TRƯỜNG ĐẠI HỌC THỦY LỢI

GIẢI TÍCH SỐ

[Tài liệu giảng dạy ở bậc đại học]

Nguyễn Thị Vinh

HÀ NỘI 2010

1 CHƯƠNG 1: MỞ ĐẦU

1.1 GIẢI TÍCH SỐ LÀ GÌ

Giải tích số (Numerical Analysis) hay còn gọi là *Phương pháp số* (Numerical Methods) hay *Phương pháp tính* (Calculating Methods) là một khoa học nghiên cứu các lời giải số của các bài toán của toán học.

Ba nhiệm vụ chính của giải tích số là:

1. Xấp xỉ hàm số: Thay một hàm có dạng phức tạp bằng một hàm hoặc nhiều hàm có dạng đơn giản hơn. Các bài toán thường gặp là nội suy và xấp xỉ hàm.
2. Giải gần đúng các phương trình: Bao gồm các phương trình đại số và siêu việt, các hệ phương trình đại số tuyến tính và phi tuyến, giải các phương trình và hệ phương trình vi phân thường và vi phân đạo hàm riêng, …
3. Giải các bài toán tối ưu.

Tuy nhiên trong các giáo trình Giải tích số, người ta chỉ đề cập đến hai nhiệm vụ đầu, còn nhiệm vụ thứ ba dành cho các giáo trình về Qui hoạch toán học hay Tối ưu hoá.

"An approximate answer to the right problem is worth a great deal more than a precise answer to the wrong problem."

(John W Turkey 1915-2000)

1.2 SỰ KHÁC BIỆT GIỮA TOÁN HỌC LÍ THUYẾT VÀ TOÁN HỌC TÍNH TOÁN

TOÁN HỌC LÍ THUYẾT	TOÁN HỌC TÍNH TOÁN
Chứng minh sự tồn tại nghiệm	Tốc độ hội tụ của nghiệm
Khảo sát dáng điệu của nghiệm	Sự ổn định của thuật toán
Một số tính chất định tính của nghiệm	Thời gian tính toán trên máy và dung lượng nhớ cần sử dụng

Ví dụ 1: Tính tích phân

$$I_n = \int_0^1 x^n e^{x-1} dx \quad (n \ge 1)$$

Tích phân từng phần ta được

$$I_{n-1} = x^n e^{x-1}\Big|_0^1 - n\int_0^1 x^{n-1}e^{x-1}dx = 1 - nI_{n-1} \qquad (1.1)$$

$$I_1 = \int_0^1 xe^{x-1}dx = xe^{x-1}\Big|_0^1 - \int_0^1 e^{x-1}dx = e^{-1} \approx 0{,}36787$$

Vậy ta có thể tính tích phân trên và để ý rằng $I_n \ge 0$ với mọi n. Trên thực tế không phải như vậy! Công thức trên cho kết quả không chính xác, khi n = 9 $I_9 \approx$ =0,068480 < 0.
dù ta tăng độ chính xác của e^{-1} đến bao nhiêu đi nữa!
Nguyên nhân là do sai số ban đầu mắc phải khi tính e^{-1} bị khuếch đại lên sau mỗi lần tính.

Để khắc phuc, ta biến đổi công thức (1.1)

$I_{n-1} = n^{-1} (1 - I_n)$

và để ý rằng

$$0 \le I_n \le \int_0^1 x^n dx = (1+n)^{-1} \Rightarrow \lim_{n\to\infty} I_n = 0$$

Giả sử ta cần tính I_{19} và cho $I_{20} \approx 0$ thì sai số mắc phải là $\varepsilon_{20} < 1/21$. Khi đó $I_{19} \approx 1/20$ với sai số $\varepsilon_{19} < 1/21 \times 1/20$, … , và đến $I_9 \approx 0{,}091623$

Ví dụ 2: Giải hệ phương trình đại số tuyến tính $AX = b$ (1.2)
với A là ma trận vuông cấp n không suy biến ($\det A \neq 0$), b là véc tơ cột n thành phần. Do vậy có thể giải theo qui tắc Cramer:

$x_i = \Delta_i / \Delta$, $i = 1, \dots , n$ (1.3)

Để tính nghiệm theo (1.3), ta phải tính (n+1) định thức cấp n. Mỗi định thức có n! số hạng, mỗi số hạng có n thừa số, do đó phải thực hiện n-1 phép nhân. Vậy riêng số phép nhân phải thực hiện đã là (n+1) n! (n-1). Giả sử n = 20, và máy tính thực hiện được 5000 phép nhân trong 1 giây thì để thực hiện số phép nhân trên phải mất 300 000 000 năm!

Ví dụ 3: Xét hệ (1.2) với

$$A = \begin{pmatrix} 0{,}1 & 0 & \dots & 0 \\ 0 & 0{,}1 & \dots & 0 \\ \dots\dots & & & \\ 0 & 0 & \dots & 0{,}1 \end{pmatrix}, \ n = 100$$

Khi đó $\det A = 10^{-100} \approx 0$.

Theo quan điểm lí thuyết thi A bị suy biến nên không giải được. Tuy nhiên ta có thể nhẩm ra ngay $A^{-1} = 10\ E$, với E là ma trận đơn vị cấp n và dễ dàng tính được nghiệm của hệ bằng phương pháp ma trận nghịch đảo!

Kết luận: Trong quá trình giải các bài toán, có thể nảy sinh nhiều vấn đề mà toán học lí thuyết không quan tâm hoặc không giải quyết được. Vậy cần có một khoa học riêng chuyên nghiên cứu các phương pháp số giải các bài toán. Đó là khoa học tính toán mà giải tích số là một môn học

1.3 CÁC BƯỚC GIẢI MỘT BÀI TOÁN CỦA GIẢI TÍCH SỐ

Để giải quyết một bài toán, người ta phải thực hiện *quá trình mô phỏng* sau đây:

1. Xây dựng mô hình toán học của bài toán thực tế.
2. Phân tích mô hình: tính tương thích của mô hình với bài toán thực tế, sự tồn tại nghiệm của bài toán.
3. Rời rạc hoá mô hình: dùng các phương pháp tính toán để qui bài toán liên tục về bài toán với số ẩn hữu hạn.
4. Xây dựng thuật toán: chú ý đến độ phức tạp thuật toán, tính hội tụ, tính ổn định của thuật toán

5. Cài đặt chương trình và chạy thử, kiểm tra kết quả, sửa lỗi và nâng cấp chương trình

1.4 THUẬT TOÁN VÀ ĐỘ PHỨC TẠP

1.4.1 Thuật toán

➢ *Khái niệm thuật toán.*

Thuật toán là một dãy hữu hạn các bước, mỗi bước mô tả chính xác các phép toán hoặc hành động cần thực hiện, để giải quyết một vấn đề.

