TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC

-------o0o-------

BÁO CÁO MÔN HỌC

BỘ MÔN: GIẢI TÍCH SỐ

ĐỀ TÀI:

GIẢI PHƯƠNG TRÌNH ĐA THỨC

DANH SÁCH SINH VIÊN NHÓM 6:

1. Nguyễn Quốc Trung 20195937 CTTN Toán Tin K64
2. Nguyễn Tạ Quốc Cường 20195846 CTTN Toán Tin K64

Ta xét phương trình dạng đa thức

$$f(x) = p(x) = a_0x^n + a_1x^{n-1} + \dots + a_{n-1}x + a_n = 0 \qquad (1)$$

trong đó $a_0 \neq 0$. Để tìm nghiệm thực gần đúng của phương trình (1), ta hoàn toàn có thể áp dụng các phương pháp đã học như lặp đơn, tiếp tuyến, dây cung, ... Nhưng do (1) có dạng đa thức nên cũng có những đặc trưng riêng, có thể giải theo phương pháp khác.

I. SƠ ĐỒ HOOCNE

Tính giá trị của đa thức

Tại $x = c$; nếu $f(c) = p(c) = 0$ thì hiển c là nghiệm của phương trình (1). Nói chung với c bất kỳ thì $p(c) \neq 0$. Ta sẽ tìm $p(c)$

Do $p(x) = a_0x^n + a_1x^{n-1} + \dots + a_{n-1}x + a_n = (a_0x + a_1)x^{n-1} + a_2x^{n-2} + \dots$

$$= ((a_0x + a_1)x + a_2)x^{n-2} + \dots = (\dots((a_0x + a_1)x + a_2)x \dots + a_n)$$

Khi $x = c$ ta tính dần:

$a_0 = b_0$

$a_1 + b_0c = b_1$

$a_2 + b_1c = b_2$

...

$a_n + b_{n-1}c = b_n$ thì $b_n = p(c)$

Quá trình này được mô tả bằng bảng sau. Gọi là sơ đồ Hoocne

Hệ số đa thức	a_0	a_1	a_2	......	a_{n-1}	a_n
		cb_0	cb_1	......	cb_{n-2}	cb_{n-1}
c	b_0	b_1	b_2	......	b_{n-1}	$b_n = p(c)$

Ta thấy các số $b_0, b_1, b_2, \dots, b_{n-1}$ là hệ số của đa thức thương Q(x) khi chia đa thức p(x) cho đơn thức $x - c$.

Thực vậy. Ta đặt:

$$Q(x) = \beta_0x^{n-1} + \beta_1x^{n-2} + \dots + \beta_{n-2}x + \beta_{n-1} \qquad (2)$$

Thì $$p(x) = Q(x)(x - c) + \beta_n \qquad (3)$$

Khai triển (3) ta được:

$$p(x) = \beta_0x^n + (\beta_1 - \beta_0c)x^{n-1} + \dots + (\beta_{n-1} - \beta_{n-2}c)x + (\beta_n - \beta_{n-1}c)$$

Đồng nhất theo lũy thừa của x với phương trình (1)

$\beta_0 = a_0$

$\beta_1 - \beta_0 c = a_1$

...

$\beta_n - \beta_{n-1} c = a_n$

....

Hay $a_0 = \beta_0$

$a_1 + \beta_0 c = \beta_1$

...

$a_n + \beta_{n-1} c = \beta_n$

So sánh với hệ có được khi thay $x = c$ ta thấy:

$b_0 = \beta_0, \qquad b_1 = \beta_1, \qquad ..., \qquad b_n = \beta_n = p(c)$

Hay nói cách khác

$$p(x) = (x - c)[b_0 x^{n-1} + b_1 x^{n-2} + ... + b_{n-1}] + p(c)$$

Như vậy sơ đồ Hoocne có thể tính được giá trị của đa thức p(x) tại điểm x=c, đồng thời cũng cho ta đa thức thương của đa thức p(x) chia cho x-c

Ví dụ: Cho $p(x) = x^4 - 4x^3 - 7x^2 + 22x + 24$

Tính $p(5)$ và tìm đa thức $p_1(x)$ sao cho:

$$p(x) = (x - 5)p_1(x) + p(5)$$

Lập sơ đồ Hoocne ta được:

p(x)	1	-4	-7	22	24
		5	5	-10	60
5	1	1	-2	12	84 = p(5)

