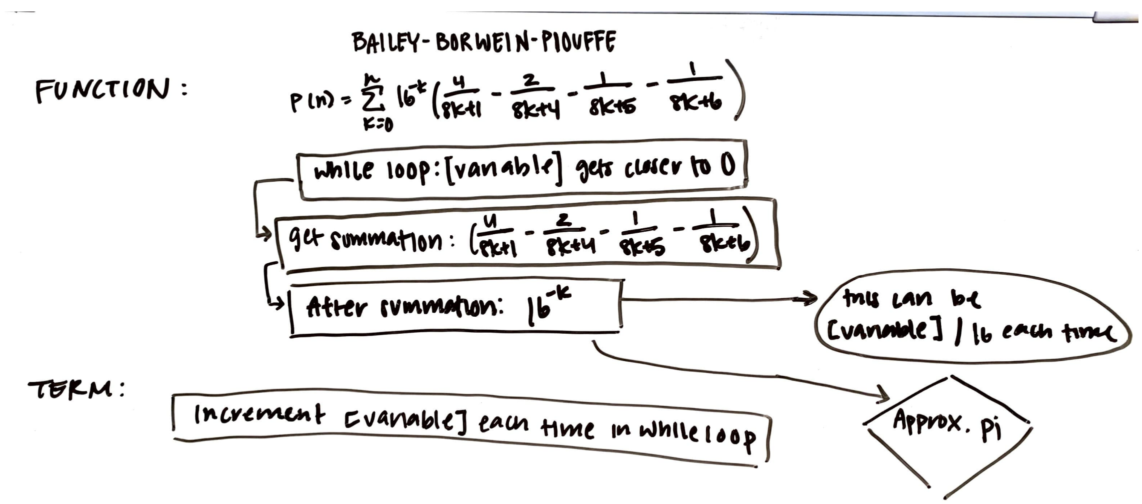

Figure 7: Diagram for approximating pi with bbp