Năm đặc trưng của thuật toán:

- *Input.* Mỗi thuật toán cần có một số (có thể bằng không) dữ liệu vào (input). Đó là các giá trị cần đưa vào khi thuật toán bắt đầu làm việc. Các dữ liệu này cần được lấy từ các tập hợp giá trị cụ thể nào đó.
- *Output*. Mỗi thuật toán cần có một hoặc nhiều dữ liệu ra (output). Đó là các giá trị có quan hệ hoàn toàn xác định với các dữ liệu vào và là kết quả của sự thực hiện thuật toán.
- *Tính xác định*. Mỗi bước của thuật toán cần phải được mô tả một cách chính xác, chỉ có cách hiểu duy nhất
- *Tính khả thi.* Tất cả các phép toán có mặt trong các bước của thuật toán phải đủ đơn giản.
- *Tính dừng*. Mọi bộ dữ liệu vào thoả mãn các điều kiện của dữ liệu vào (tức là được lấy ra từ tập giá trị của các dữ liệu vào), qua xử lí bằng thuật toán phải dừng sau một số hữu hạn bước thực hiện.

➢ *Các vấn đề liên quan đến thuật toán.*

- *Tính đúng đắn của thuật toán*

Khi thuật toán được tạo ra chúng ta cần phải chứng minh rằng thuật toán được thực hiện sẽ cho ta kết quả đúng với mọi dữ liệu vào hợp lệ. Điều này gọi là chứng minh tính đúng đắn của thuật toán.

- *Các vấn đề nảy sinh*

Khi giải một bài toán người ta có thể xét nhiều thuật toán khác nhau xem độ phức tạp của chúng ra sao, dùng ngôn ngữ lập trình nào hay cài đặt phần mềm nào để chạy chương trình, cấu trúc dữ liệu nào phù hợp?

- *Các yêu cầu cơ bản khi giải một bài toán*

1 Hiểu đúng bài toán
2 Tìm đúng thuật toán
3 Không nhầm lẫn trong lập trình
4 Dữ liệu quét hết các trường hợp
5 Tốc độ tính toán nhanh
6 Bộ nhớ phù hợp
7 Phần mềm dễ sử dụng, dễ nâng cấp theo yêu cầu

1.4.2 Độ phức tạp thuật toán

➢ *Định nghĩa*

Độ phức tạp thuật toán là một công cụ đo, so sánh, lựa chọn các thuật toán khác nhau để tìm ra thuật toán tốt nhất cho lời giải bài toán

➢ *Hai tiêu chuẩn để đánh giá độ phức tạp thuật toán*

- Độ phức tạp về thời gian tính toán
- Độ phức tạp về phạm vi bộ nhớ dùng cho thuật toán (dung lượng của không gian nhớ cần thiết để lưu trữ dữ liệu vào, các kết quả tính toán trung gian và các kết quả của thuật toán)

Chúng ta sẽ chỉ quan tâm đến thời gian thực hiện thuật toán, vì vậy *một thuật toán có hiệu quả* được xem là *thuật toán có thời gian chạy ít hơn các thuật toán khác.*

➢ *Cách đánh giá thời gian thực hiện thuật toán*

- Cỡ của các dữ liệu vào.
- Chương trình dịch để chuyển chương trình nguồn thành mã máy.
- Thời gian thực hiện các phép toán của máy tính được sử dụng để chạy chương trình.

Thời gian chạy chương trình phụ thuộc vào rất nhiều nhân tố, nên ta không thể biết được chính xác thời gian chạy là bao nhiêu đơn vị thời gian chuẩn(bao nhiêu giây)

Thời gian thực hiện thuật toán như là hàm số của *cỡ dữ liệu vào n.*

Cỡ của dữ liệu vào là một tham số đặc trưng cho dữ liệu vào, nó ảnh hưởng quyết định đến thời gian thực hiện chương trình. Cỡ dữ liệu phụ thuộc vào thuật toán cụ thể, *thường là số nguyên dương n*. Ta sử dụng hàm T(n), trong đó n là cỡ dữ liệu vào để biểu diễn thời gian thực hiện một thuật toán.

Khi đánh giá thời gian thực hiện bằng phương pháp toán học, chúng ta sẽ bỏ qua nhân tố phụ thuộc vào cách cài đặt chỉ tập trung vào *xác định độ lớn của thời gian thực hiện T(n)*. Kí hiệu toán học *ô lớn* được sử dụng để mô tả độ lớn của hàmT(n).

Giả sử $n > 0$ ($n \in Z$), $T(n), f(n) > 0$. Ta viết $T(n) = O(f(n))$, nếu và chỉ nếu tồn tại các hằng số dương c và n_0 sao cho $T(n) \le c\, f(n)$, với mọi $n > n_0$. Ta có thể xem rằng $O(f(n))$ là cận trên của T(n). Thông thường thời gian chạy một thuật toán tỉ lệ với 1, logn, n, nlogn, n^k, hoặc a^n với a là hằng số

➢ *Các quy tắc để đánh giá thời gian thực hiện thuật toán*

- *Chia độ phức tạp thuật toán ra thành nhiều đoạn* mà mỗi đoạn có độ phức tạp

$$\left.\begin{array}{l} T_1(n) = O(f_1(n) \\ \ldots \\ T_q(n) = O(f_q(n) \end{array}\right\} \Rightarrow T_1(n) + \ \ldots \ + T_q(n) = O(\max(f_1(n), \ldots, f_q(n)))$$

Thật vậy vì $T_1(n), \ldots, T_q(n)$ là ô lớn của $f_1(n), \ldots, f_q(n)$ tương ứng do đó tồn tại hằng số $c_1, \ldots, c_q, n_1, \ldots, n_q$ sao cho

$T_1(n) \le c_1 f_1(n)$ với mọi $n > n_1$ và

$T_2(n) \le c_2 f_2(n)$ với mọi $n > n_2$,

...

