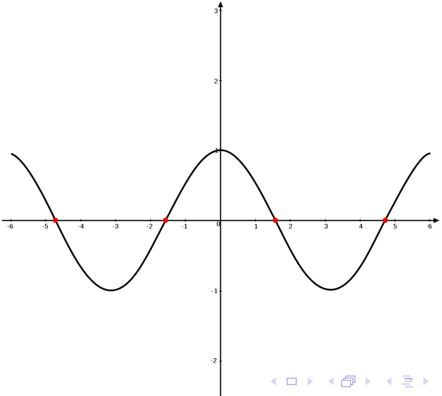
Math 4329: Numerical Analysis Chapter 03: Bisection Method

Natasha S. Sharma, PhD

Mathematical question we are interested in numerically answering

- How to find the x-intercepts of a function $f(x)$? These x-intercepts are called the roots of the equation $f(x) = 0$. **Notation: denote the exact root by α. That means, $f(\alpha) = 0$.**



Naive Approach

- Plotting the function and reading off the x-intercepts presents a graphical approach to finding the roots. This approach can be impractical.
- Instead, we seek approaches to get a formula for the root in terms of x.
  For example, if $f(x) = 3x + 4$, the root to $3x + 4 = 0$ is $x = -\frac{4}{3}$.
  If $f(x) = e^x \sin(x) - x$ the root to $e^x \sin(x) - x = 0$ is $x = 0$
- We use the numerical approach in cases when it is difficult to get a formula for the root.
  What is the root to $f(x) = e^x \cos(x) - x = 0$?

Naive Approach

- Plotting the function and reading off the x-intercepts presents a graphical approach to finding the roots. This approach can be impractical.
- Instead, we seek approaches to get a formula for the root in terms of x.
  For example, if $f(x) = 3x + 4$, the root to $3x + 4 = 0$ is $x = -\frac{4}{3}$.
  If $f(x) = e^x \sin(x) - x$ the root to $e^x \sin(x) - x = 0$ is $x = 0$
- We use the numerical approach in cases when it is difficult to get a formula for the root.
  What is the root to $f(x) = e^x \cos(x) - x = 0$?

Naive Approach

- Plotting the function and reading off the x-intercepts presents a graphical approach to finding the roots. This approach can be impractical.
- Instead, we seek approaches to get a formula for the root in terms of x.
  For example, if $f(x) = 3x + 4$, the root to $3x + 4 = 0$ is $x = -\frac{4}{3}$.
  If $f(x) = e^x \sin(x) - x$ the root to $e^x \sin(x) - x = 0$ is $x = 0$
- We use the numerical approach in cases when it is difficult to get a formula for the root.
  What is the root to $f(x) = e^x \cos(x) - x = 0$?

Naive Approach

- Plotting the function and reading off the x-intercepts presents a graphical approach to finding the roots. This approach can be impractical.
- Instead, we seek approaches to get a formula for the root in terms of x.
  For example, if $f(x) = 3x + 4$, the root to $3x + 4 = 0$ is $x = -\frac{4}{3}$.
  If $f(x) = e^x \sin(x) - x$ the root to $e^x \sin(x) - x = 0$ is $x = 0$
- We use the numerical approach in cases when it is difficult to get a formula for the root.
  What is the root to $f(x) = e^x \cos(x) - x = 0$?

Roadmap for the numerical method to finding root

Each of the numerical approaches fit the following structure:

1. Start with an initial guess x_0 and set an error tolerance $\varepsilon > 0$. For instance, $\varepsilon = 10^{-4}$.
2. Generate a sequence of approximations to α $x_1, \ x_2, \cdots, \ x_n \cdots$ such that $f(x_n)$ is getting closer to 0. How close is good enough?
3. $$|f(x_n)| < \varepsilon \text{ and } |x_n - x_{n-1}| < \varepsilon.$$
4. Such methods are called iterative methods because it is based on the iterations indexed by n generating the approximations to the root α. x_n are called the iterates.

