TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC

BÁO CÁO GIẢI TÍCH SỐ

ĐỀ TÀI: PHƯƠNG PHÁP LẶP ĐƠN GIẢI PHƯƠNG TRÌNH F(X)=0

Giảng viên hướng dẫn: Hà Thị Ngọc Yến

Nhóm sinh viên**:**

1. Đặng Trần Tiến – MSSV: 20195927
2. Vũ Đinh Trường An – MSSV: 20195837

Tháng 10 năm 2020

MỤC LỤC

LỜI NÓI ĐẦU

Trong phạm vi môn Giải tích số, sau khi tìm hiểu về bài toán giải phương trình f(x)=0 bằng phương pháp lặp đơn, chúng em xin phép được trình bày lại nội dụng của phương pháp lặp đơn trong bài báo cáo này. Nội dung chính của bài báo cáo bao gồm: Lý thuyết cơ bản, Thuật toán, Chương trình và hệ thống ví dụ và Ưu điểm và nhược điểm của phương pháp.

Do khả năng còn hạn chế nên bản báo cáo không thể tránh khỏi những thiếu sót về mặt nội dung và hình thức nên rất mong nhận được sự đóng góp của cô và các bạn để bản báo cáo có thể hoàn thiện hơn.

Chúng em xin gửi lời cảm ơn chân thành đến cô Hà Thị Ngọc Yến đã có những hướng dẫn, góp ý rất chi tiết để chúng em có thể hoàn thành bài báo cáo này.

I. GIỚI THIỆU PHƯƠNG PHÁP

Trong kỹ thuật, ta thường gặp bài toán: Tìm nghiệm thực của phương trình f(x)=0. Đối với các phương trình đa thức bậc lớn hơn 3 hay các loại phương trình phức tạp khác thì hầu như sẽ gặp khó khăn trong việc đi tìm nghiệm đúng của phương trình. Hơn nữa, khi làm việc với số ở dạng số thập phân, tìm được nghiệm đúng của phương trình trong dạng số thập phân vô hạn ta phải quy tròn số thì nghiệm thu được sẽ là nghiệm gần đúng. Như vậy, để tìm được nghiệm thực gần đúng của phương trình ta phải tiến hành theo các bước sau:

1) Chọn xấp xỉ đầu x_0
2) Từ x_0, tìm thuật toán để sửa dần nghiệm, được các xấp xỉ mới gần với nghiệm hơn
3) Đánh giá sai số của nghiệm gần đúng tìm được

Và một trong những thuật toán sửa dần nghiệm để thu được nghiệm gần đúng là *phương pháp lặp đơn* (lặp cổ điển).

II. LÝ THUYẾT

1. Bài toán

Giải phương trình f(x) = 0 có khoảng cách ly nghiệm α cho trước (a,b)

2. Nhắc lại về khoảng cách ly nghiệm

Khoảng (a,b) được gọi là khoảng cách ly nghiệm của phương trình f(x) = 0 nếu trong khoảng (a,b) có đúng một nghiệm của phương trình.

Định lý: Nếu f(x) liên tục và đơn điệu trên (a,b) và f(a).f(b) < 0 thì (a,b) là khoảng cách ly nghiệm của phương trình f(x) = 0.

Phương pháp tìm khoảng cách ly nghiệm:

- Phương pháp khảo sát hàm số (lập bảng biến thiên)
- Phương pháp vẽ đồ thị hàm số

3. Phương pháp lặp đơn

a. Ý tưởng bài toán

Viết lại phương trình $f(x) = 0$ trong dạng sau:

$$x = g(x)$$

$g(x)$ là hàm số liên tục trên [a,b]

b. Xây dựng công thức

Chọn x_0 bất kỳ thuộc [a,b] làm xấp xỉ đầu và tính $x_1 = g(x_0), x_2 = g(x_1), x_3 = g(x_2)\ldots$

Tổng quát: $x_n = g(x_{n-1})$ với n = 1, 2, 3… (1)

Đây chính là quá trình lặp, n là thứ của phép lặp.

c. Sự hội tụ của phương pháp

Định nghĩa sự hội tụ của quá trình lặp: quá trình lặp (1) gọi là hội tụ, nếu dãy $\{x_n\}$ tính theo (1) hội tụ khi $n \to \infty$.

