**Practical 3:**

**Objective: Finding roots of equations: Bisection Method**

The bisection method in mathematics is a root-finding method that repeatedly bisects an interval and then selects a subinterval in which a root must lie for further processing. It is a very simple and robust method, but it is also relatively slow.

The method is applicable for numerically solving the equation  for the  [real](https://en.wikipedia.org/wiki/Real_number" \o "Real number) variable , where  is a [continuous function](https://en.wikipedia.org/wiki/Continuous_function) defined on an interval and where  and  have opposite signs. In this case  and  are said to bracket a root since, by the [intermediate value theorem](https://en.wikipedia.org/wiki/Intermediate_value_theorem), the continuous function  must have at least one root in the interval .

**Initial Requirements:** We have an initial bound on the root, that is, f(*a*) and (*b*) have opposite signs.

**Iteration Process:** Given the interval , define. Then

* if (unlikely in practice), then halt, as we have found a root,
* if and have opposite signs, then a root must lie on , so assign ,
* else and must have opposite signs, and thus a root must lie on , so assign .

**Halting Conditions:** There are three conditions which may cause the iteration process to halt:

1. As indicated, if .
2. We halt if both of the following conditions are met:
   * The width of the interval (after the assignment) is sufficiently small, that is *b* - *a* < εstep, and ( ex)a=0.111111 –b= 0.111118)
   * The function evaluated at one of the end point || or || < εabs.
3. If we have iterated some maximum number of times, say *N*, and have not met Condition 1, we halt and indicate that a solution was not found.

If we halt due to Condition 1, we state that is our approximation to the root. If we halt according to Condition 2, we choose either  or , depending on whether || < | or | > ||, respectively.

If we halt due to Condition 3, then we indicate that a solution may not exist (the function may be discontinuous).

**Sample Pseudo-code :**

INPUT: Function *f*, endpoint values *a*, *b*, tolerance *TOL(-=diffent from stoping criteria)*, maximum iterations *NMAX*

CONDITIONS: *a* < *b*, either *f*(*a*) < 0 and *f*(*b*) > 0 or *f*(*a*) > 0 and *f*(*b*) < 0

OUTPUT: value which differs from a root of *f*(*x*)=0 by less than *TOL*

*N* ← 1

**While** *N* ≤ *NMAX* *# limit iterations to prevent infinite loop*

*c* ← (*a* + *b*)/2 *# new midpoint*

**If** *f*(*c*) = 0 or (*b* – *a*)/2 < *TOL* **then** *# solution found*

Output(*c*)

**Stop**

**EndIf**

*N* ← *N* + 1 *# increment step counter*

**If** sign(*f*(*c*)) = sign(*f*(*a*)) **then** *a* ← *c* **else** *b* ← *c* *# new interval*

**EndWhile**

Output("Method failed.") *# max number of steps exceeded*

**Partial Code (MATLAB):**

>> format long

>> eps\_abs = 1e-5;

>> eps\_step = 1e-5;

>> a = 0.0;

>> b = 2.0;

>> while (b - a >= eps\_step || ( abs( f(a) ) >= eps\_abs && abs( f(b) ) >= eps\_abs ) )

c = (a + b)/2;

if ( f(c) == 0 )

break;

elseif ( f(a)\*f(c) < 0 )

b = c;

else

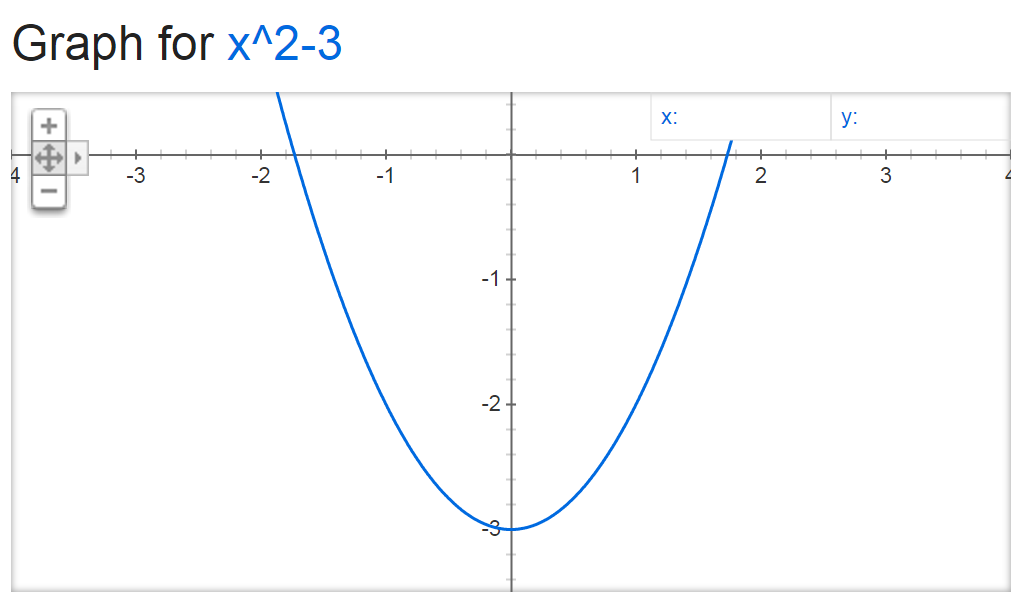
a = c;

