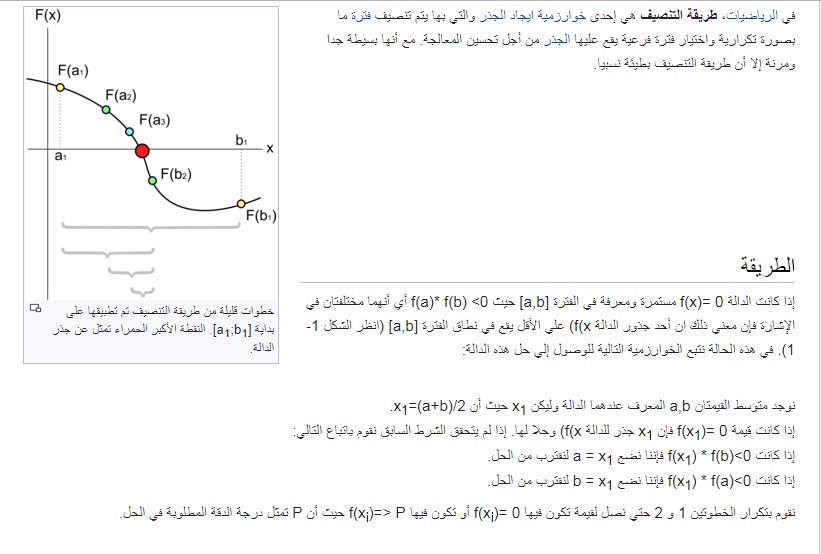
**Numerical Analysis to approximate Nonlinear equations**

1. **bisection method:-**

****

**bisection method**

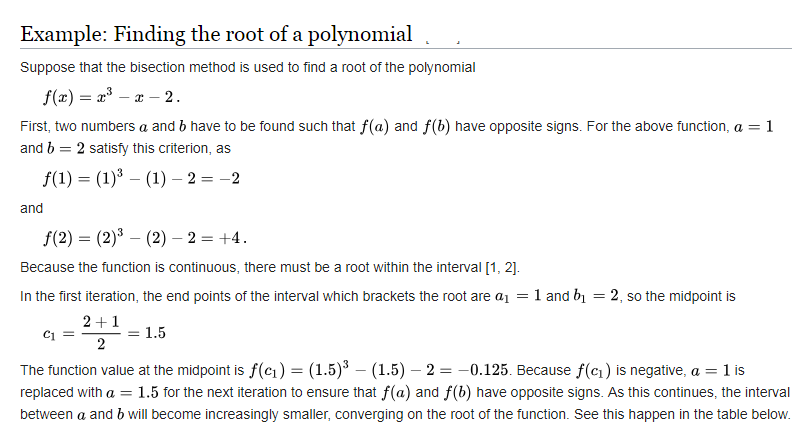
In [mathematics](https://en.wikipedia.org/wiki/Mathematics), the **bisection method** is a [root-finding method](https://en.wikipedia.org/wiki/Root-finding_method) that applies to any [continuous functions](https://en.wikipedia.org/wiki/Continuous_function) for which one knows two values with opposite signs. The method consists of repeatedly [bisecting](https://en.wikipedia.org/wiki/Bisection) the [interval](https://en.wikipedia.org/wiki/Interval_(mathematics)) defined by these values and then selecting the subinterval in which the function changes of sign, and therefore must contain a [root](https://en.wikipedia.org/wiki/Root_of_a_function). It is a very simple and robust method, but it is also relatively slow. Because of this, it is often used to obtain a rough approximation to a solution which is then used as a starting point for more rapidly converging methods

