ALGORITHM

tic %For finding the execution time

format long g %for a better decimal precision

Declare two empty vectors / arrays f, k and an index i=1 to the arrays

WHILE x goes from 1.0 to 2.0

y <- x.^3+4.\*x.^2-10

g <- toc

f(i) <- y

k(i) <- g

i <- i+1

if y>0

x <- x-0.1

break

end

end

y1 <- x

WHILE x goes from y1 to y1+0.1

y <- x.^3 + 4.\*x.^2 - 10

g <- toc

f(i) <- y

k(i) <- g

i <- i+1

if y>0

x <- x-0.01

break

end

end

y1 <- x

WHILE x goes from y1 to y1+0.01

y <- x.^3 + 4.\*x.^2 - 10

g <- toc

f(i) <- y

k(i) <- g

i <- i+1

if y>0

x <- x-0.001

break

end

end

y1 <- x

WHILE x goes from y1 to(y1+0.001)

y <- x.^3 + 4.\*x.^2 - 10

g <- toc

f(i) <- y

k(i) <- g

i <- i+1

if y>0

x <- x-0.0001

break

end

end

y1 <- x

WHILE x <- y1 goes from (y1+0.0001)

y <- x.^3 + 4.\*x.^2 - 10

g <- toc

f(i) <- y

k(i) <- g

i <- i+1

if y>0

x <- x-0.00001

break

end

end

y1 <- x

for x <- y1:+0.000001:(y1+0.00001)

y <- x.^3 + 4.\*x.^2 - 10

g <- toc

f(i) <- y

k(i) <- g

i <- i+1

if y>0

x <- x-0.000001

break

end

end

plot(f,k) //Plots the precision with the speed of precision

toc %Finally gives the total program execution time