Vậy $p(5) = 84$ và $p_1(x) = x^3 + x^2 - 2x + 12$

II. MIỀN CHỨA NGHIỆM CỦA ĐA THỨC

Định lý 1: Xét phương trình

$$p(x) = a_0 x^n + a_1 x^{n-1} + ... + a_{n-1} x + a_n = 0 \qquad (4)$$

$(a_0 \neq 0, a_i \in R$ với $i = \overline{1, n})$

Đặt $A = \max\{|a_i|, i = \overline{1,n}\}$

thì các nghiệm (thực hoặc phức) của phương trình (4) thỏa mãn

$$|x| < 1 + \frac{A}{|a_0|} \tag{5}$$

Nghĩa là các nghiệm của phương trình (4) nằm trong mặt tròn tâm O bán kính $R = 1 + \frac{A}{|a_0|}$ ở trong mặt phẳng phức.

Chứng minh định lý: Xét các số x không thỏa mãn (5), tức là

$$|x| \geq 1 + \frac{A}{|a_0|} \tag{6}$$

Từ đó suy ra $|a_0| \geq \frac{A}{|x|-1}$ hay $|a_0 x^n| \geq \frac{A|x^n|}{|x|-1}$

Mặt khác

$$|a_1x^{n-1} + a_2x^{n-2} + \dots + a_n| \leq |a_1||x^{n-1}| + \dots + |a_n| \leq A\{|x|^{n-1} + |x|^{n-2} + \dots + 1\}$$

$$= A\frac{|x|^n - 1}{|x|-1}$$

Do (6) thì $|x| > 1$ nên $\frac{|x|^n-1}{|x|-1} < \frac{|x|^n}{|x|-1}$

$$\rightarrow |a_1x^{n-1} + a_2x^{n-2} + \cdots + a_n| < A\frac{|x|^n}{|x|-1} \leq |a_0x^n|$$

Ta thấy số hạng đầu trội hơn hẳn các số hạng sau của đa thức. Do đó x thỏa mãn (6) thì không thể là nghiệm của phương trình (4). Định lý được chứng minh.

Từ đây ta có hệ quả:

Số $N = 1 + \frac{A}{|a_0|}$ là cận trên của modun các nghiệm của phương trình (4).

Bây giờ ta xét trường hợp nghiệm thực của phương trình (4). Ta đã biết các nghiệm âm của phương trình $p(x) = 0$ cũng là nghiệm dương của phương trình $p(-x) = 0$. Vì vậy ta chỉ cần đề cập đến cận trên của các nghiệm dương.

Định lý 2: Nếu $a_0 > 0, a_i > 0$ với $i = \overline{1,n}$ thì $p(x) > 0\ \forall x > 0$; phương trình (4) không có nghiệm dương.

Giả sử $a_0 > 0, a_k (k \geq 1)$ là hệ số âm đầu tiên tính từ $k = 1,2, \dots$ gọi B = max của các trị tuyệt đối của các hệ số âm. Thì các nghiệm dương của phương trình (4) thỏa mãn

$$x < 1 + \sqrt[k]{\frac{B}{a_0}}$$

Cách chứng minh hoàn toàn tương tự chứng minh định lý 1.

<u>VD:</u> Cho phương trình

$$p(x) = x^4 - 5x^3 - 3x^2 - 2x + 1 \qquad (*)$$

1. Tìm miền chứa nghiệm (thực và phức)

2. Tìm cận trên, cận dưới miền chứa nghiệm thực

Giải:

1. Miền chứa nghiệm: $a_0 = 1, A = \max\{1,5,3,2\} = 5$

Vậy $N = 1 + \frac{A}{a_0} = 1 + \frac{5}{1} = 6$

→ Miền chứa nghiệm là miền tròn tâm O bán kính R = 6 nằm trong mặt phẳng phức.

2. Tìm cận trên, dưới của nghiệm thực.

Cận trên: hệ số âm đầu tiên là $a_1 = -5$ do đó $k = 1$, còn $B = \max\{|-5|, |-3|, |-2|\} = 5$

→ Cận trên của nghiệm thực dương là

$$x < M = 1 + \sqrt[1]{\frac{5}{1}} = 6$$

Ta tìm cận dưới của các nghiệm thực âm. Từ phương trình (*) ta thay x bởi –x:

$$p(-x) = x^4 + 5x^3 - 3x^2 + 2x + 1 = 0$$

Hệ số âm đầu tiên $b_2 = -3$ → $k = 2$ và $B = |-3| = 3$

$$\rightarrow -x < \overline{M} = 1 + \sqrt{\frac{3}{1}} = 2.733 \text{ hay } x > -\overline{M} = -2.733$$

Vậy nghiệm thực của phương trình (*) nằm trong khoảng $-2.733 < x < 6$

<u>*Nhược điểm:</u> Ta không thể phân biệt khi nào phương trình vô nghiệm.