Đặt $n_0 = \max(n_1,n_2, \ldots, n_q)$, khi đó với mọi $n > n_0$, ta có

$$T_1(n) + T_2(n) + \ldots + T_q(n) \le (c_1+c_2+ \ldots + c_q) \max (f_1(n),f_2(n), \ldots, f_q(n)).$$

- *Các loai lệnh*

- Các phép gán, đọc, in, goto là câu lệnh. Các lệnh này được gọi là lệnh đơn và có độ phức tạp thời gian là $T(n) = O(C) = O(1)$

- Nếu $S_1, S_2,.. .S_n$ là câu lệnh thì

$\{\ S_1; S_2; ...; S_n\ \}$

là câu lệnh hợp thành (câu lệnh ghép khối)

- Nếu $S, S_1, S_2, \ldots, S_n$ là các câu lệnh và $E_1, E_2, \ldots$ là biểu thức logic thì

```
if (E1)
      S;
else if (E2)
      S1;
      ......
else Sn;
```

gọi là câu lệnh `if`

- Nếu $S_1, S_2,.. .S_n$ là câu lệnh, E là biểu thức có kiểu thứ tự đếm được, và $v_1, v_2... , v_n$ là các giá trị cùng kiểu với E thì

```
switch
      case (constant)
            {
                  v1: S1; break;
                  v2: S2; break;
                  ............
                  vn : Sn; break;
                  default: Sn+1;
            }
```

lệnh này được gọi là câu lệnh `case`

- Nếu S là câu lệnh và E là biểu thức logic thì

```
while (E)  S;
```

là câu lệnh `while`

- Nếu S là câu lệnh và E là biểu thức logic thì

```
do S while (E);
```

là câu lệnh `do … while`

- Nếu S là câu lệnh, E là biến kiểu thứ tự đếm được thì

```
for (i = E ; i <= n; i++) S;
```

là câu lệnh `for`

- *Đánh giá thời gian thực hiện một chương trình*

Nhờ định nghĩa đệ quy các lệnh, chúng ta có thể phân tích một chương trình xuất phát từ các lệnh đơn, rồi từng bước đánh giá các lệnh phức tạp hơn, cuối cùng đánh giá được thời gian thực hiện chương trình.

Giả sử lệnh gán không chứa lời gọi các hàm có thời gian thực hiện. Khi đó để đánh giá thời gian thực hiện một chương trình, ta có thể áp dụng phương pháp đệ quy sau đây:

- Thời gian thực hiện các lệnh đơn: gán, đọc, viết, `goto` là O(1).
- Lệnh hợp thành. Thời gian thực hiện lệnh hợp thành được xác định bởi luật tổng.
- Lệnh `if`: Giả sử thời gian thực hiện các lệnh S_1, S_2 là $O(f_1(n))$ và $O(f_2(n))$ tương ứng. Khi đó thời gian thực hiện lệnh `if` là $O(\max(f_1(n),f_2(n)))$.
- Lệnh `case` được đánh giá như lệnh `if`.
- Lệnh `while`: Giả sử thời gian thực hiện lệnh S (thân của lệnh while) là O(f(n). Giả sử g(n) là số tối đa các lần thực hiện lệnh, khi thực hiện lệnh `while`. Khi đó thời gian thực hiện lệnh `while` là O(f(n)g(n)).
- Lệnh `for` được đánh giá tương tự lệnh `while`.

Nếu lệnh gán có chứa các lời gọi hàm thì thời gian thực hiện nó không thể xem là O(1) được, vì khi nó thực hiện lệnh gán còn phụ thuộc vào thời gian thực hiện các hàm có trong lệnh gán. việc đánh giá các hàm không đệ quy được tiến hành bằng cách áp dụng các quy tắc trên (việc đánh giá thời gian thực hiện các hàm đệ quy sẽ khó khăn hơn nhiều).

Chú ý:

- Nếu thuật toán được chia thành nhiều đoạn thì trong trường hợp
  Đoạn chia có chu trình lồng thắt: độ phức tạp tính là tích
  Đoạn rời rạc: độ phức tạp tính là max
- Người ta chia các bài toán thành 3 lớp, có
  Độ phức tạp là hàm đa thức (của cỡ dữ liệu vào n)
  Độ phức tạp là hàm mũ
  Độ phức tạp là NP - đầy đủ

- Hai loại bài toán sau là không khả thi, đối với loại đầu nên giảm bậc của đa thức về càng gần 1 càng tốt. Để hạ bậc của đa thức, người ta thường sử dụng cấu trúc dữ liệu đầu vào (input). Các dữ liệu đầu vào là bình đẳng với nhau.

Ví dụ 1: Lệnh lặp

```
for (i = 1; i <= n-1; i++)
    for (j = 1; j <= n-1; j++)  S = S + x[i,j];
```

có $T(n) = O(Cnn) = O(Cn^2)$

Ví dụ 2: Đánh giá thời gian thực hiện của 2 thuật toán tính max của dãy a[1], …, a[n]

`max:=a[1];`
`for(i=1; i<=n-1; i++)`
`    for(j=i+1; j<=n; j++)`
	`max=a[1];`
	`for(j=2; j<=n; j++)`

`if(a[i]>a[j])`
`{  tg=a[i];`
`a[i]=a[j];`
`a[j]=tg; }`
`max=a[n];`
	`if (a[j]>max) max=a[j];`

Thuật toán này *làm thay đổi dãy đã cho*	Thuật toán này *không làm thay đổi dãy đã cho*
Số phép so sánh $\le (n-1)^2$ Số phép gán $\le 3\,(n-1)^2$ $T(n) = O(Cn^2)$	Số phép so sánh $= n-1$ Số phép gán $\le n$ $T(n) = O(Cn)$

1.5 SỐ XẤP XỈ VÀ SAI SỐ

1.5.1 Số xấp xỉ, sai số tuyệt đối và sai số tuong đối

- *Số xấp xỉ*

Định nghĩa 1: Số a được gọi là số xấp xỉ của số A nếu a sai khác A không đáng kể và được dùng thay A trong tính toán.