end

end

**Part A:**

Consider finding the root of . Let εstep = 0.01, εabs = 0.01 (torelence val)and start with the interval [1, 2].



1. Write a program to compute .

Output format:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | **Update** |  |
| **1.0** | **2.0** | **-2.0** | **1.0** | **1.5** | **-0.75** | **a=c** | **0.5** |
| **1.5** | **2.0** | **-0.75** | **1.0.** | **1.75** | **0.0625** | **b=c** | **0.25** |
| **1.5** | **1.75** | **-0.75** | **0.0625** | **1.625** | **-0.35938** | **a=c** | **0.125** |
| **1.625** | **1.75** | **-0.35938** | **0.0625** | **1.6875** | **-0.15234** | **a=c** | **0.0625** |
| **1.6875** | **1.75** | **-0.15234** | **0.0625** | **1.71875** | **-0.0459** | **a=c** | **0.03125** |
| **1.71875** | **1.75** | **-0.0459** | **0.0625** | **1.73438** | **0.00806** | **b=c** | **0.01563** |
| **1.71875** | **1.73438** | **-0.0459** | **0.00806** | **1.72656** | **-0.01898** | **a=c** | **0.00781** |

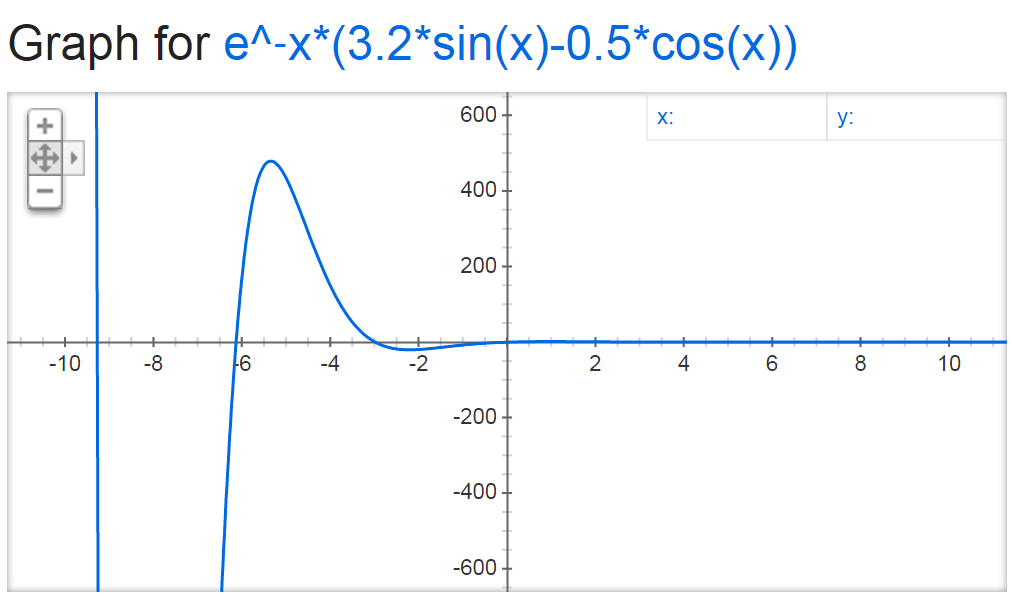
1. What is your approximate of the root?

Root : c= 1.73438

**root : c = 1.72656 X**

**Part B:**

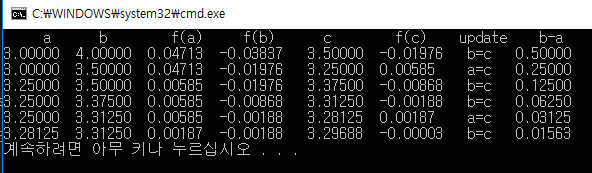
Consider finding the root of on the interval , this time with εstep  = 0.001, εabs = 0.001 .