VD: Phương trình $x^2 - 2x + 5 = 0$ không có nghiệm thực. Tuy nhiên khi áp dụng định lý trên, ta thấy $k = 1, B = 2$ → Cận trên của nghiệm thực dương là: $M = 1 + \sqrt[1]{\frac{2}{1}} = 3$.

Hoặc đối với phương trình $x^2 + 2x + 5 = 0$ không tồn tại hệ số âm, trường hợp này khi lập trình ta mặc định cận trên của nghiệm thực dương là 0. (Phương trình không có nghiệm dương), tương tự đối với nghiệm âm.

***Ứng dụng vào giải phương trình đa thức:**

Ý tưởng:

B1: Tìm cận trên và cận dưới miền chứa nghiệm thực của phương trình theo định lý 2

B2: Sử dụng gói min-max để tìm các điểm cực trị trong miền nghiệm vừa tìm

B3: Nếu 2 điểm liền nhau tạo thành một khoảng cách ly nghiệm, sử dụng các phương pháp đã học để giải phương trình (dây cung/ tiếp tuyến/ chia đôi)

III. PHƯƠNG PHÁP LOBACHEVSKY

Phương pháp này được áp dụng để tìm các nghiệm gần đúng của đa thức, mà không cần tìm miền chứa nghiệm cũng như không cần tìm khoảng phân ly nghiệm. Sau đây lần lượt xét các trường hợp sau.

3.1 Trường hợp chỉ có nghiệm thực phân biệt

Giả sử phương trình đa thức bậc n

$$p(x) = a_0x^n + a_1x^{n-1} + \dots + a_{n-1}x + a_n = 0 \tag{7}$$

$(a_0 \neq 0, a_i \in R, i = \overline{1,n})$ chỉ có các nghiệm thực x_i với trị tuyệt đối khác nhau.

Ta đánh số các nghiệm đó thứ tự giảm dần theo trị tuyệt đối:

$$|x_1| > |x_2| > \dots > |x_n| \tag{8}$$

Dựa vào (7) ta xây dựng phương trình mới

$$Q(x) = c_0x^n + c_1x^{n-1} + \dots + c_{n-1}x + c_n = 0 \ (c_0 \neq 0) \tag{9}$$

Có nghiệm là $-x_1^m, -x_2^m, \dots, -x_n^m$. Hay:

$$Q(x) = c_0(x + x_1^m)(x + x_2^m) \dots (x + x_n^m) = 0 \tag{10}$$

So sánh (9) với (10) ta có

$$x_1^m + x_2^m + \dots + x_n^m = \frac{c_1}{c_0}$$

$$x_1^m x_2^m + x_2^m x_3^m + \dots + x_{n-1}^m x_n^m = \frac{c_2}{c_0}$$

...

$$x_1^m x_2^m \dots x_n^m = \frac{c_n}{c_0}$$

(công thức Viete) hay

$$x_1^m \left[1 + \left(\frac{x_2}{x_1}\right)^m + \dots + \left(\frac{x_n}{x_1}\right)^m\right] = \frac{c_1}{c_0}$$

$$x_1^m x_2^m \left[1 + \left(\frac{x_2 x_3}{x_1 x_2}\right)^m + \dots + \left(\frac{x_{n-1} x_n}{x_1 x_2}\right)^m\right] = \frac{c_2}{c_0}$$

...

$$x_1^m x_2^m \dots x_n^m = \frac{c_n}{c_0}$$

Do giả thiết (8) nên khi m tăng lên các phần trong ngoặc vuông của hệ trên → 1

Vậy với m đủ lớn thì

$$x_1^m \approx \frac{c_1}{c_0}$$

$$x_1^m x_2^m \approx \frac{c_2}{c_0}$$

...

$$x_1^m x_2^m \dots x_n^m = \frac{c_n}{c_0}$$

Từ đó → $|x_1| \approx \sqrt[m]{\left|\frac{c_1}{c_0}\right|}, |x_2| \approx \sqrt[m]{\left|\frac{c_2}{c_1}\right|}, \dots, |x_n| \approx \sqrt[m]{\left|\frac{c_n}{c_{n-1}}\right|}$

Để được nghiệm thực gần đúng ta lần lượt thay $\pm|x_i|$, với $i = \overline{1, n}$ vào (7) để xác định dấu của nghiệm.