Trong trường hợp này, dãy $\{x_n\}$ tính theo (1) hội tụ, nghĩa là $\lim_{n\to\infty} x_n = \bar{x}$.

Nhưng do $g(x)$ là hàm số liên tục trên [a,b] nên

$$\lim_{n\to\infty} x_n = \lim_{n\to\infty} g(x_{n-1}) = g[\lim_{n\to\infty} x_{n-1}]$$

Từ đó ta suy ra $\bar{x} = g(\bar{x})$.

Vậy trong trường hợp đó thì $\bar{x} = \alpha$ là nghiệm của $x = g(x)$, cũng là nghiệm của phương trình $f(x) = 0$

Điều kiện hội tụ của hội tụ của phương pháp (điều kiện co):

Định lý : Nếu (a,b) là khoảng cách ly nghiệm α của phương trình x = g(x) và g(x) là hàm số liên tục, có đạo hàm liên tục trên [a,b] thoả mãn :

$$|g'(x)| \leq q < 1 \; \forall x \in [a,b]$$

thì với x_0 bất kì ∈ [a,b], dãy $x_n = g(x_{n-1})$ hội tụ và hội tụ tới α.

Chứng minh. Do g(x) liên tục trên [a,b] và khả vi trong (a,b)

Theo định lí Lagrange, ta có :

$g(x_0) - g(\alpha) = g'(c_1)\,(x_0 - \alpha)$ với c_1 là trị trung gian nằm giữa x_0 và α

Do α là nghiệm của x = g(x) nên $\alpha = g(\alpha)$, x_1 = g(x_0) nên

$|\, x_1 - \alpha \,| = |\, g'(c_1) \,|.|\, x_0 - \alpha \,|$

$|\, x_1 - \alpha \,| \leq q \,|\, x_0 - \alpha \,|$

Tương tự, ta có:

$|\, x_2 - \alpha \,| \leq q \,|\, x_1 - \alpha \,|$

…

$|\, x_n - \alpha \,| \leq q \,|\, x_{n-1} - \alpha \,|$

Nhân vế với vế các bất đẳng thức trên

$|\, x_n - \alpha \,| \leq q^n |\, x_0 - \alpha \,|$

Do $0 < |\, g'(x) \,| \leq q < 1$ nên khi n → ∞ thì $x_n \to \alpha$ (theo nguyên lý kẹp)

Hay $\lim\limits_{n\to\infty} x_n = \alpha$ với x_0 bất kì ∈ [a,b]

d. Công thức đánh giá sai số

Trong tính toán, dãy $\mathrm{x_n = g(x_{n-1})}$ không thể lặp với số lần lặp vô hạn mà phải dừng lại ở bước lặp thứ n nào đó. Cho nên, cần phải đánh giá sai số qua hai công thức sai số sẽ được trình bày sau đây:

- *Công thức sai số thứ nhất:*

Theo định lý Lagrange và điều kiện co, ta có:

$$| x_n - \alpha | \leq q \mid x_{n-1} - \alpha \mid = q \mid (x_{n-1} - x_n) + (x_n - \alpha) \mid$$

$$\leq q \mid x_{n-1} - x_n \mid + q \mid x_n - \alpha \mid$$

$$(1-q) | x_n - \alpha | \leq q \mid x_{n-1} - x_n \mid$$

$$| x_n - \alpha | \leq \frac{q}{1-q} |x_n - x_{n-1}|$$

Để nghiệm gần đúng đến sai số là $\varepsilon > 0$, bé tùy ý thì

$$| x_n - \alpha | \leq \frac{q}{1-q} |x_n - x_{n-1}| \leq \varepsilon$$

$$| x_n - x_{n-1} | \leq \frac{1-q}{q} \varepsilon$$

Đây là công thức đánh giá sai số theo hai phép lặp liên tiếp và được gọi là *công thức sai số hậu nghiệm.*