1. Write a program to compute .

Output format: X틀림 (pow(x,-1) 이 아니라 exp(-x)로 해야함)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | **Update** |  |
| **3** | **4** | **0.31553** | **-0.52374** | **3.5** | **-0.18694** | **b=c** | **0.5** |
| **3** | **3.5** | **0.31553** | **-0.18694** | **3.25** | **0.04641** | **a=c** | **0.25** |
| **3.25** | **3.5** | **0.04641** | **-0.18694** | **3.375** | **-0.07517** | **b=c** | **0.125** |
| **3.25** | **3.375** | **0.04641** | **-0.07517** | **3.3125** | **-0.01556** | **b=c** | **0.0625** |
| **3.25** | **3.3125** | **0.04641** | **-0.01556** | **3.28125** | **0.01514** | **a=c** | **0.03125** |
| **3.28125** | **3.3125** | **0.01514** | **-0.01556** | **3.29688** | **-0.00028** | **b=c** | **0.01563** |
| **3.28125** | **3.29688** | **0.01514** | **-0.00028** | **3.28906** | **0.00741** | **a=c** | **0.00781** |
| **3.28906** | **3.29688** | **0.00741** | **-0.00028** | **3.29297** | **0.00356** | **a=c** | **0.00391** |
| **3.29297** | **3.29688** | **0.00356** | **-0.00028** | **3.29492** | **0.00164** | **a=c** | **0.00195** |
| **3.29492** | **3.29688** | **0.00164** | **-0.00028** | **3.2959** | **0.00068** | **a=c** | **0.00098** |

****

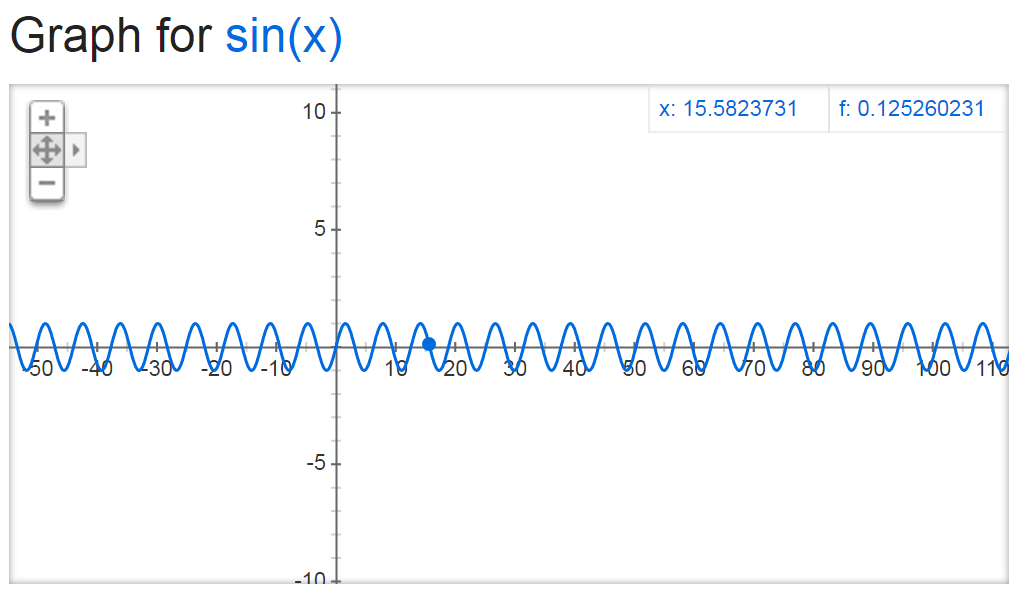
1. What is your approximate of the root?

Root of x = 3.29688

**Root : c=3.2959 X**

**Part C:**

Solve by using the bisection method: starting with [1, 99], εstep = εabs = 0.00001.



1. Write a program to compute .

Output format:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | ***Update*** |  |
| **1** | **99** | **0.84147** | **-0.99921** | **50** | **-0.26237** | **b=c** | **49** |
| **1** | **50** | **0.84147** | **-0.26237** | **25.5** | **0.35906** | **a=c** | **24.5** |
| **25.5** | **50** | **0.35906** | **-0.26237** | **37.75** | **0.05087** | **a=c** | **12.25** |
| **.** | **.** | **.** | **.** | **.** | **.** | **.** | **.** |
| **40.84070** | **40.84071** | **0** | **-0.00001** | **40.84071** | **-0.00000** | **b=c** | **0.00001** |

1. What is your approximate of the root?

We cannot get roots of this f(x) by bisection method.

Because there are too many roots in the domain [1,99] but we can only get one root by bisection method.