Vậy vấn đề hiện tại là ta cần xây dựng đa thức dạng (9), ta tiến hành từng bước như sau:

Từ (7), ta xây dựng phương trình mới có nghiệm là $-x_i^2$ $\left(i = \overline{1, n}\right)$ (gọi là quá trình bình phương nghiệm). Áp dụng liên tiếp quá trình bình phương nghiệm k – 1 lần nữa ta được phương trình mà nghiệm là $-x_i^{2^k}$ tức là $-x_i^m$ (với $m = 2^k$)

Trước hết ta xét kỹ bước đầu tiên

Do $p(x) = 0$ có nghiệm là x_i nên:

$$p(x) = a_0(x - x_1)(x - x_2) \dots (x - x_n)$$

Thay x bởi –x

$(-1)^n p(-x) = a_0(x + x_1)(x + x_2) \dots (x + x_n)$ suy ra

$$(-1)^n p(-x) p(x) = a_0^2(x^2 - x_1^2)(x^2 - x_2^2) \dots (x^2 - x_n^2)$$

Thay x^2 bởi –y

$$p(x)p(-x) = a_0^2(y + x_1^2)(y + x_2^2) \dots (y + x_n^2) = p_1(y)$$

Vậy đa thức $p_1(y)$ có nghiệm $y_i = -x_i^2$. Ta khai triển $p_1(y)$

$$p_1(y) = a_0^{(1)} y^n + a_1^{(1)} y^{n-1} + \dots + a_n^{(1)} \qquad (11)$$

Bây giờ ta tính cụ thể các $a_i^{(1)}$

Do $\quad p(x) = a_0x^n + a_1x^{n-1} + \dots + a_{n-1}x + a_n$

Nên $\quad p(-x) = (-1)^n[a_0x^n - a_1x^{n-1} + \dots + (-1)^n a_n]$

Hay $\quad (-1)^n p(-x) = a_0x^n - a_1x^{n-1} + \dots + (-1)^n a_n$

Từ đó ta có

$$(-1)^n p(x)p(-x) = a_0^2x^{2n} - (a_1^2 - 2a_0a_2)x^{2n-2} + (a_2^2 - 2a_1a_3 + 2a_0a_4)x^{2n-4}$$
$$+ \dots + (-1)^n a_n^2$$

Thay x^2 bởi (-y), thay $p(x)p(-x) = p_1(y)$ ta được

$$p_1(y) = a_0^2y^n + (a_1^2 - 2a_0a_2)y^{n-1} + (a_2^2 - 2a_1a_3 + 2a_0a_4)y^{n-2} + \dots + a_n^2 \qquad (12)$$

So sánh (11) và (12) ta thấy

$$\left.\begin{aligned} a_0^{(1)} &= a_0^2 \\ a_1^{(1)} &= (a_1^2 - 2a_0a_2) \\ a_2^{(1)} &= (a_2^2 - 2a_1a_3 + 2a_0a_4) \\ &\dots \\ a_n^{(1)} &= a_n^2 \end{aligned}\right\} \qquad (12.1)$$

Vậy ta được đa thức (11). Thay y bởi x ta có $p_1(x)$

Tiếp tục quá trình bình phương nghiệm. Sau khi thực hiện k lần ta thu được

$$p_k(z) = a_0^{(k)}z^n + a_1^{(k)}z^{n-1} + \dots + a_n^{(k)} = 0$$

Có nghiệm là $z_i = -x_i^{2^k}$ nghĩa là $z_i = -x_i^m$ (với $m = 2^k$)

Khi k đủ lớn ta được dạng (9) với $c_i = a_i^{(k)}$

Và ta có: $\quad x_i^{2^k} \approx \dfrac{a_i^{(k)}}{a_{i-1}^{(k)}} \qquad i = \overline{1, n} \qquad (13)$