Ví dụ: 3,14 và 3,15 là 2 số xấp xỉ của số π

- *Sai số tuyệt đối và sai số tương đối*

Định nghĩa 2: $|A - a|$ gọi là *sai số tuyệt đối* của số xấp xỉ a

Nhận xét 1: Do không bết số đúng A nên không biết được $|A - a|$, người ta đưa thêm vào khái niệm sai số tuyệt đối giới hạn như sau:

Định nghĩa 3: *Sai số tuyệt đối giới hạn* của số a là số **Δa**: $|A-a| \le \Delta a$. Vậy.

$$a - \Delta a \le A \le a + \Delta a \quad \text{hay} \quad \boxed{A = a \pm \Delta a} \qquad (1.4)$$

Ví dụ: Ta biết rằng $3,14 \le \pi \le 3,15$ nên ta cố thể chọn $\Delta\pi \le 0,01$.

Nếu đê ý rằng $3,14 < \pi < 3,15$ thì ta nhận được giá trị tốt hơn. Người ta thường chọn Δπ là số nhỏ nhất thỏa mãn (1.4).

Nhận xét 2: Sai số tuyệt đối không thể hiện mức độ chính xác của phép đo hoặc tính toán. Để thể hiện điều đó, người ta đưa vào khái niệm sau:

Định nghĩa 4: *Sai số tương đối* của số xấp xỉ a là số

$$\delta = \frac{|A - a|}{|A|}$$

Định nghĩa 5: Sai số tương đối giới hạn của số xấp xỉ a là số δa**:** $\delta \le \delta a$

Nghĩa là $\delta a \ge \frac{\Delta a}{|A|}$ hay $\Delta a \le |A|\,\delta a$ (1.5)

Nhận xét 3: Do không biết số đúng A và vì $a \approx A$ nên thay vì dùng (1.5), ta dùng

$$\Delta a = |a|\,\delta a$$

hay

$$\boxed{A = a\,(1 \pm \delta a)} \qquad (1.6)$$

Ví dụ:

Mảnh vải A đo được chiều dài a = 6 m với sai số tuyệt đối giới hạn Δa = 0,2m. Mảnh vải B đo được chiều dài b = 3 m với sai số tuyệt đối giới hạn Δb = 0,2m. Rõ ràng 2 mảnh vải có cùng sai số tuyệt đối giới hạn là 0,2m và

$$A = 6 \pm 0{,}2;\ \ B = 3 \pm 0{,}2$$

nhưng sai số tương đối giới hạn của phép đo mảnh vải A là

$$\delta a = \frac{\Delta a}{|a|} = \frac{0{,}2}{6} = \frac{1}{30} = \frac{1}{2}\delta b$$

Vậy phép đo mảnh vải A chính xác hơn.

1.5.2 Cách viết số xấp xỉ

- *Chữ số có nghĩa của một số*: Chữ số có nghĩa của một số là những chữ số của số đó kể từ chữ số khác không đầu tiên tính từ trái sang phải

Ví dụ: 0,0052 có 2 chữ số có nghĩa và
23,05 có 4 chữ số có nghĩa

- *Chữ số đáng tin*: Chữ số đáng tin của một số thực là những chữ số ở vị trí thứ i (i = n, n-1, …,1, 0, -1, -2, …, m) thoả mãn điều kiện:

$\Delta a \leq 0{,}5\ 10^{\,i-n+1}$ với $n \geq 0$, $m \leq 0$ là các số nguyên.

Ví dụ: Số $a = 325{,}4 = 3 \times 10^2 + 2 \times 10 + 5 + 4 \times 10^{-1}$ có

$$\Delta a = 0{,}01 \leq 0{,}5 \times 10^{-1-2+1} = 0{,}5 \times 10^{-2}$$

Vậy chữ số 4 ở hàng thập phân và các chữ số ở bên trái nó là các chữ số đáng tin.
Qui ước: nếu viết số gần đúng mà không kèm theo sai số thì mọi chữ số đều là chữ số đáng tin

- *Cách viết số xấp xỉ*:

Cách 1: $A = a \pm \Delta a$

Cách 2: mọi chữ số có nghĩa đều đáng tin, tức là $\Delta a \leq 0{,}5.10^{-n}$ với n là vị trí của chữ số cuối cùng nằm bên phải, kể từ dấu phẩy.

Ví dụ: Số a = 3,14 có chữ số 4 là chữ số đáng tin và do đó các chữ số đứng trước nó đều đáng tin và số này có sai số tuyệt đối $\Delta a \leq 0{,}5.10^{-2}$

1.5.3 Qui tròn số và sai số qui tròn

Qui tròn một số là vứt bỏ một vài chữ số sao cho số nhận được a_1 được gần đúng nhất với số a ban đầu

- *Qui tắc qui tròn*

Sai số qui tròn tuyệt đối ≤ 0,5 đơn vị của chữ số ở hàng giữ lại cuối cùng

- *Sai số qui tròn*:

Gọi sai số qui tròn tuyệt đối của a là $\Theta a_1 = |a_1 - a|$, ta có

$$|A - a_1| = |A - a + a - a_1| \leq \Delta_a + \Theta a_1$$

Vậy ta có thể chọn được $\Delta a_1 = \Delta_a + \Theta a_1$ điều đó có nghĩa là *sau qui tròn, sai số tuyệt đối tăng lên*

Ví dụ: Số a = 3,141 được qui tròn thành 3,14, còn số 3,1415 được qui tròn thành 3,142

1.5.4 Các công thức tính sai số

- *Công thức chung*

Bài toán: Cho hàm $u = f(x_1, x_2, \ldots, x_n)$ khả vi và biết các sai số tuyệt đối giới hạn Δx_i của các đối số x_i và giá trị gần đúng u. Hãy tính sai số tuyệt đối Δ_u và sai số tương đối δ_u

Giải: Sử dụng các định nghĩa về sai số và công thức toán ta có

$$\Delta_u = \sum_{i=1}^{n}\left|\frac{\partial u}{\partial x_i}\right|\Delta_{x_i}, \quad \delta_u = \sum_{i=1}^{n}\left|\frac{\partial}{\partial x_i}\ln u\right|\Delta_{x_i} \tag{1.7}$$

Áp dụng công thức (1.7) cho các hàm số học ta có các công thức tính sai số sau.

