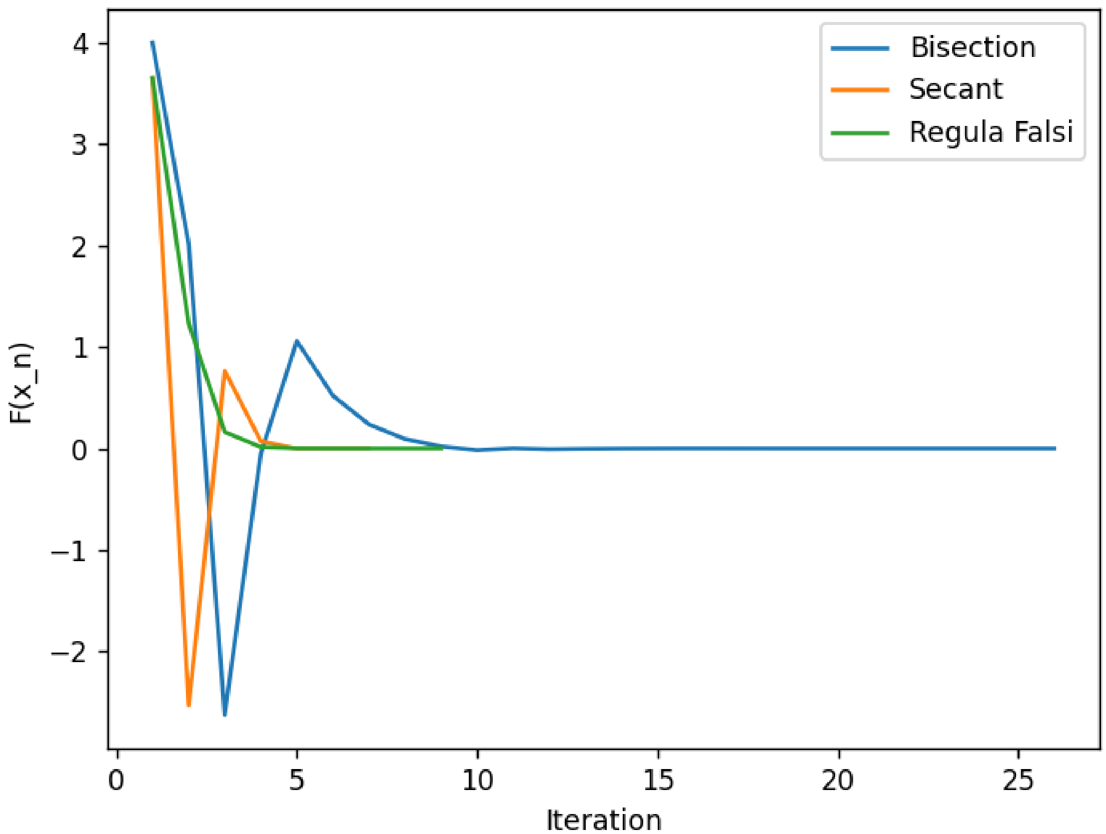
Bisection found root at -1.244258702 in 26 iterations with 2.3347995371463537e-09 error

Secant found root at -1.244258699 in 7 iterations with 1.535136128407901e-10 error

Regula Falsi found root at -1.244258694 in 9 iterations with 5.1814363028006205e-09 error



The value of the function with machine precision is 8.605235200000001

The value with rounding at each arithmetic operation is 8.606

The absolute error is 0.0007647999999988997 and relative error is 8.887612973075967e-05