- Công thức sai số thứ hai:

Theo định lí Lagrange, ta có

$$| g(x_{n-1}) - g(x_{n-2}) | = | g'(c) |. | x_{n-1} - x_{n-2} |$$

Trong đó c là trị trung gian nằm giữa x_{n-1} và x_{n-2}

$$| x_n - x_{n-1} | \leq q \mid x_{n-1} - x_{n-2} \mid$$

mà $| x_n - \alpha | \leq \frac{q}{1-q} |x_n - x_{n-1}|$ nên

$$| x_n - \alpha | \leq \frac{q}{1-q} |x_n - x_{n-1}| \leq \frac{q^2}{1-q} \mid x_{n-1} - x_{n-2} \mid$$

$$\leq \frac{q^3}{1-q} \mid x_{n-2} - x_{n-3} \mid$$

$$\leq \cdots \leq$$

$$\leq \frac{q^n}{1-q} \; | x_1 - x_0 |$$

Để nghiệm gần đúng đến sai số là $\varepsilon > 0$, thì

$$| x_n - \alpha | \leq \frac{q^n}{1-q} \; | x_1 - x_0 | \leq \varepsilon$$

$$| x_1 - x_0 | \leq \frac{1-q}{q^n} \varepsilon$$

Đây là công thức đánh giá sai số theo hai phần tử đầu tiên của dãy và được gọi là *công thức sai số tiên nghiệm.*

từ $| x_1 - x_0 | \leq \frac{1-q}{q^n} \varepsilon$

suy ra $q^n \leq \frac{1-q}{| x_1 - x_0 |} \varepsilon \; => n.lnq \leq ln\left|\frac{(1-q)\varepsilon}{x_1 - x_0}\right|$

hay $n \geq \left[ln\left|\frac{(1-q)\varepsilon}{x_1 - x_0}\right| / lnq\right]$ $(do\ lnq < 0)$

Đến đây ta có thể tìm được số phép lặp cần thiết là [n]+1 để đạt được sai số ε > 0 mong muốn trước khi quá trình lặp diễn ra.

e. Đánh giá phương pháp

- Ưu điểm : có thể chọn xấp xỉ đầu x_0 bất kì trong [a,b], thuật toán đơn giản
- Nhược điểm : không có phương pháp tổng quát đưa phương trình f(x) = 0 về phương trình x = $g(x)$. Phương pháp lặp đơn chỉ giải được phương trình có sẵn dạng x = $g(x)$ hoặc đưa được về dạng này, thoả mãn điều kiện co.

III. THUẬT TOÁN

1. Chương trình chính

INPUT: a, b, xo, ε, g

OUTPUT: Nghiệm gần đúng của phương trình.

Bước 1 : Nhập a, b, xo, ε, g

Bước 2 : Sử dụng gói kiểm tra đầu vào

Bước 3 : Khởi tạo hàm $|g'(x)|$

Bước 4 : Tìm q bằng Gói giá trị lớn nhất

Bước 5: Khởi tạo gói lặp đơn tiên nghiệm và gói lặp đơn hậu nghiệm

Bước 6 : Kiểm tra điều kiện co, in ra kết quả và kết thúc

2. *Gói kiểm tra đầu vào*

INPUT: a, b, xo

OUTPUT: input của chương trình chính đã hợp lệ

(lưu ý: a là cận dưới, b là cận trên)

Bước 1 : Nhận a, b, xo

Bước 2 : Nếu a ≥ b thì yêu cầu nhập lại

Bước 3 : Nếu xo < a hoặc xo > b thì yêu cầu nhập lại

Bước 4 : Kết thúc gói kiểm tra

3. Khởi tạo hàm $|g'(x)|$

INPUT: g(x)