- *Sai số của tổng đại số:*

Nếu $u = x_1 + x_2 + \ldots + x_n$ thì

$$\Delta_u = \Delta_{x_1} + \Delta_{x_2} + \ldots + \Delta_{x_n}$$

$$\delta_u = \frac{\Delta_u}{|u|} = \frac{\Delta_{x_1} + \Delta_{x_2} + \ldots + \Delta_{x_n}}{|u|}$$

- *Sai số của tích:*

Nếu $u = x_1\ x_2\ \ldots\ x_n$ thì

$$\delta_{x_1} + \delta_{x_2} + \ldots + \delta_{x_n}, \ \Delta_u = |u|\delta_u$$

- *Sai số của thương:*

Nếu $u = x_1 / x_2$ ($x_2 \neq 0$) thì

$$\delta_u = \delta_{x_1} + \delta_{x_2}, \ \Delta_u = |u|\delta_u$$

Ví dụ: Tính các sai số tuyệt đối và tương đối của thể tích hình cầu , biết $V = \pi d^3/6$; $d = 3{,}7$ cm $\pm$ 0,05 cm và $\pi = 3{,}14 + 0{,}0016$.

Giải: Theo công thức tính sai số của tích và thương ta có

$$\delta_V = \delta_\pi + 3\delta_d = \frac{0{,}0016}{3{,}14} + 3\frac{0{,}05}{3{,}7} = 0.0411 \Rightarrow \Delta_V = V.\delta_V = 1{,}0882 \approx 1{,}1\,(cm^3)$$

1.6 BÀI TẬP CHƯƠNG 1

1> Giải hệ phương trình

$$0{,}461x_1 + 0{,}311x_2 = 0.150$$
$$0{,}209x_1 + 0{,}141x_2 = 0.068$$

Bằng cách sử dụng phép toán số học làm tròn đến ba chữ số thập phân. Nghiệm chính xác là $x_1 = 1$, $x_2 = -1$; Bạn có so sánh gì giữa hai kết quả?

2> Xét thuật toán tạo nhiễu y cho đại lượng x:

$$A = x \times 10^n$$
$$B = A + x$$
$$y = B - A$$

Tính y khi x = 0,123456 và n = 3 bằng cách sử dụng phép toán số học làm tròn đến sáu chữ số thập phân. Sai số x – y bằng bao nhiêu?

3> Một hình vuông có cạnh a đo được bằng 12,04 (m) với độ chính xác của thước đo là $\Delta a = 0{,}005$ (m). Đánh giá sai số tương đối, sai số tuyệt đối của diện tích của hình vuông $S = a^2$ và tính ra diện tích này (chỉ giữ lại các chữ số có nghĩa).

4> Giả sử $z = 0{,}180 \times 10^2$ là nghiệm gần đúng của phương trình ax = b với $a = 0{,}111 \times 10^0$, $b = 0{,}200 \times 10^1$. Hãy sử dụng phép tính số học với ba chữ số thập phân để tính sai số $\Delta = |b - az|$. Sau đó tính sai số trên bằng phép tính số học chính xác và so sánh các kết quả.

2 CHƯƠNG 2: GIẢI HỆ PHƯƠNG TRÌNH TUYẾN TÍNH

2.1 PHƯƠNG PHÁP KHỬ GAUSS - JORDAN

2.1.1 Thuật toán

- *Bước khử xuôi:* Đưa hệ

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + & \dots & + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + & \dots & + a_{2n}x_n = b_2 \\ \dots\dots\dots\dots & \dots\dots & \dots \\ a_{n1}x_1 + a_{n2}x_n + & \dots & + a_{nn}x_n = b_n \end{cases} \quad (2.1)$$

về *hệ dạng tam giác trên* bằng các phép biến đổi tương đương qua n lần tác động lên A

$$\begin{cases} a_{11}^{(n)}x_1 + a_{12}^{(n)}x_2 + & \dots & + a_{1n}^{(n)}x_n = b_1^{(n)} \\ \qquad a_{22}^{(n)}x_2 + & \dots & + a_{2n}^{(n)}x_n = b_2^{(n)} \\ \dots\dots\dots\dots & \dots\dots & \dots \\ & & a_{nn}^{(n)}x_n = b_n^{(n)} \end{cases} \quad (2.2)$$

Các phép biến đổi trên thực hiện được ở lần tác động thứ i lên ma trận A nếu

$$\left|a_{ii}^{(i-1)}\right| \neq 0 \ \forall i = \overline{1,n}$$

và là phần tử có trị tuyệt đối lớn nhất trên cột i, cụ thể ta có công thức biến đổi

$$a_{jk}^{(i)} = a_{jk}^{(i-1)} - a_{ji}^{(i-1)} a_{ik}^{(i-1)} / a_{ii}^{(i-1)} \quad \forall i = \overline{1,n};\ \forall j = \overline{i+1,n}; \forall k = \overline{i,n}$$

- *Bước quét ngược:* Giải lần lượt x_n, x_{n-1}, … , x_1 từ hệ tương đương đã được chéo hóa (2.2)

2.1.2 Ưu, nhược điểm của phương pháp

Phương pháp lặp Gauss-Jordan đơn giản, dễ lập trình, tuy nhiên *nếu* $\left|a_{ii}^{(i-1)}\right| \approx 0$ *thì kết quả có thể không chính xác.* Do vậy *ở ma trận biến đổi thứ i ta phải tìm phần tử trội trên cột i và thực hiện phép đổi chỗ 2 hàng nếu cần thiết để đảm bảo hàng thứ i là hàng chứa phần tử trội*

2.1.3 Các ví dụ

Ví dụ 1: Giải hệ

$$\begin{cases} 5x_1 - x_2 + 2x_3 = 6 \\ x_1 - 4x_2 + x_3 = -2 \\ -2x_1 - x_2 + 4x_3 = 1 \end{cases}$$