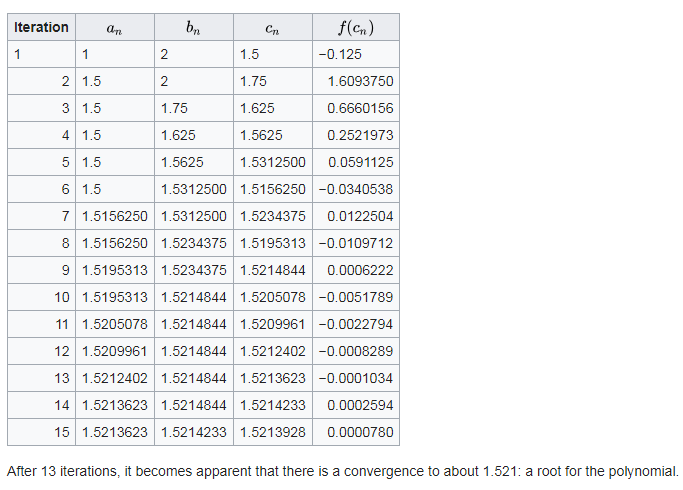
## The method

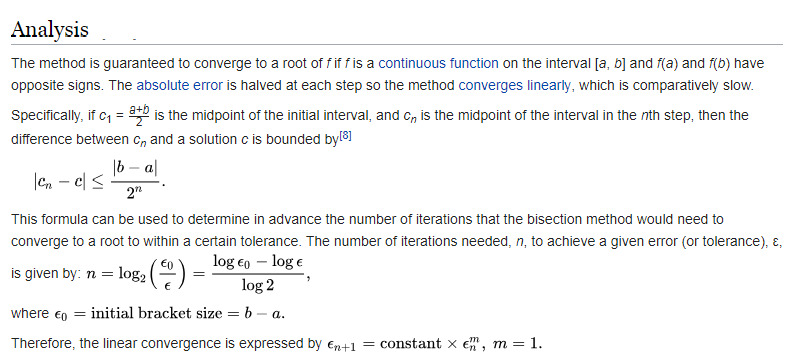
The method is applicable for numerically solving the equation *f*(*x*) = 0 for the [real](https://en.wikipedia.org/wiki/Real_number) variable *x*, where *f* is a [continuous function](https://en.wikipedia.org/wiki/Continuous_function) defined on an interval [*a*, *b*] and where *f*(*a*) and *f*(*b*) have opposite signs. In this case *a* and *b* are said to bracket a root since, by the [intermediate value theorem](https://en.wikipedia.org/wiki/Intermediate_value_theorem), the continuous function *f* must have at least one root in the interval (*a*, *b*).

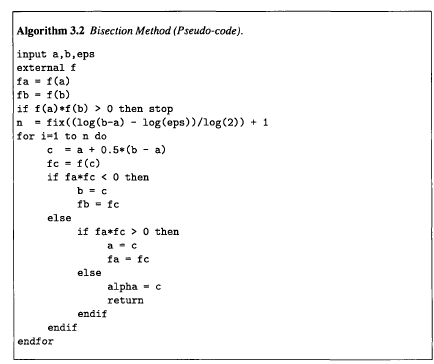
At each step the method divides the interval in two by computing the midpoint *c* = (*a*+*b*) / 2 of the interval and the value of the function *f*(*c*) at that point. Unless *c* is itself a root (which is very unlikely, but possible) there are now only two possibilities: either *f*(*a*) and *f*(*c*) have opposite signs and bracket a root, or *f*(*c*) and *f*(*b*) have opposite signs and bracket a root.[[5]](https://en.wikipedia.org/wiki/Bisection_method#cite_note-5) The method selects the subinterval that is guaranteed to be a bracket as the new interval to be used in the next step. In this way an interval that contains a zero of *f* is reduced in width by 50% at each step. The process is continued until the interval is sufficiently small.

Explicitly, if *f*(*a*) and *f*(*c*) have opposite signs, then the method sets *c* as the new value for *b*, and if *f*(*b*) and *f*(*c*) have opposite signs then the method sets *c* as the new *a*. (If *f*(*c*)=0 then *c* may be taken as the solution and the process stops.) In both cases, the new *f*(*a*) and *f*(*b*) have opposite signs, so the method is applicable to this smaller interval







****

**المصادر**

1. Kaw، Autar؛ Kalu، Egwu (2008)، Numerical Methods with Applications (الطبعة 1st). <http://numericalmethods.eng.usf.edu/topics/textbook_index.html>
2. Corliss، George (1977)، "Which root does the bisection algorithm find?"، *SIAM Review*، **19** (2): 325–327,I[SSN](https://ar.wikipedia.org/wiki/%D8%B1%D9%82%D9%85_%D8%AF%D9%88%D9%84%D9%8A_%D9%85%D8%B9%D9%8A%D8%A7%D8%B1%D9%8A_%D9%84%D9%84%D8%AF%D9%88%D8%B1%D9%8A%D8%A7%D8%AA) [1095-7200](https://www.worldcat.org/issn/1095-7200)، [doi](https://ar.wikipedia.org/wiki/%D9%85%D8%B9%D8%B1%D9%81_%D8%A7%D9%84%D8%A7%D8%B4%D9%8A%D8%A7%D8%A1_%D8%A7%D9%84%D8%B1%D9%82%D9%85%D9%8A%D8%A9):[10.1137/1019044](https://dx.doi.org/10.1137%2F1019044)