OUTPUT: $|g'(x)|$

Bước 1 : Nhận g

Bước 2 : Khởi tạo số $delta > 0$ bé tuỳ ý cho trước

Bước 3 : $dy = g(x + delta) - g(x - delta),\ dx = 2 * delta$

Bước 4 : $|g'(x)| = |\ dy\ /\ dx\ |$ và kết thúc

Chú ý : Hàm $|g'(x)|$ trong chương trình có tên là abs_g_phay

$|g'(x)| = |\ dy\ /\ dx\ |$ là do :

$$2g'(x) = \lim_{\Delta x \to 0} \frac{g(x+\Delta x) - g(x)}{\Delta x} + \lim_{\Delta x \to 0} \frac{g(x) - g(x-\Delta x)}{\Delta x}$$

$nên$ $$g'(x) = \lim_{\Delta x \to 0} \frac{g(x+\Delta x) - g(x-\Delta x)}{2\Delta x}$$

4. Gói giá trị lớn nhất

INPUT: a, b, f

OUTPUT: max f trên [a,b]

Bước 1 : Nhập a, b, f

Bước 2 : Khởi tạo bước nhảy $eps > 0$ bé tuỳ ý cho trước và max = f(a)

Bước 3 : $i = a + eps$, nếu i > b thì sang bước 5

Bước 4 : Nếu $f(i)$ > max thì max = f(i) quay lại bước 3

Bước 5 : Đưa ra kết quả là max và kết thúc

Chú ý : khi nhập f = $|g'(x)|$ ta sẽ có q = max $|g'(x)|$

5. Gói lặp đơn tiên nghiệm

INPUT: xo, q, ε, g

OUTPUT: nghiệm gần đúng của phương trình

Bước 1 : Nhận xo, q, ε, g

Bước 2 : Tính x = g(xo)

Bước 3 : Nếu x = xo thì nghiệm cần tìm là x = xo

Bước 4 : n = ln$\left|\frac{(1-q)\varepsilon}{x - xo}\right|$ / lnq ,sau đó n = [n] + 1

Bước 5 : for i = 2 to n do x = g(x)

Bước 6 : Đưa ra kết quả nghiệm cần tìm là x và kết thúc

6. Gói lặp đơn hậu nghiệm

INPUT: xo, q, ε, g

OUTPUT: nghiệm gần đúng của phương trình

Bước 1 : Nhận xo, q, ε, g

Bước 2 : Tính x = g(xo)

Bước 3 : Nếu x = xo thì nghiệm cần tìm là x = xo

Bước 4 : Kiểm tra $|\ x - \ x_0\ | \ < \ \frac{1-q}{q}\ \varepsilon$. Nếu thoả mãn, dừng thuật toán, nghiệm cần tìm là x và kết thúc

Bước 5 : xo = x, x = g(x), quay lại bước 4

7. Gói điều kiện co

INPUT: q

OUTPUT: nghiệm gần đúng (theo cả hai công thức sai số)

Bước 1 : Nhận q

Bước 2 : Nếu 0 < q <1 thì sử dụng cả hai gói lặp đơn tiên nghiệm và hậu nghiệm

Bước 3 : Nếu q <=0 hoặc q>=1 thì in ra hàm g(x) không thoả mãn điều kiện co

Bước 4 : Kết thúc chương trình

IV. CHƯƠNG TRÌNH VÀ HỆ THỐNG VÍ DỤ

1. Ví dụ 1: Giải gần đúng phương trình x= $\sqrt[3]{3x+2}$ với k.c.l $(1.5,2.5)$

```
double g(double x) // nhap ham g(x)
{
    double mu=(double) 1/3;
    double t=3*x+2;
    if (t<0) return -exp((log(-t)*mu));
    else
        return exp((log(t)*mu));
```

INPUT:

```
}
```

(do hạn chế về kĩ thuật lập trình nên chúng em sẽ nhập hàm trực tiếp trên IDE)

```
Nhap can duoi a=1.5
Nhap can tren b=2.5
Nhap xap xi dau Xo=1.5
Nhap sai so epsilon=1.0e-9
```