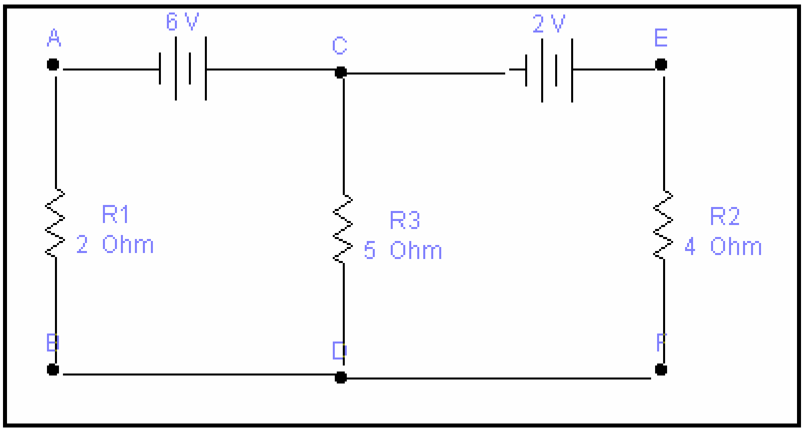
Bước khử xuôi: Đưa hệ đã cho về hệ tương đương

$$\left(\begin{array}{ccc|c} 5 & -1 & 2 & 6 \\ 0 & -3{,}8 & 0{,}6 & -3{,}2 \\ 0 & 0 & 4{,}579 & 4{,}579 \end{array}\right)$$

Bước quét ngược: Giải ra ta được $x_3 = 1$; $x_2 = 1$; $x_1 = 1$

Ví dụ 2:

Cho mạch điện như hình vẽ hãy áp dụng định luật Kirhoft để tìm giá trị các dòng điện qua các điện trở trong các mạch nhánh. Các giá trị cho trên hình vẽ:



Giải:

Áp dụng định luật Kirhoft 1 cho nút C ta có :

$$I_1 + I_2 - I_3 = 0 \tag{2.3}$$

Áp dụng định luật Kirrhoft 2 cho:

Vòng ACDB:

$$V_1 = R_1 I_1 + R_3 I_3 \tag{2.4}$$

Vòng AEFB :

$$V_1 - V_2 = R_1 I_1 - R_2 I_2 \tag{2.5}$$

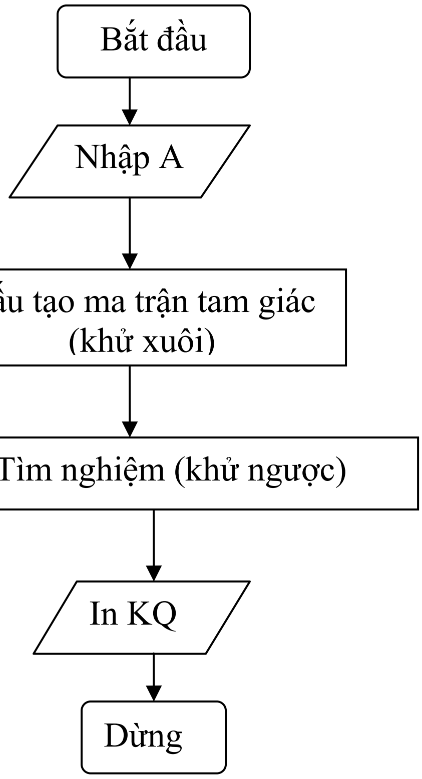
Thay các giá trị bằng số đã biết vào các phương trình (2.3), (2.4), (2.5) ta có hệ

$$\begin{cases} I_1 + I_2 - I_3 = 0 \\ 2I_1 + 5I_3 = 6 \\ 2I_1 - 4I_2 = 4 \end{cases}$$

Áp dụng phương pháp giải Gauss ta tìm được kết quả :

$I_1 = 1{,}158$; $I_2 = -0{,}421$; $I_3 = 0{,}737$(A)

2.1.4 Sơ đồ khối và chương trình




```
void Gauss(int n,matran a, mang1 x)
  {
     int i, j, k, max;
     float t;
     // Buoc khu xuoi
     for(i = 0; i < n; i++)  // Tim phan tu max cua cot i
       {
      max = i;
      for(j = i + 1; j < n; j++)
           if (fabs(a[j][i]) > fabs(a[max][i])) max = j;
      if (max != i) // Doi cho 2 hang max va i neu can
        for(k = i; k <= n; k++)
         {
           t = a[i][k];
           a[i][k] = a[max][k];
           a[max][k] = t;
         }
      for(j = i + 1; j < n; j++)//bien doi cot i
        for(k = n; k >= i; k--)
          a[j][k] = a[j][k] - a[i][k] * a[j][i] / a[i][i];
     // Buoc quet nguoc
     if (!(a[n-1][n-1]))
      {
        if(a[n-1][n]) printf("   HE VO NGHIEM");
        else printf("  HE VO SO NGHIEM");
```

```
    }
   else
    {
    for(j = n-1; j >= 0; j--)
          {
          t = 0.;
          for (k = j + 1; k < n; k++) t = t + a[j][k] * x[k];
          x[j] = (a[j][n] - t) / a[j][j];
          }
    printf("  \n\r  Nghiem cua he la: \n");
    for (i = 0; i<n; i++) printf("x[%d]= %8.4f\n",i,x[i]);
    }
}
```

2.1.5 Đánh giá độ phức tạp thời gian

Thời gian chạy của bước khử ngược là $O(n^2)$ nên hầu hết công việc được làm ở bước khử xuôi. Với mỗi giá trị của vòng lặp theo i, vòng lặp theo k lặp n – i + 2 lần và vòng lặp theo j lặp n – i lần. Vậy số lần lặp tổng cộng là

$$\sum_{i=1}^{n}(n-i+2)(n-i)+O(n^2)=\frac{n^3}{3}+O(n^2)$$

Giá trị của –a[i][j]/a[i][i]được tính ngoài vòng lặp k nên mỗi lần lặp chỉ bao gồm một phép nhân và một phép cộng.

Từ đó ta có kết luận: Một hệ n phương trình, n ẩn giải được bằng phương pháp Gauss với khoảng n^3 / 3 phép cộng và nhân.

2.1.6 Ứng dụng phương pháp khử Gauss vào việc tính định thức

- *Thuật toán:* Sử dụng bước khử xuôi, ta biến đổi định thức đã cho về định thức dạng tam giác trên. Từ đó giá trị của định thức chính là tích các phần tử trên đường chéo chính của ma trận vế phải sau n lần tác động, nhân với phần dấu là $(-1)^p$, trong đó p là số lần đổi chỗ 2 hàng.