Vậy k thế nào là đủ lớn? Giả sử đến bước thứ k ta có đa thức (9), ta tính tiếp bước k+1:

$$x_i^{2^{k+1}} \approx \frac{a_i^{(k+1)}}{a_{i-1}^{(k+1)}} \qquad i = \overline{1, n} \qquad (14)$$

Ta so sánh (13) và (14). Do $a_0^{(k+1)} = \left[a_0^{(k)}\right]^2$ ta được

$$a_i^{(k+1)} \approx \left[a_i^{(k)}\right]^2 \qquad i = \overline{1, n} \qquad (15)$$

Logarit hai vế

$$\lg a_i^{(k+1)} \approx 2 \lg a_i^{(k)} \qquad i = \overline{1, n} \tag{16}$$

Nghĩa là logarit của hệ số bước sau gấp 2 lần logarit của hệ số bước trước tương ứng. Từ (15) và (16) cho thấy phương trình (7) chỉ có các nghiệm thực đơn và nghiệm gần đúng được tìm từ (14).

<u>Ví dụ:</u> Giải phương trình $p(x) = x^3 - x^2 - 22x + 40 = 0$

Áp dụng quá trình bình phương nghiệm và công thức (I) ta được kết quả như bảng sau:

k	$a_0^{(k)}$	$a_1^{(k)}$	$a_2^{(k)}$	$a_3^{(k)}$
0	1	-1	-22	40
1	1	45	564	1600
2	1	897	$1{,}74096.10^5$	$2{,}56.10^6$
3	1	$4{,}56417.10^5$	$2{,}57168.\,10^{10}$	$6{,}5536.\,10^{12}$
4	1	$1{,}56883.\,10^{11}$	$6{,}55371.\,10^{20}$	$4{,}29497.\,10^{25}$
5	1	$2{,}33015.\,10^{22}$	$4{,}29498.\,10^{41}$	$1{,}84468.\,10^{51}$
6	1	$5{,}42101.\,10^{44}$	$1{,}84469.\,10^{83}$	$3{,}40284.\,10^{102}$

Qua bảng trên ta thấy: $a_i^{(6)} \approx \left[\, a_i^{(5)} \right]^2$ nên ta dừng ở bước k = 6 → $m = 2^6 = 64$

Vậy: $|x_1| \approx [5{,}42101.\,10^{44}]^{\frac{1}{64}} \approx 5{,}00000$

$$|x_2| \approx \left[\frac{1{,}84469.10^{83}}{5{,}42101.10^{44}}\right]^{\frac{1}{64}} \approx 4{,}00000$$

$$|x_2| \approx \left[\frac{3{,}40284.10^{102}}{1{,}84469.10^{83}}\right]^{\frac{1}{64}} \approx 2{,}00000$$

Thay $\pm 5, \pm 4, \pm 2$ vào phương trình; ta được $x_1 = -5$; $x_2 = 4$; $x_3 = 2$ là nghiệm của phương trình đã cho.

3.2 Trường hợp phương trình chỉ có nghiệm thực, nhưng có nghiệm bội hay có nghiệm khá gần nhau về trị tuyệt đối

Xét phương trình (7) có nghiệm thực x_i trong đó:

$x_2 = x_3$ hay $|x_2| \approx |x_3|$ nghĩa là:

$$|x_1| > |x_2| \approx |x_3| > |x_4| \ldots > |x_n|$$

Ta vẫn áp dụng tương tự quá trình bình phương nghiệm như phần 3.1, với k đủ lớn

$$x_1^{2^k} \approx \frac{a_1^{(k)}}{a_0^{(k)}}$$

$$(x_2^2)^{2^k} \approx (x_3^2)^{2^k} \approx \frac{a_3^{(k)}}{a_1^{(k)}} \approx \frac{1}{2}\left[\frac{a_2^{(k)}}{a_1^{(k)}}\right]^2$$

$$x_4^{2^k} \approx \frac{a_4^{(k)}}{a_3^{(k)}}$$

...

$$x_n^{2^k} \approx \frac{a_n^{(k)}}{a_{n-1}^{(k)}}$$

Và quá trình này sẽ dừng khi

$$a_1^{(k+1)} \approx \left[a_1^{(k)}\right]^2$$

$$a_2^{(k+1)} \approx \frac{1}{2}\left[a_2^{(k)}\right]^2$$

$$a_3^{(k+1)} \approx \left[a_3^{(k)}\right]^2$$

...

