CH2120

Class 18

# mainProgram.f08

**program** mainProgram

**implicit** **none**

**call** mainSecant()

**end** **program** mainProgram

# mainSecant.f08

**subroutine** mainSecant

**implicit** **none**

! External function

**real**, **external** :: rootFindingFunction

! Variables: User Input

**real** xGuess1, xGuess2

**real** tolerance

! Variables: Console Output

**real** root, error

**integer** iterations

**character**(*len*=25) methodName

! Variables: Internal

**real** fGuess1, fGuess2

! Get user input for the (1) initial guess and (2) tolerance.

**write**(\*,\*) "Enter the first guess for the root:"

**read**(\*,\*) xGuess1

**write**(\*,\*) "Enter the second guess for the root:"

**read**(\*,\*) xGuess2

**write**(\*,\*) "Enter the tolerance:"

**read**(\*,\*) tolerance

! Evaluate function values at the initial guesses.

fGuess1 = rootFindingFunction(xGuess1)

fGuess2 = rootFindingFunction(xGuess2)

! Display the column headers of the output table.

**write**(\*,10) "|", "Method", "|", "Root", "|", "Error", "|", "Iterations", "|"

! Check: Is the first initial guess a root?

**if**(fGuess1 == 0) **then**

root = xGuess1

error = 0.0

iterations = 0

methodName = "None"

**call** displayRoot(methodName, root, error, iterations)

**return**

**end** **if**

! Check: Is the second initial guess a root?

**if**(fGuess2 == 0) **then**

root = xGuess2

error = 0.0

iterations = 0

methodName = "None"

**call** displayRoot(methodName, root, error, iterations)

**return**

**end** **if**

! Check: Is the slope of the initial secant equal to zero?

**if**(fGuess1 == fGuess2) **then**

**stop** "Error: The Secant method will not converge. Slope of the secant is zero."

**end** **if**

! Get and display the root as estimated by the secant method.

methodName = "Secant"

**call** secantRoot(xGuess1, xGuess2, tolerance, root, error, iterations)

**call** displayRoot(methodName, root, error, iterations)

10 **format**(a3, a25, a3, a10, a3, a10, a3, a12, a3)

**end** **subroutine** mainSecant

# secantRoot.f08

**subroutine** secantRoot(x1Input, x2Input, tolerance, root, error, iterations)

**implicit** **none**

! External function

**real**, **external** :: rootFindingFunction

! Variables: Input Arguments

**real**, **intent**(in) :: x1Input, x2Input, tolerance

! Variables: Output Arguments

**real**, **intent**(out) :: error, root

**integer**, **intent**(out) :: iterations

! Variables: Internal: x's, f's, and slope

**real** x1, x2, xIntercept

**real** f1, f2, fIntercept

**real** slope

! Initialize x's.

x1 = x1Input

x2 = x2Input

! Initialize f's.

f1 = rootFindingFunction(x1)

f2 = rootFindingFunction(x2)

iterations = 0

**do** **while**((error > tolerance) .**or**. (iterations <= 2))

iterations = iterations + 1

slope = (f1 - f2) / (x1 - x2)

**if**(slope == 0) **then**

**stop** "Error: The Secant method will not converge. Slope is zero."

**end** **if**

xIntercept = x2 - (f2 / slope)

fIntercept = rootFindingFunction(xIntercept)

error = *abs*(x2 - xIntercept)

! Discard point 1. New Point 1 = Current Point 2

x1 = x2

f1 = f2

! New Point 2 = Current Point 3

x2 = xIntercept

f2 = fIntercept

**end** **do**

root = xIntercept

**end** **subroutine** secantRoot

# rootFindingFunction.f08

**real** **function** rootFindingFunction(x)

**implicit** **none**

**real**, **intent**(in) :: x

rootFindingFunction = (3 \* x) + *sin*(x) - *exp*(x)

! rootFindingFunction = x\*\*11 - 1

**end** **function** rootFindingFunction

# displayRoot.f08

**subroutine** displayRoot(methodName, root, error, iterations)

**implicit** **none**

**character**(*len*=25), **intent**(in) :: methodName

**real**, **intent**(in) :: root, error

**integer**, **intent**(in) :: iterations

**write**(\*,10) "|", methodName, "|", root, "|", error, "|", iterations, "|"

10 **format**(a3, a25, a3, f10.4, a3, f10.4, a3, i12, a3)

**end** **subroutine** displayRoot

# Output

## Output 1 [(3 \* x) + *sin*(x) - *exp*(x)]

Enter the first guess for the root:

0.8

Enter the second guess for the root:

1.0

Enter the tolerance:

1e-6

| Method | Root | Error | Iterations |

|Secant | 0.3604 | 0.0000 | 7 |

## Output 2: f(x) is highly nonlinear [x\*\*11 - 1]

Enter the first guess for the root:

2.0

Enter the second guess for the root:

4.0

Enter the tolerance:

1e-6

| Method | Root | Error | Iterations |

|Secant | 1.0000 | 0.0000 | 18 |