Roadmap for the numerical method to finding root

Each of the numerical approaches fit the following structure:

1. Start with an initial guess x_0 and set an error tolerance $\varepsilon > 0$. For instance, $\varepsilon = 10^{-4}$.
2. Generate a sequence of approximations to α $x_1,\ x_2, \cdots,\ x_n \cdots$ such that $f(x_n)$ is getting closer to 0. How close is good enough?
3. $$|f(x_n)| < \varepsilon \text{ and } |x_n - x_{n-1}| < \varepsilon.$$
4. Such methods are called iterative methods because it is based on the iterations indexed by n generating the approximations to the root α.
   x_n are called the iterates.

Roadmap for the numerical method to finding root

Each of the numerical approaches fit the following structure:

1. Start with an initial guess x_0 and set an error tolerance $\varepsilon > 0$. For instance, $\varepsilon = 10^{-4}$.
2. Generate a sequence of approximations to α $x_1, x_2, \cdots, x_n \cdots$ such that $f(x_n)$ is getting closer to 0. How close is good enough?
3. $$|f(x_n)| < \varepsilon \text{ and } |x_n - x_{n-1}| < \varepsilon.$$
4. Such methods are called iterative methods because it is based on the iterations indexed by n generating the approximations to the root α. x_n are called the iterates.

Roadmap for the numerical method to finding root

Each of the numerical approaches fit the following structure:

1. Start with an initial guess x_0 and set an error tolerance $\varepsilon > 0$. For instance, $\varepsilon = 10^{-4}$.
2. Generate a sequence of approximations to α $x_1,\ x_2, \cdots,\ x_n \cdots$ such that $f(x_n)$ is getting closer to 0. How close is good enough?
3. $$|f(x_n)| < \varepsilon \text{ and } |x_n - x_{n-1}| < \varepsilon.$$
4. Such methods are called iterative methods because it is based on the iterations indexed by n generating the approximations to the root α.
   x_n are called the iterates.

Goals

1. Explore numerical methods/algorithms to find approximate roots of the an equation $f(x) = 0$.
2. Design* our own numerical methods/algorithms to obtain an approximate root.
    - Bisection Method
    - Newton's Method
    - Secant Method
    - General theory to design our own methods (* One-Point Iteration Methods)

Chaper 3: Rootfinding

Goals

1. Explore numerical methods/algorithms to find approximate roots of the an equation $f(x) = 0$.
2. Design* our own numerical methods/algorithms to obtain an approximate root.
    1. Bisection Method
    2. Newton's Method
    3. Secant Method
    4. General theory to design our own methods (* One-Point Iteration Methods)

Goals

1. Explore numerical methods/algorithms to find approximate roots of the an equation $f(x) = 0$.
2. Design* our own numerical methods/algorithms to obtain an approximate root.
   1. Bisection Method
   2. Newton's Method
   3. Secant Method
   4. General theory to design our own methods (* One-Point Iteration Methods)

Chaper 3: Rootfinding

Goals

1. Explore numerical methods/algorithms to find approximate roots of the an equation $f(x) = 0$.
2. Design* our own numerical methods/algorithms to obtain an approximate root.
    1. Bisection Method
    2. Newton's Method
    3. Secant Method
    4. General theory to design our own methods (* One-Point Iteration Methods)

Chaper 3: Rootfinding

Goals

1. Explore numerical methods/algorithms to find approximate roots of the an equation $f(x) = 0$.
2. Design* our own numerical methods/algorithms to obtain an approximate root.
    1. Bisection Method
    2. Newton's Method
    3. Secant Method
    4. General theory to design our own methods (* One-Point Iteration Methods)

Goals

1. Explore numerical methods/algorithms to find approximate roots of the an equation $f(x) = 0$.
2. Design* our own numerical methods/algorithms to obtain an approximate root.
   1. Bisection Method
   2. Newton's Method
   3. Secant Method
   4. General theory to design our own methods (* One-Point Iteration Methods)

Goals

1. Explore numerical methods/algorithms to find approximate roots of the an equation $f(x) = 0$.
2. Design* our own numerical methods/algorithms to obtain an approximate root.
   1. Bisection Method
   2. Newton's Method
   3. Secant Method
   4. General theory to design our own methods (* One-Point Iteration Methods)

