Evaluate the function at various points to estimate roots:

**Code:**

%evaluates an input function every x units

lowerBound = -10;

upperBound = 10;

xValue = lowerBound;

step = 0.5; %space between evaluated locations

results = []; %array to hold the results

while xValue <= upperBound %evaluate the function at various points

y = f1(xValue); %f1 is a separate function file

fprintf('At x = %f, y = %f\n', xValue, y);

results = [results f1(xValue)]; %append the result in an array.

xValue += step;

end;

**Output:**

At x = -10.000000, y = 5508.264186

At x = -9.500000, y = 4048.910570

At x = -9.000000, y = 2841.426093

At x = -8.500000, y = 1858.176775

At x = -8.000000, y = 1073.028640

At x = -7.500000, y = 461.347707

At x = -7.000000, y = -0.000000

At x = -6.500000, y = -332.648461

At x = -6.000000, y = -556.731654

At x = -5.500000, y = -690.883557

At x = -5.000000, y = -752.238148

At x = -4.500000, y = -756.429406

At x = -4.000000, y = -717.591310

At x = -3.500000, y = -648.357837

At x = -3.000000, y = -559.862967

At x = -2.500000, y = -461.740676

At x = -2.000000, y = -362.124944

At x = -1.500000, y = -267.649750

At x = -1.000000, y = -183.449071

At x = -0.500000, y = -113.156885

At x = 0.000000, y = -58.907172

At x = 0.500000, y = -21.333909

At x = 1.000000, y = 0.428924

At x = 1.500000, y = 8.747351

At x = 2.000000, y = 7.487392

At x = 2.500000, y = 2.015069

At x = 3.000000, y = -0.803596

At x = 3.500000, y = 7.397418

At x = 4.000000, y = 36.484133

At x = 4.500000, y = 97.822570

At x = 5.000000, y = 204.278752

At x = 5.500000, y = 370.218699

At x = 6.000000, y = 611.508435

At x = 6.500000, y = 945.513979

At x = 7.000000, y = 1391.101354

At x = 7.500000, y = 1968.636581

At x = 8.000000, y = 2699.985683

At x = 8.500000, y = 3608.514680

At x = 9.000000, y = 4719.089595

At x = 9.500000, y = 6058.076448

At x = 10.000000, y = 7653.341262

Secant Method to find roots:

**Code:**

%Secant Method root search

p = [-7.5 -6.5]; %array of p values with the 2 initial values

i = 2 %start with p0 and p1

maxIteration = 15;

epsilon = 0.0001;

while i <= maxIteration

y = f1(p(i));

fprintf('iteration %f p=%f y=%f\n', i, p(i), y);

newval = p(i) - (f1(p(i))\*(p(i)-p(i-1))/(f1(p(i)) - f1(p(i-1))));

if abs(newval - p(i)) < epsilon %check if desired accuracy is met

fprintf('Answer found:%f with f(x)=%f\n',newval, f1(newval));

break;

end;

p = [p newval];

++i;

end;

**Output:** (each section is running the secant method with different bounds)

iteration 2.000000 p=-6.500000 y=-332.648461

iteration 3.000000 p=-6.918955 y=-62.111184

iteration 4.000000 p=-7.015140 y=11.974975

iteration 5.000000 p=-6.999593 y=-0.320076

iteration 6.000000 p=-6.999998 y=-0.001578

Answer found:-7.000000 with f(x)=0.000000

iteration 2.000000 p=1.000000 y=0.428924

iteration 3.000000 p=0.990145 y=0.140084

iteration 4.000000 p=0.985366 y=-0.001905

Answer found:0.985430 with f(x)=0.000008

iteration 2.000000 p=3.000000 y=-0.803596

iteration 3.000000 p=2.857451 y=-0.729715

iteration 4.000000 p=1.449520 y=8.418472

iteration 5.000000 p=2.745146 y=-0.182636

iteration 6.000000 p=2.717634 y=0.004620

iteration 7.000000 p=2.718313 y=-0.000224

Answer found:2.718282 with f(x)=-0.000000

iteration 2.000000 p=3.500000 y=7.397418

iteration 3.000000 p=3.048994 y=-0.635033

iteration 4.000000 p=3.084650 y=-0.441649

iteration 5.000000 p=3.166080 y=0.243087

iteration 6.000000 p=3.137171 y=-0.040397

iteration 7.000000 p=3.141291 y=-0.002790

iteration 8.000000 p=3.141597 y=0.000037

Answer found:3.141593 with f(x)=-0.000000