

```

> # Define the function
f := x → exp(x) + 0.33 * x^2 - 2;

# Initial guesses
x0 := -4.5;
x1 := -1.7;

# Parameters
tol := 1e-6;
maxIter := 6; # we want up to x6

# Secant Method Loop
for n from 2 to maxIter do
    # Apply secant formula
    x2 := evalf( (x1 - f(x1)) * (x1 - x0) / (f(x1) - f(x0)) );

    printf( "x[%d] = %.12f, f(x[%d]) = %.12f\n", n, x2, n, f(x2) );

    # Stop if within tolerance
    if abs( f(x2) ) < tol then
        printf( "Converged to root at x = %.12f after %d iterations\n", x2, n );
        break;
    end if;

    # Shift values
    x0 := x1;
    x1 := x2;
end do;

```

$$f := x \mapsto e^x + 0.33 \cdot x^2 - 2$$

$$x_0 := -4.5$$

$$x_1 := -1.7$$

$$tol := 1. \times 10^{-6}$$

$$maxIter := 6$$

$$x_2 := -2.135131910$$

$$x[2] = -2.135131910000, \quad f(x[2]) = -0.377370876000$$

$$x_0 := -1.7$$

$$x_1 := -2.135131910$$

$$x_2 := -2.472833907$$

$$x[3] = -2.472833907000, \quad f(x[3]) = 0.102264979000$$

```

                                x0 := -2.135131910
                                x1 := -2.472833907
                                x2 := -2.400831184
x[4] = -2.400831184000, f(x[4]) = -0.007240596000

                                x0 := -2.472833907
                                x1 := -2.400831184
                                x2 := -2.405592062
x[5] = -2.405592062000, f(x[5]) = -0.000119785000

                                x0 := -2.400831184
                                x1 := -2.405592062
                                x2 := -2.405672149
x[6] = -2.405672149000, f(x[6]) = 0.000000145000
Converged to root at x = -2.405672149000 after 6 iterations
=
> # DONE (
=
>

```