Towards Bisection Method

1. Estimate the approximate location of α.
   That is, find an interval $[a, b]$ containing α.
   - Intermediate Value Theorem **[Appendix A]**:
     If f is continuous on $[a, b]$ and $f(a) \cdot f(b) < 0$ then f has atleast one zero in (a, b).
2. Repeatedly half the interval containing the root (based on the Intermediate Value Theorem).
   That is, trap the root in shrinking interval by generating a sequence of iterates $\{c_n\}_{n \geq 0}$: $c_1, \ c_2, \cdots, \ c_n \cdots$ which live in $[a, b]$ and converge to α.

Towards Bisection Method

1. Estimate the approximate location of α.
   That is, find an interval $[a, b]$ containing α.
   - Intermediate Value Theorem **[Appendix A]**:
     If f is continuous on $[a, b]$ and $f(a) \cdot f(b) < 0$ then f has atleast one zero in (a, b).
2. Repeatedly half the interval containing the root (based on the Intermediate Value Theorem).
   That is, trap the root in shrinking interval by generating a sequence of iterates $\{c_n\}_{n \geq 0}$: $c_1, \ c_2, \cdots, \ c_n \cdots$ which live in $[a, b]$ and converge to α.

Towards Bisection Method

1. Estimate the approximate location of α.
   That is, find an interval $[a, b]$ containing α.
   - Intermediate Value Theorem **[Appendix A]**:
     If f is continuous on $[a, b]$ and $f(a) \cdot f(b) < 0$ then f has atleast one zero in (a, b).
2. Repeatedly half the interval containing the root (based on the Intermediate Value Theorem).
   That is, trap the root in shrinking interval by generating a sequence of iterates $\{c_n\}_{n\geq 0}: \ c_1, \ c_2, \cdots, \ c_n \cdots$ which live in $[a, b]$ and converge to α.

Towards Bisection Method

1. Estimate the approximate location of α.
   That is, find an interval $[a, b]$ containing α.
   - Intermediate Value Theorem **[Appendix A]**:
     If f is continuous on $[a, b]$ and $f(a) \cdot f(b) < 0$ then f has atleast one zero in (a, b).
2. Repeatedly half the interval containing the root (based on the Intermediate Value Theorem).
   That is, trap the root in shrinking interval by generating a sequence of iterates $\{c_n\}_{n \geq 0}$: $c_1, \; c_2, \cdots, \; c_n \cdots$ which live in $[a, b]$ and converge to α.

Towards Bisection Method

1. Estimate the approximate location of α.
   That is, find an interval $[a, b]$ containing α.
   - Intermediate Value Theorem **[Appendix A]**:
     If f is continuous on $[a, b]$ and $f(a) \cdot f(b) < 0$ then f has atleast one zero in (a, b).
2. Repeatedly half the interval containing the root (based on the Intermediate Value Theorem).
   That is, trap the root in shrinking interval by generating a sequence of iterates $\{c_n\}_{n \geq 0}$: $c_1, \; c_2, \cdots, \; c_n \cdots$ which live in $[a, b]$ and converge to α.

Towards Bisection Method

1. Estimate the approximate location of α.
   That is, find an interval $[a, b]$ containing α.
   - Intermediate Value Theorem **[Appendix A]**:
     If f is continuous on $[a, b]$ and $f(a) \cdot f(b) < 0$ then f has atleast one zero in (a, b).
2. Repeatedly half the interval containing the root (based on the Intermediate Value Theorem).
   That is, trap the root in shrinking interval by generating a sequence of iterates $\{c_n\}_{n \geq 0}$: $c_1, c_2, \cdots, c_n \cdots$ which live in $[a, b]$ and converge to α.

Bisection Method

Suppose that we can find $a < b$ such that $f(a) \cdot f(b) < 0$. Let $\varepsilon > 0$ denote the given error tolerance.

B1 Define $c = \frac{a+b}{2}$.

B2 If $b - c \leq \varepsilon$, then accept c as the root and stop.

B3 If $\text{sign}[f(b)] \cdot \text{sign}[f(c)] \leq 0$, then set $a = c$. Otherwise, set $b = c$. Return to B1.