$$a_n^{(k+1)} \approx \left[a_n^{(k)}\right]^2$$

Trong trường hợp phương trình (7) có s nghiệm thực bằng nhau hoặc có trị tuyệt đối xấp xỉ nhau là $x_i, x_{i+1}, \ldots, x_{i+s-1}$ thì ta thay điều kiện thứ 2 của hệ trên bởi điều kiện:

$$a_i^{(k+1)} \approx \frac{1}{s}\left[a_i^{(k)}\right]^2$$

3.3 Trường hợp phương trình (7) có nghiệm phức

Ta giả sử phương trình có cặp nghiệm phức là x_2 và x_3:

$$x_2 = r(cos\varphi + isin\varphi)$$

$$x_3 = r(cos\varphi - isin\varphi)$$

Và các nghiệm thỏa mãn điều kiện

$$|x_1| > |x_2| = |x_3| > |x_4| > \ldots > |x_n|$$

Mặt khác ta có:

$x_2 + x_3 = 2rcos\varphi$

$x_2 x_3 = r^2$

Hoàn toàn tương tự như trên, bằng phương pháp bình phương nghiệm liên tiếp và với k đủ lớn:

$$x_1^{2^k} \approx \frac{a_1^{(k)}}{a_0^{(k)}}$$

$$(r^2)^{2^k} \approx \frac{a_3^{(k)}}{a_1^{(k)}}$$

$$x_4^{2^k} \approx \frac{a_4^{(k)}}{a_3^{(k)}}$$

...

$$x_n^{2^k} \approx \frac{a_n^{(k)}}{a_{n-1}^{(k)}}$$

Từ đó ta xác định được modun r của x_2, x_3 và trị tuyệt đối của các nghiệm thực x. Ta xét dấu của các nghiệm thực như trên, còn acgumen φ của x_2, x_3 sẽ xác định:

Do $\quad x_1 + x_2 + x_3 + \dots + x_n = -\frac{a_1}{a_0}$

Ta có $\quad x_1 + 2rcos\varphi + x_4 + \dots + x_n = -\frac{a_1}{a_0}$ từ đó ta giải được φ

Quá trình bình phương nghiệm sẽ dừng khi

$$a_i^{(k+1)} \approx \left[a_i^{(k)}\right]^2 \qquad \text{với } i \neq 2$$

Riêng với $a_2^{(k)}$, dựa vào hệ thức Viete ta có.

$$x_1^{2^k} x_2^{2^k} + x_1^{2^k} x_3^{2^k} + x_1^{2^k} x_4^{2^k} + \dots + x_{n-1}^{2^k} x_n^{2^k} = \frac{a_2^{(k)}}{a_0^{(k)}}$$

Ta suy ra $\quad x_1^{2^k}\left(x_2^{2^k} + x_3^{2^k}\right) + x_1^{2^k} x_4^{2^k} + \dots + x_{n-1}^{2^k} x_n^{2^k} = \frac{a_2^{(k)}}{a_0^{(k)}}$

Vì x_2, x_3 là 2 số phức liên hợp nên:

$x_2^{2^k} + x_3^{2^k} = 2r^{2^k}(cos2^k\varphi) \;\rightarrow\; a_2^{(k)}$ có chứa số hạng $(cos2^k\varphi)$ do đó thay đổi không theo quy luật. Đó là dấu hiệu cho thấy phương trình (7) có nghiệm phức.

Như vậy ta có thể rút ra kết luận: Từ quá trình bình phương nghiệm ta suy ra:

1) Nếu k đủ lớn để

$$a_i^{(k+1)} \approx \left[a_i^{(k)}\right]^2 \qquad\qquad i = \overline{0,n}$$

Thì phương trình (7) chỉ có nghiệm thực phân biệt.

2) Khi 1) không xảy ra, nhưng có i để

$$a_i^{(k+1)} \approx \frac{1}{s}\left[a_i^{(k)}\right]^2$$

Với k đủ lớn, ta có thể tin rằng (7) có một nghiệm thực bội là x_i hay có s nghiệm thực mà có trị tuyệt đối gần nhau. Hiện tượng này có thể xảy ra với 2, 3 hệ số, lúc đó sẽ có 2, 3 nghiệm bội...

3) Nếu có hệ số thay đổi không theo quy luật thì có thể tin rằng phương trình (7) có một cặp nghiệm phức. Nếu có tới 2, 3 hệ số thay đổi không theo quy luật thì tương ứng sẽ có 2, 3 cặp nghiệm phức liên hợp.