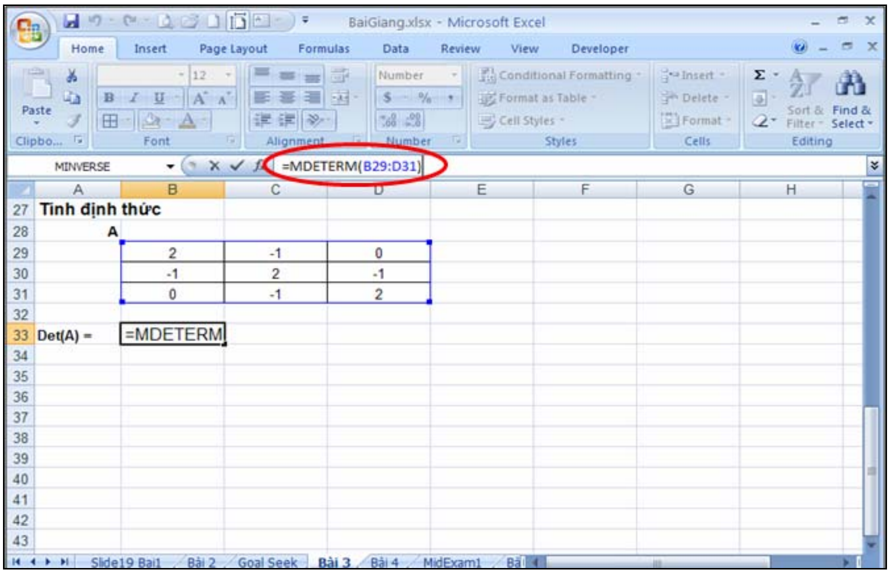
Ví dụ: Tính định thức của ma trận $A=\begin{bmatrix}1 & 0 & 1 & 2\\ -1 & 2 & 3 & 1\\ 4 & 0 & -2 & 1\\ 0 & 2 & 1 & 2\end{bmatrix}$

Sau bước khử xuôi ta đưa được A về dạng tam giác trên với p = 2 lần đổi chỗ các hàng

$$\begin{bmatrix}4 & 0 & -2 & 1\\ 0 & 2 & 2{,}5 & 1{,}25\\ 0 & 0 & 1{,}5 & 1{,}75\\ 0 & 0 & 0 & 2{,}5\end{bmatrix}$$

Lấy tích các phần tử trên đường chéo chính với $(-1)^2$ ta được det(A) = 30

➢ *Sử dụng hàm MDETERM trong EXCEL tìm det(A)*



2.2 GIẢI HỆ PTTT DẠNG BA ĐƯỜNG CHÉO

2.2.1 Đặt vấn đề

Khi ma trận vế phải của hệ A X = b có dạng ba đường chéo, một số thuật toán đơn giản giúp giảm được đáng kể thời gian tính toán và dung lượng bộ nhớ

$$\left[\begin{array}{cccccccc|c}
c_1 & d_1 & 0 & 0 & \dots & 0 & 0 & 0 & e_1 \\
b_2 & c_2 & d_2 & 0 & \dots & 0 & 0 & 0 & e_2 \\
0 & b_3 & c_3 & d_3 & \dots & 0 & 0 & 0 & e_3 \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & 0 & 0 & 0 & \dots & c_{n-2} & d_{n-2} & 0 & e_{n-2} \\
0 & 0 & 0 & 0 & \dots & b_{n-1} & c_{n-1} & d_{n-1} & e_{n-1} \\
0 & 0 & 0 & 0 & \dots & 0 & b_n & c_n & e_n
\end{array}\right]$$

2.2.2 Áp dụng phương pháp khử Gauss–Jordan:

Nhận xét: Đối với bước khử xuôi, chỉ có trường hợp j = i+1 và k = i+1 cần phải có mặt vì a[j] [k] = 0 với mọi k > i + 1. Đưa hệ đã cho về dạng 2 đường chéo ở tam giác trên. Đối với ma trận có dạng như vậy, bước khử xuôi và bước quét ngược được rút gọn về 1 vòng lặp:

```
for (i = 0; i < n-1; i++)
        {
          a[i+1][i+1]-=a[i][i+1]*a[i+1][i]/a[i][i];
          a[i+1][n]-=a[i][n]*a[i+1][i]/a[i][i];
        }
for (i = n-1; i >= 0; i--)
      x[i]=(a[i][n]-a[i][i+1]*x[i+1])/a[i][i];
```

Vậy nếu giải hệ ba đường chéo bằng phương pháp Gauss-Jordan thì độ phức tạp thời gian là tuyến tính.

2.2.3 Phương pháp truy đuổi (nắn thẳng) giải hệ ba đường chéo

Dung lượng đòi hỏi để chứa mảng hai chiều có thể được thay thế bằng bốn mảng một chiều, mỗi mảng cho một đường chéo và một mảng cho vec tơ cột vế phải. Sau đó dùng phương pháp truy đuổi giải hệ, độ phức tạp thuật toán là thời gian tuyến tính.

- *Thuật toán*

Viết lại phương trình thứ i của hệ

$$b_i x_{i-1} + c_i x_i + d_i x_{i+1} = e_i, \ i = 1, \dots, n \quad , (b_1 = 0 \text{ và } d_n = 0) \tag{2.6}$$

và đưa vào quan hệ phụ:

$$x_{i+1} = f_i x_i + h_i, \ i=1,\dots, n \tag{2.7}$$

Thay (2.7) vào (2.6) ta tính được f_i và h_i bằng công thức truy hồi:

$$f_{i-1} = -\frac{b_i}{c_i + d_i f_i}; \quad h_{i-1} = \frac{e_i - d_i h_i}{c_i + d_i f_i} \quad \forall i = \overline{1, n-1} \tag{2.8}$$

Để tính f_{n-1} và h_{n-1} ta xét phương trình cuối cùng

$$b_n x_{n-1} + c_n x_n = e_n$$

hằng đẳng hệ số với (2.6) khi i = n, ta tính được

$$c_n x_n = b_n x_{n-1} + e_n \Leftrightarrow x_n = -\frac{b_n}{c_n} x_{n-1} + \frac{e_n}{c_n}$$

hay $\quad f_{n-1} = -\frac{b_n}{c_n}; \quad h_{n-1} = \frac{e_n}{c_n}$

rồi thay vào (2.8) tính ngược lên.