Remarks

1. The interval $[a, b]$ is shrunk reducing by $1/2$ for each loop of steps B1–B3 .
2. The test B2 will be satisfied eventually, and with it the condition $|\alpha - c| \leq \varepsilon$ will be satisfied.
3. Note In B3 we test the $\text{sign}[f(b)] \cdot \text{sign}[f(c)]$ in order to avoid the under or overflow due to multiplication of $f(b)$ and $f(c)$.

Suppose that we can find $a < b$ such that $f(a) \cdot f(b) < 0$. Let $\varepsilon > 0$ denote the given error tolerance.

B1 Define $c = \frac{a+b}{2}$.

B2 If $b - c \leq \varepsilon$, then accept c as the root and stop.

B3 If $\text{sign}[f(b)] \cdot \text{sign}[f(c)] \leq 0$, then set $a = c$. Otherwise, set $b = c$. Return to B1.

Remarks

1. The interval $[a, b]$ is shrunk reducing by $1/2$ for each loop of steps B1–B3 .
2. The test B2 will be satisfied eventually, and with it the condition $|\alpha - c| \leq \varepsilon$ will be satisfied.
3. Note In B3 we test the $\text{sign}[f(b)] \cdot \text{sign}[f(c)]$ in order to avoid the under or overflow due to multiplication of $f(b)$ and $f(c)$.

Bisection Method

Suppose that we can find $a < b$ such that $f(a) \cdot f(b) < 0$. Let $\varepsilon > 0$ denote the given error tolerance.

B1 Define $c = \frac{a+b}{2}$.

B2 If $b - c \leq \varepsilon$, then accept c as the root and stop.

B3 If $\text{sign}[f(b)] \cdot \text{sign}[f(c)] \leq 0$, then set $a = c$. Otherwise, set $b = c$. Return to B1.

Remarks

1. The interval $[a, b]$ is shrunk reducing by $1/2$ for each loop of steps B1–B3 .
2. The test B2 will be satisfied eventually, and with it the condition $|\alpha - c| \leq \varepsilon$ will be satisfied.
3. Note In B3 we test the $\text{sign}[f(b)] \cdot \text{sign}[f(c)]$ in order to avoid the under or overflow due to multiplication of $f(b)$ and $f(c)$.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.

Read the questions carefully about the kind of root we are looking for!

Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.

Location of the root α is in [1,2].

Note: This interval need not be unique! [0,2] also works!

But the smaller the interval the faster the root finding method will work.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.
Read the questions carefully about the kind of root we are looking for!
Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.
Location of the root α is in [1,2].
Note: This interval need not be unique! [0,2] also works!
But the smaller the interval the faster the root finding method will work.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.
Read the questions carefully about the kind of root we are looking for!
Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.
Location of the root α is in [1,2].
Note: This interval need not be unique! [0,2] also works!
But the smaller the interval the faster the root finding method will work.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.

Read the questions carefully about the kind of root we are looking for!

Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.

Location of the root α is in [1,2].

Note: This interval need not be unique! [0,2] also works!

But the smaller the interval the faster the root finding method will work.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.

Read the questions carefully about the kind of root we are looking for!

Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.

Location of the root α is in [1,2].

Note: This interval need not be unique! [0,2] also works!

But the smaller the interval the faster the root finding method will work.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.
Read the questions carefully about the kind of root we are looking for!
Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.
Location of the root α is in [1,2].
Note: This interval need not be unique! [0,2] also works!
But the smaller the interval the faster the root finding method will work.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.

Read the questions carefully about the kind of root we are looking for!

Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.

Location of the root α is in [1,2].

Note: This interval need not be unique! [0,2] also works! But the smaller the interval the faster the root finding method will work.

Example

Note that we could be specifically interested in finding the smallest or the largest positive root or negative root.

Read the questions carefully about the kind of root we are looking for!

Example: Find the largest root of

$$f(x) = x^6 - x - 1 = 0$$

accurate within $\varepsilon = 0.001$.

Location of the root α is in [1,2].

Note: This interval need not be unique! [0,2] also works!