Tuy nhiên, những điều nói trên chỉ là nhận xét sơ bộ. Muốn khẳng định được ta cần tính toán đến cùng và kiểm tra bằng cách thay lại vào phương trình (7) để biết giá trị nào thỏa mãn.

***Ưu điểm: Có thể tìm được các nghiệm phức**

***Nhược điểm:**

- Quá trình bình phương nghiệm dẫn đến hệ số rất lớn

- Không kiểm soát được sai số của nghiệm phương trình

- Do phép đánh giá $a_i^{(k+1)} \approx \left[a_i^{(k)}\right]^2$ để dừng quá trình lặp dựa trên ý kiến chủ quan nên khi viết chương trình khó có thể dừng thuật toán một cách chính xác. (Chẳng hạn như ta muốn dừng lặp sau k bước (k cho trước), khi $\lg a_i^{(k+1)} - 2\lg a_i^{(k)}$ < epsilon (*) **(với epsilon cho trước) thì với một số phương trình, (*) có thể không chính xác.**

IV. THUẬT TOÁN

Đối với phương pháp Lobachevsky, do khả năng lập trình còn yếu kém nên nhóm em gặp phải một số vấn đề khi lập trình. Do vậy em sử dụng phương pháp tìm miền chứa nghiệm và các điểm cực trị.

INPUT: Bậc của đa thức n, các hệ số a[0], a[1], …, a[n]

OUTPUT: Nghiệm của phương trình f(x)=0

*Các biến toàn cục được sử dụng

- n *Bậc của đa thức*
- Mảng a[100] *Các hệ số của đa thức theo bậc giảm dần*
- a1, b1 *Khoảng để tìm các cực trị trong gói min-max*
- sign, map (ánh xạ) *Dùng trong GDA*
- Mảng diem[100] *Dùng để lưu trữ cận trên, cận dưới của nghiệm thực cùng các cực trị*
- m *Biến đếm của mảng diem[]*
- N1, N2 *Cận trên và cận dưới của nghiệm thực phương trình*

*Một số hàm được sử dụng trong thuật toán

- Gói min-max bao gồm các hàm:

  + **ff(x)** *Nhập hàm f(x)*

  + **f1(x0)** *Trả về hàm f'(x0)*

  + **fixeta(x0)** *Trả về eta hợp lí*

  + **gda(x0)** *Gradient Desent Asent – Tìm các cực trị)*

- **canbac(a, n1)** *Dùng để tính căn bậc n1 của số a (với a dương và n1 nguyên dương)*

  B1: Gán $f = e^{\frac{1}{n1}\ln(a)} = a^{\frac{1}{n1}} = \sqrt[n1]{a}$

  B2: Trả về giá trị của f

- **f(x)** *Tính giá trị của hàm f(x) tại x bằng Hoocne*

  B1: Gán $b_0 = a_0$

  B2: Khởi tạo biến lặp $i = 1$

  B3: Kiểm tra nếu $i \leq n$ thì $b_i = a_i + b_{i-1}c$, nếu sai chuyển đến B5

  B4: $i = i + 1$, quay lại B3

  B5: Trả về giá trị b_n

- **timmiennghiem()** *Tìm miền chứa nghiệm thực (-N2;N1) của phương trình*

B1: Tìm $A = \max\{|a_i|, i = \overline{1,n}\}$

B2: Tính $R = 1 + \frac{A}{|a_{0|}}$

B3: Tìm $k_1(a_{k_1}$ là hệ số âm đầu tiên)

B4: Tìm B_1 = max trị tuyệt đối các hệ số âm. Nếu không tồn tại B_1 thì báo về "phương trình không có nghiệm dương", gán $N_1 = 0$ rồi chuyển đến B6

B5: Tính $N_1 = 1 + \sqrt[k_1]{\frac{B_1}{a_0}}$

B6: Với $i \equiv 1(mod\ 2)$ thì $a_i = -a_i$ (Thay x bởi –x vào phương trình)

B7: Tìm $k_2(a_{k_2}$ là hệ số âm đầu tiên)

B8: Tìm B_2 = max trị tuyệt đối các hệ số âm. Nếu không tồn tại B_2 thì báo về "phương trình không có nghiệm âm", gán $N_2 = 0$ rồi chuyển đến B10

