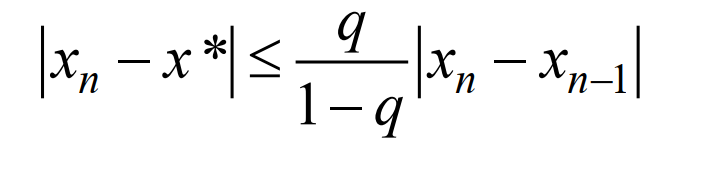
1. **Thuật toán tổng thể**

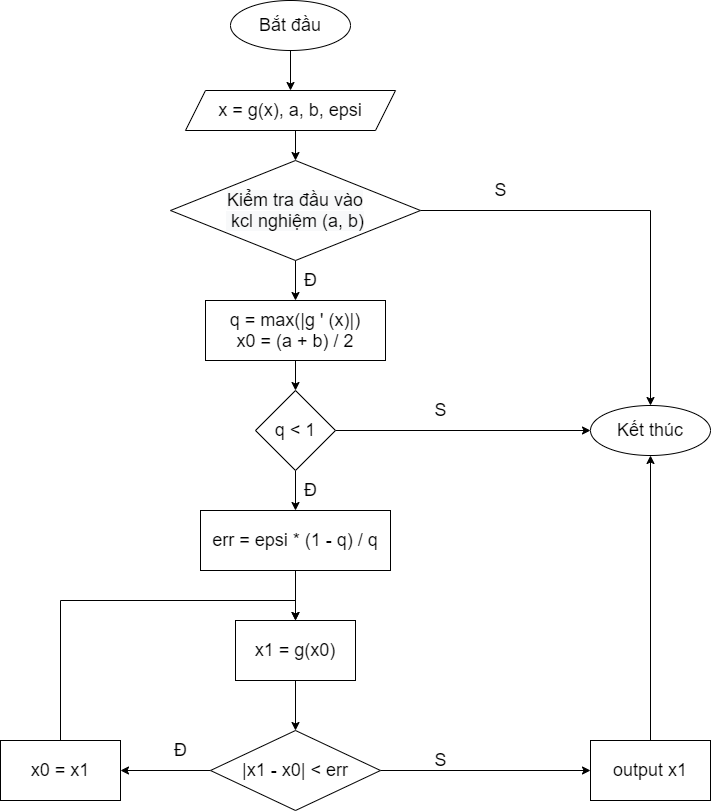
Input: f(x), khoảng phân li nghiệm (a, b), sai số epsi

Output: Nghiệm gần đúng x

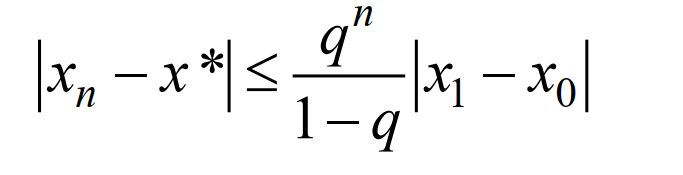
1. Thuật toán theo công thức hậu nghiệm

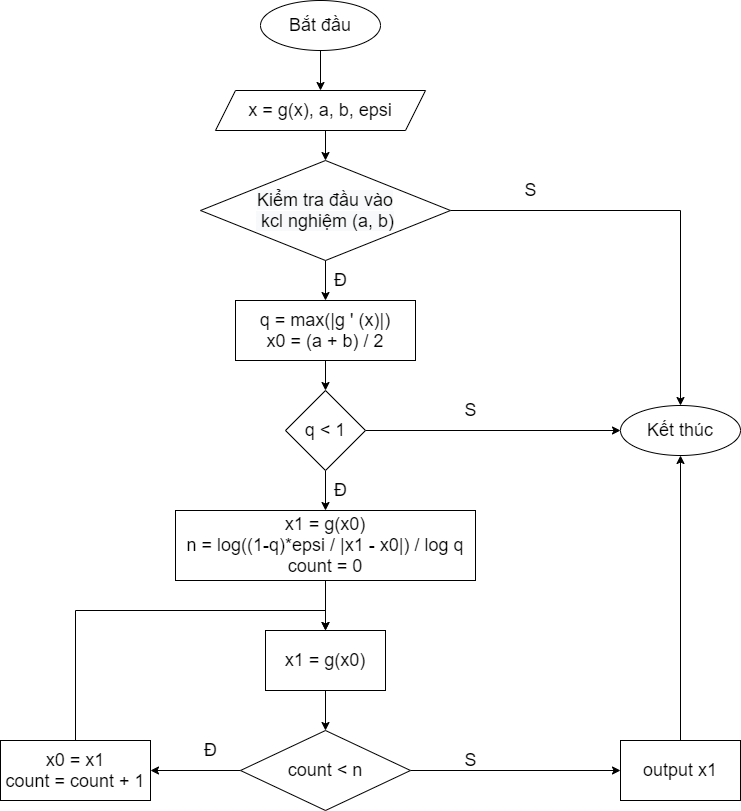
Công thức sai số hậu nghiệm:





1. Thuật toán theo công thức tiên nghiệm





1. **Thuật toán chi tiết. (Giả mã)**

|  |
| --- |
| 1. Hàm số g(x) = x được suy ra từ biểu thức f(x) = 0  input: x  output: g(x)  Function g:  return g(x) = x ///hàm số được suy ra từ f(x) = 0  2. Hàm tính max của |g’(x)| trên đoạn [a, b]  Input: a, b, g(x)  Output: max(|g’(x)|)  Function max\_dg  h = e-7  max = 0  delta = (a – b) / 10000  while (a <= b):  x = a  dy = f(x + h) – f(x – h)  dx = 2 \* h  if |dy / dx| > max:  max = |dy / dx|  a = a + delta  return max  3. Phương pháp lặp đơn  Input: g(x), a, b, epsi  Output: nghiệm gần đúng x  3.1. Thuật toán theo công thức tiên nghiệm  Function lap\_don\_TN:  if f(a) \* f(b) < 0:  q = max(g, a, b)  x0 = (a + b) / 2  if q < 1:  x1 = g(x0)  n = log((1 – q)\*epsi / |x1 – x0|) / log q  count = 0  while (count < n):  x1 = g(x0)  count = count + 1  x0 = x1  return x1  3.2. Thuật toán theo công thức hậu nghiệm  Function lap\_don\_HN:  if f(a) \* f(b) < 0:  q = max(g, a, b)  x0 = (a + b) / 2  x1 = x0  if q < 1:  err = epsi \* (1 – q) / q  do:  x0 = x1  x1 = g(x0)  while |x1 – x0| < err  return x1 |

1. **Ưu và nhược điểm của phương pháp**
2. Ưu điểm

* Tốc độ hội tụ nhanh
* Dễ cài đặt trong lập trình máy tính
* Hội tụ với giá trị x0 ban đầu bất kì trên đoạn [a, b]

1. Nhược điểm

* Điều kiện khoảng cách li nghiệm (a, b)
* Tính chất liên tục, khả vi của hàm số g(x) trên đoạn [a, b]
* Điều kiện hội tụ: q = max(|g’(x)|) < 1
* Khối lượng tính toán khá lớn do phải tìm max(|g’(x)|) trên đoạn [a, b] mất 10000 vòng lặp
* Không phải phương trình nào cũng dễ dàng đưa được về dạng x = g(x)
* Chưa có một phương pháp cụ thể nào để đưa f(x) = 0 về dạng x = g(x)

1. **Tóm tắt phương pháp**

* Đầu tiên từ phương trình: f(x) = 0 => ta phải chuyển được về dạng x = g(x); vấn đề là không phải phương trình f(x) = 0 nào cũng đưa được về dạng x = g(x)

1. Điều kiện thỏa mãn phương pháp:

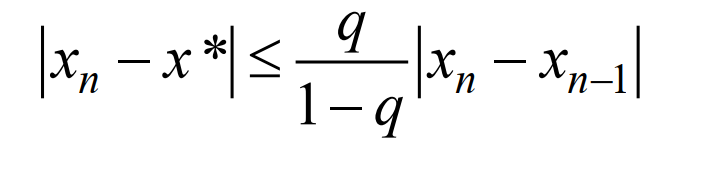
* Hàm số f(x) liên tục trên (a, b) => x = g(x)
* Hàm số g(x) liên tục khả vi trên đoạn [a, b]
* Khoảng cách li nghiệm (a, b)
* Điều kiện hội tụ: |g’(x)| <= q < 1 với mọi giá trị x thuộc [a, b]

1. Dãy lặp của phương pháp

* Hệ số co: **q = max|g’(x)|,** với x thuộc [a, b]
* Công thức lặp:

**xn = g(xn-1)**

1. Áp dụng công thức sai số: hậu nghiệm hoặc tiên nghiệm
2. Hậu nghiệm:



1. Tiên nghiệm: 