But the smaller the interval the faster the root finding method will work.

Performance of the bisection Method

n	a	b	c	b-c	f(c)
1	1	2	1.5	0.5	8.8906
2	1	1.5	1.25	0.25	1.5647
3	1	1.25	1.125	0.125	-0.0977
4	1.125	1.25	1.1875	0.0625	0.6167
5	1.125	1.1875	1.1562	0.0312	0.2333
6	1.125	1.1562	1.1406	0.0156	0.0616
7	1.125	1.1406	1.1328	0.0078	-0.0196
8	1.1328	1.1406	1.1367	0.0039	0.0206
9	1.1328	1.1367	1.1348	0.0020	0.0004
10	1.1328	1.1348	1.1338	0.00098	-0.0096

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \to 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \to 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \to 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \to 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \to 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \to 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \to 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \to 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \to 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \to 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \rightarrow 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \rightarrow 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \to 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \to 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Remarks on the Performance

1. Observe the shrinking of the interval $[a, b]$ as $n \to 10$. This shrinking is
   - Dictated by the value of $f(c)$.
   - This shrinking is by a factor of $1/2$ as illustrated by the column $b - c$.
2. Look at the initial **rapid** decay in the value of $f(c)$ as $n \to 10$:
   - For $n = 1$, the reduction is by a factor of 5.7.
   - For $n = 2$, the reduction is by 16.
   - For $n = 3$, the factor is 0.1584, for $n = 4$ the factor is 2.6 etc.
3. Numerically, one can also observe the **impact of the round-off errors on the calculations**.

Error Bounds

We want to know how many loops of the Bisection Method need to run to achieve a $\varepsilon > 0$ level of accuracy?
On the next slide, we present the theory behind determining n, the number of iterations needed to achieve a ε accuracy,

Error Bounds

Let $a_n,\ b_n,\ c_n$ denote the computed values of $a,\ b,\ c$ at the n^{th} iteration. We noticed the following relationship:

$$b_{n+1} - a_{n+1} = \frac{1}{2}(b_n - a_n), \quad n \geq 1$$

and

$$b_n - a_n = \frac{1}{2^{n-1}}(b - a)$$

where $b - a$ denotes the length of the initial interval satisfying $f(a) \cdot f(b) < 0$.
Since the root α is trapped in the shrinking interval $[a_n, c_n]$ or $[c_n, b_n]$, we have:

$$|\alpha - c_n| \leq c_n - a_n = b_n - c_n = \frac{1}{2}(b_n - a_n) \leq \frac{1}{2}\left(\frac{1}{2^{n-1}}(b - a)\right)$$

$$|\alpha - c_n| \leq \cdots \leq \frac{1}{2}\left(\frac{1}{2^{n-1}}(b-a)\right) = \frac{1}{2^n}(b-a).$$

As $n \to \infty$, the iterates $c_n \to \alpha$.

The question we are interested in answering:
How fast will we be within ε-distance from the root α?

$$|\alpha - c_n| \leq \cdots \leq \frac{1}{2}\left(\frac{1}{2^{n-1}}(b-a)\right) = \frac{1}{2^n}(b-a).$$

As $n \to \infty$, the iterates $c_n \to \alpha$.

The question we are interested in answering:
How fast will we be within ε-distance from the root α?

That is, for what n will the following error bound hold?
Keep in mind, this is without any a-priori information about α and without calculating all the iterations c_n!

$$|\alpha - c_n| \leq \varepsilon = 10^{-3}$$

$$\begin{aligned} |\alpha - c_n| \leq \cdots \leq \frac{1}{2}\left(\frac{1}{2^{n-1}}(b-a)\right) \\ = \frac{1}{2^n}(b-a) \\ \leq 0.001 \end{aligned}$$

Find n such that $\quad n \geq \frac{\log(\frac{b-a}{\varepsilon})}{\log 2} \quad$ holds that is equivalent to

$$n \geq \frac{\log(\frac{1}{0.001})}{\log 2} \approx 9.97$$

Exercise

Repeat the above exercise with $f(x) = x - \cos(x)$, (x measured in radians).