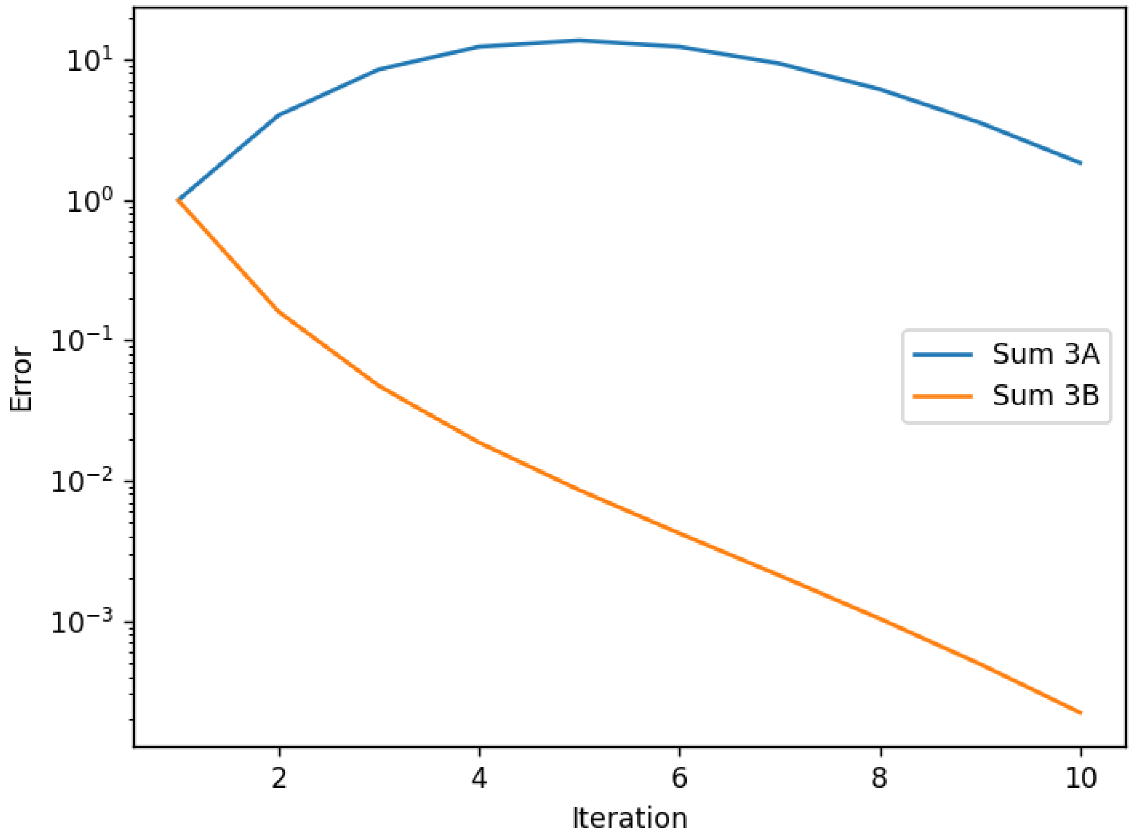
The value of nesting with machine precision is 8.605235200000001

The value of nesting with rounding at each arithmetic operation is 8.605

The absolute error is 0.00023520000000054608 and relative error is 2.7332198892198326e-05

The value of sum 3A after 10 iterations is -1.82711

The value of sum 3A after 10 iterations is 0.00696



A reason for the second sum converging to the true value so much quicker is that it

always has the same sign, meaning that every additional term gets it closer to the real value.

The other sum flips across the x axis, meaning it doesn't always get closer and takes longer to converge.

Q4 $x^4 + \sin x$ degree 3

a) $x_0 = 4$ $x_1 = x_0 + 0.2$

$$x^4 + \sin x + \frac{(4x^3 + \cos x)(0.2)}{!1} + \frac{(12x^2 - \sin x)(0.2)^2}{!2} + \frac{(24x - \cos x)(0.2)^3}{!3}$$

$$256 + \sin 4 + (256 + \cos 4)(0.2) + \frac{(192 - \sin x)\ (0.2)^2}{2} + \frac{(96 - \cos 4)\ (0.2)^3}{6}$$

b) $x_0 = 3$ $x_1 = x_0 - 0.7$

$$x^4 + \sin x + \frac{(4x^3 + \cos x)(-0.7)}{!1} + \frac{(12x^2 - \sin x)(-0.7)^2}{!2} + \frac{(24x - \cos x)(-0.7)^3}{!3}$$

$$81 + \sin 3 + (108 + \cos 3)(-0.7) + \frac{(108 - \sin 3)(-0.7)^2}{2} + \frac{(72 - \cos 3)(-0.7)^3}{6}$$

Q5

a) An unsigned int can store a larger number b/c it doesn't give up a bit to the sign

b) A double uses more memory than a float. 64 bits as opposed to 32 bits.
More memory means doubles are more accurate.

c) B/c integers are defined with known increments of 1, it is possible to use the == operator to compare their values

a = 1
b = 1
return a == b → should return a True

d) Set an appropriate threshold epsilon below which you can declare equality. Then compare the absolute value of their difference to that epsilon. If it is lower, then they can be declared equal and not equal otherwise.

a = x } float/doubles
b = y

eps = 1E-8

return abs(a-b) < eps → returns true when difference is small enough

e) Machine error for doubles is $2.2204 \cdot 10^{-6}$

Q6

a) 1482/2 = 741 (0) 741/2 = 370 (1) 370/2 = 185 (0) 185/2 = 92 (1) 92/2 = 46 (0)

46/2 = 23 (0) 23/2 = 11 (1) 11/2 = 5 (1) 5/2 = 2 (1) 2/2 = 1 (0) 1/2 = 0 (1)

$\boxed{(10111001010)_2 = (1482)_{10}}$

b) 0.1·2 = 0.2 0.2·2 = 0.4 0.4·2 = 0.8 0.8·2 = 1.6 0.6·2 = 1.2 0.2·2 = 0.4 0.4·2 = 0.8 0.8·2 = 1.6

$\boxed{(0.00011001)_2 = (0.1)_{10}}$

C) Yes, it makes the value larger

base 10 32 → 320 ← ten times larger

base 2 10000 → 100000 ← two times larger

$(16)_{10}$ $(32)_{10}$

D) if a zero is added to the right of a decimal number, nothing changes

$(0.5 = 0.50)_{10}$ $(0.1 = 0.10)_2$

True True