B9: Tính $N_2 = 1 + \sqrt[k_2]{\frac{B_2}{a_0}}$

B10: Gán a1 = -N2, b1 = N1

- **bisection(a,b)** *Thuật toán chia đôi với khoảng cách ly nghiệm (a,b)*

B1: Gán c = (a+b)/2

B2: Kiểm tra nếu f(c) = 0 hoặc f(c) < epsilon (cho trước), đúng thì in ra c rồi kết thúc

B3: Kiểm tra nếu f(c)*f(a) < 0, đúng thì gán b = c, nếu sai gán a = c, sau đó lặp lại từ B1 đến B3

*Hàm main()

B1: Nhập input n, mảng a[n]

B2: Tìm miền nghiệm (sử dụng hàm timmiennghiem())

B3: Dùng gói tìm min-max để tìm cực trị, lưu các cực trị vào mảng diem[]

B4: Lưu hai giá trị -N1, N2 vào mảng diem[]

B5: Sắp xếp mảng theo thứ tự tăng dần bằng **sort()**

B6: Khởi tạo biến lặp p = 0

B7: Kiểm tra nếu p > m+1 thì kết thúc thuật toán

B8: Kiểm tra f(diem[p]) = 0 hoặc |f(diem[p])| < epsilon, nếu sai chuyển đến B10

B9: In ra p là nghiệm, p = p+1

B10: Kiểm tra f(diem[p],diem[p+1]) < 0) nếu sai chuyển đến B12

B11: bisection(diem[p],diem[p+1]) *(do đây là khoảng cách ly nghiệm)*

B12: p = p+1, quay lại B7

VD1: Giải phương trình $x^4 + 3x^3 - 11x^2 - 3x + 10 = 0$

Input:

```
double ff(double x)  //Nhap ham f(x)
{
    return pow(x,4)+3*pow(x,3)-11*pow(x,2)-3*x+10;
}
```

C:\Users\Admin\Downloads\giaiptdt.exe

```
Nhap bac cua da thuc: 4
Nhap cac he so: 1
3
-11
-3
10

Khoang chua nghiem cua phuong trinh la (-12.000000,4.316625)

-5.000000
-1.000000
1.000000
2.000000
--------------------------------
Process exited after 8.364 seconds with return value 0
Press any key to continue . . .
```

Output: Chương trình đưa ra kết quả phương trình có 4 nghiệm thực -5, -1, 1 và 2

VD2: Giải phương trình $(x + 1)^3 = 0$ (phương trình có nghiệm tại điểm uốn)

Input:

```
double ff(double x)  //Nhap ham f(x)
{
    return pow(x,3)+3*pow(x,2)+3*x+1;
}
```

C:\Users\Admin\Downloads\giaiptdt.exe

```
Nhap bac cua da thuc: 3
Nhap cac he so: 1
3
3
1

Khoang chua nghiem cua phuong trinh la (-4.000000,0.000000)

-1.000577
--------------------------------
Process exited after 3.785 seconds with return value 0
Press any key to continue . . .
```

Output: Phương trình có nghiệm thực -1.000577

Nghiệm tìm được chỉ mang tính xấp xỉ so với nghiệm đúng của phương trình, do tính chất của gói min-max

Tuy nhiên có một nhược điểm khi step quá bé, một số điểm cực trị sẽ bị lưu lại nhiều lần với các kết quả xấp xỉ nhau, ví dụ khi cho step = 0.001 ở chương trình trên chỉ cho 1 kết quả nghiệm, nhưng khi thay step = 0.0001 thì sẽ được kết quả như dưới:

C:\Users\Admin\Downloads\giaiptdt.exe

```
Nhap bac cua da thuc: 3
Nhap cac he so: 1
3
3
1

Khoang chua nghiem cua phuong trinh la (-4.000000,0.000000)

-1.000577
-0.999877
-0.999677
-0.999475
--------------------------------
Process exited after 5.701 seconds with return value 0
Press any key to continue . . .
```

***Nhược điểm của chương trình: Ở B8 phép kiểm tra |f(diem[p])| < epsilon có thể không chính xác, vì gói min-max chỉ cho phép tính gần đúng các điểm cực trị. Do đó đối với các phương trình có nghiệm tại cực trị dễ xảy ra trường hợp thiếu kết quả.**

Bên cạnh đó do sử dụng phương pháp chia đôi nên đối với các phương trình có khoảng nghiệm lớn sẽ rất mất thời gian.