OUTPUT:

```
Theo cong thuc sai so tien nghiem----
So lan lap : 17
Nghiem gan dung X= 1.999999999968143
Theo cong thuc sai so hau nghiem-----
So lan lap : 15
Nghiem gan dung X= 1.999999999490298
--------------------------------
```

Phương trình x= $\sqrt[3]{3x+2}$ trong khoảng cách ly $(1.5, 2.5)$ có nghiệm x=2

Ví dụ cho thấy chương trình có chạy qua kết quả đúng đến sai số epsilon = 1.0e-9 nhập từ bàn phím, chứng minh code hoạt động.

2 .Ví dụ 2: giải gần đúng phương trình x= $\sqrt[3]{3x+2}$ với k.c.l $(-1.5, -0.5)$

INPUT:

```
double g(double x) // nhap ham g(x)
{
    double mu=(double) 1/3;
    double t=3*x+2;
    if (t<0) return -exp((log(-t)*mu));
    else
        return exp((log(t)*mu));
}
```

```
Nhap can duoi a=-1.5
Nhap can tren b=-0.5
Nhap xap xi dau Xo=-1.5
Nhap sai so epsilon=1.0e-9
```

OUTPUT:

```
Ham g(x) khong thoa man dieu kien anh xa co, vui long dieu chinh lai [a,b]
```

Đây là trường hợp g(x) không thoả mãn điều kiện ánh xạ co, nếu vẫn để quá trình lặp diễn ra sẽ bị rơi vào trạng thái lặp vô hạn:

```
X45971= -1.000021755021850
X45972= -1.000021754548586
X45973= -1.000021754075343
X45974= -1.000021753602121
X45975= -1.000021753128918
X45976= -1.000021752655737
X45977= -1.000021752182576
X45978= -1.000021751709436
X45979= -1.000021751236316
X45980= -1.000021750763217
X45981= -1.000021750290138
X45982= -1.000021749817080
X45983= -1.000021749344043
X45984= -1.000021748871026
X45985= -1.000021748398030
X45986= -1.000021747925054
X45987= -1.000021747452099
X45988= -1.000021746979165
X45989= -1.000021746506251
X45990= -1.000021746033357
X45991= -1.000021745560484
X45992= -1.000021745087632
X45993= -1.000021744614801
X45994= -1.000021744141989
X45995= -1.000021743669199
X45996= -1.000021743196429
X45997= -1.000021742723679
X45998= -1.000021742250950
X45999= -1.000021741778242
X46000= -1.000021741305554
X46001= -1.000021740832887
X46002= -1.000021740360240
X46003= -1.000021739887614
X46004= -1.000021739415009
X46005= -1.000021738942424
X46006= -1.000021738469859
X46007= -1.000021737997315
X46008= -1.000021737524792
X46009=
X46005= -1.000021738942424
X46006= -1.000021738469859
X46007= -1.000021737997315
X46008= -1.000021737524792
X46009=
```

```
X88586= -1.000011289107612
X88587= -1.000011288980170
X88588= -1.000011288852732
X88589= -1.000011288725296
X88590= -1.000011288597863
X88591= -1.000011288470433
X88592= -1.000011288343006
X88593= -1.000011288215582
X88594= -1.000011288088160
X88595= -1.000011287960742
X88596= -1.000011287833326
X88597= -1.000011287705914
X88598= -1.000011287578504
X88599= -1.000011287451096
X88600= -1.000011287323692
X88601= -1.000011287196291
X88602= -1.000011287068893
X88603= -1.000011286941497
X88604= -1.000011286814104
X88605= -1.000011286686715
X88606= -1.000011286559328
X88607= -1.000011286431943
X88608= -1.000011286304562
X88609= -1.000011286177184
X88610= -1.000011286049809
X88611= -1.000011285922436
X88612= -1.000011285795067
X88613= -1.000011285667700
X88614= -1.000011285540336
X88615= -1.000011285412975
X88616= -1.000011285285617
X88617= -1.000011285158261
X88618= -1.000011285030909
X88619= -1.000011284903560
X88620= -1.000011284776213
X88621= -1.000011284648869
X88622= -1.000011284521528
X88623= -1.000011284394190
X88624= -1.000011284266855
X88625= -1.000011284139523
X88626= -1.000011284012193
X88627= -1.000011283884867
X88628= -1.000011283757543
X886
```

Dựa vào vòng lặp vô hạn trên ta vẫn có thể dự đoán được nghiệm của phương trình x= $\sqrt[3]{3x+2}$ trong khoảng cách ly $(-1.5, -0.5)$ là x= -1 khi mà nghiệm có xu hướng hội tụ về -1 .Tất nhiên, dự báo này chỉ có thể làm được trong một số trường hợp nhất định ví dụ như trong trường hợp này (không phải mọi trường hợp điều dự báo được khi quá trình lặp rơi vào trạng thái vô hạn.

3. Ví dụ 3: giải gần đúng phương trình x= $\sqrt[3]{3x+2}$ với k.c.l (1.5,2.5)

Quay trở lại với ví dụ 1 ở trên để so sánh độ chính xác của công thức sai số

```
double g(double x) // nhap ham g(x)
{
    double mu=(double) 1/3;
    double t=3*x+2;
    if (t<0) return -exp((log(-t)*mu));
    else
        return exp((log(t)*mu));
}
```

INPUT:

```
Nhap can duoi a=1.5
Nhap can tren b=2.5
Nhap xap xi dau Xo=1.5
Nhap sai so epsilon=1.0e-9
```

OUTPUT:

```
Theo cong thuc sai so tien nghiem----
X1= 1.866255578408624
X2= 1.965988792042286
X3= 1.991460790852711
X4= 1.997862914960307
X5= 1.999465585953609
X6= 1.999866387562459
X7= 1.999966596332715
X8= 1.999991649048310
X9= 1.999997912259898
X10= 1.999999478064838
X11= 1.999999869516201
X12= 1.999999967379050
X13= 1.999999991844762
X14= 1.999999997961190
X15= 1.999999999490298
X16= 1.999999999872574
X17= 1.999999999968143
So lan lap : 17
Nghiem gan dung X= 1.999999999968143
```

```
Theo cong thuc sai so hau nghiem----
X1= 1.866255578408624
X2= 1.965988792042286
X3= 1.991460790852711
X4= 1.997862914960307
X5= 1.999465585953609
X6= 1.999866387562459
X7= 1.999966596332715
X8= 1.999991649048310
X9= 1.999997912259898
X10= 1.999999478064838
X11= 1.999999869516201
X12= 1.999999967379050
X13= 1.999999991844762
X14= 1.999999997961190
X15= 1.999999999490298
So lan lap : 15
Nghiem gan dung X= 1.999999999490298
```

Trong quá trình test độ chính xác của 2 công thức lặp đơn tiên nghiệm và hậu nghiệm, chúng em nhận thấy không có công thức nào luôn cho độ chính xác luôn cao hơn (lúc có công thức tiên nghiệm chính xác hơn, nhưng có lúc công thức hậu nghiệm lại chính xác, có lúc độ chính xác của 2 công thức là tương đối nhau). Điều này có thể đến từ thuật toán tính đạo hàm $g'(x)$, thuật toán tìm $q = \max|g'(x)|$ còn chưa được tối ưu khi vẫn có hạn chế ở bước chọn $delta, eps$ bé tuỳ ý khiến sai số trong quá trình tính toán còn khá lớn; và cũng có thể do kết quả phụ thuộc vào xấp xỉ đầu x_0 mà ta chọn.

TÀI LIỆU THAM KHẢO

Lê Trọng Vinh (2007). *Giáo trình Giải tích số*, Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.