Tính x_1: xét phương trình đầu tiên của (2.6)

$$c_1 x_1 + d_1 x_2 = e_2$$

mà

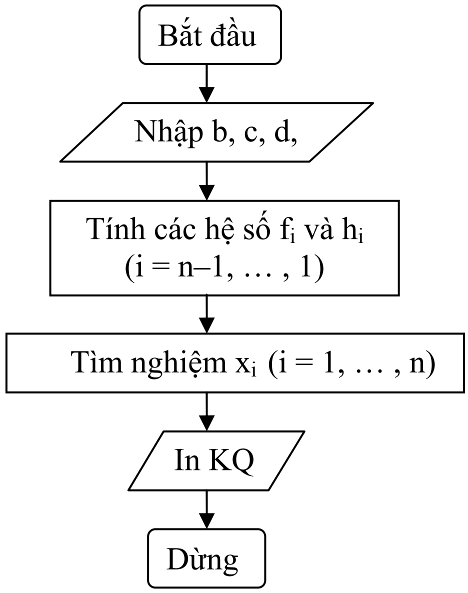
$$x_2 = f_1 x_1 + h_1.$$

Giải ra ta được

$$x_1 = \frac{e_1 - d_1 h_1}{c_1 + d_1 f_1}$$

Thay vào công thức (2.7) ta tìm được các x_i còn lại

➢ *Sơ đồ khối và chương trình*




```
void Tridiagonal(int n,mang1 b,mang1 c,mang1 d,mang1 e,mang1 x)
   {
     mang1 f,h;
     int i;
     // Tinh cac vec to f va h
     printf("\n\rTinh cac vecto he so f va h cua nghiem x: \n");
     f[n-2]=-b[n-1]/c[n-1];
     h[n-2]=e[n-1]/c[n-1];
     for (i=n-2;i>0;i--)
      {
        f[i-1]=-b[i]/(c[i]+d[i]*f[i]);
        h[i-1]=(e[i]-d[i]*h[i])/(c[i]+d[i]*f[i]);
      }
     for (i=0;i<=n-2;i++)
      printf(" f[%d] = %10.4f   h[%d] = %10.4f",i,f[i],i,h[i]);
     //Tinh nghiem X
     printf("\n\r     Nghiem cua he la: \n");
     x[0]=(e[0]-d[0]*h[0])/(c[0]+d[0]*f[0]);
     printf("  x[0] = %10.4f \n",x[0]);
     for (i=1;i<n;i++)
      {
       x[i]=f[i-1]*x[i-1]+h[i-1];
       printf("  x[%d] = %10.4f \n",i,x[i]);
      }
   }
```

2.3 PHƯƠNG PHÁP LẶP SEIDEL

2.3.1 Thuật toán

Bước 1: Đưa hệ

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \quad \dots \quad + a_{12}x_2 = b_1 \\ a_{21}x_1 + a_{22}x_2 + \quad \dots \quad + a_{2n}x_n = b_2 \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ a_{n1}x_1 + a_{n2}x_n + \quad \dots \quad + a_{nn}x_n = b_n \end{cases} \tag{2.9}$$

về hệ tương đương

$$\begin{cases} x_1 = c_{12}x_2 + \quad \dots \quad + c_{1n}x_n + b_1' \\ x_2 = c_{21}x_1 + c_{23}x_3 + \dots + c_{2n}x_n + b_2' \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ x_n = c_{n1}x_1 + \quad \dots \quad + c_{n\,n-1}x_{n-1} + b_n' \end{cases} \tag{2.10}$$

bằng cách ở phương trình thứ i chuyển các $x_k \neq x_i$ sang phải và chia 2 vế cho a_{ii}, i=1, ..., n

Bước 2: Thay xấp xỉ ban đầu $X^{(0)} = (x_1^{(0)},...,x_n^{(0)})$ của nghiệm vào (2.10) tìm xấp xỉ thứ nhất $X^{(1)} = (x_1^{(1)},...,x_n^{(1)})$ của nghiệm

$$\begin{cases} x_1^{(1)} = c_{12}x_2^{(0)} + \dots\dots\dots + c_{1n}x_n^{(0)} + b_1' \\ x_2^{(1)} = c_{21}x_1^{(1)} + c_{23}x_3^{(0)} + \dots + c_{2n}x_n^{(0)} + b_1' \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ x_n^{(1)} = c_{n1}x_1^{(1)} + \dots\dots\dots + c_{nn-1}x_{n-1}^{(1)} + b_1' \end{cases} \tag{2.11}$$

Ta sử dụng các $x_1^{(1)}$, ..., $x_i^{(1)}$ được tính ở các hàng trên để tính các $x_{i+1}^{(1)},...,x_n^{(1)}$, với i = 1, ..., n –1 ở các hàng dưới .

Bước 3: Tìm xấp xỉ thứ 2 $X^{(2)} = (x_1^{(2)},...,x_n^{(2)})$ của nghiệm như bước 2 bằng cách thay tương ứng $x_i^{(0)}$ bởi $x_i^{(1)}$,$x_i^{(1)}$ bởi $x_i^{(2)}$, i=1,..,n.

Tiếp tục như vậy cho đến bước lặp thứ i = M cho trước hoặc đến khi số bước lặp i thoả mãn

$$\left\| X^{(i)} - X^* \right\|_\infty \leq \varepsilon, \quad \varepsilon > 0 \text{ cho trước} \tag{2.12}$$

2.3.2 Điều kiện hội tụ và đánh giá sai số của phương pháp

- *Điều kiện đủ để phương pháp lặp Gauss-Seidel hội tụ*:

Nếu ma trận A có tính chất đường chéo trội, tức là

$$|a_{ii}| > \sum_{j=1,\, j\neq i}^{n} |a_{ij}| \quad , i = 1,..., n \tag{2.13}$$

thì nghệm $X^{(n)}$ hội tụ đến nghiệm duy nhất X* mà không phụ thuộc vào việc chọn véc tơ $X^{(0)}$ ban đầu.

- *Đánh giá sai số của phương pháp:*

Người ta chứng minh được rằng

$$\left\|X^{(k)} - X^*\right\|_\infty \le \frac{\mu}{1-\mu}\left\|X^{(k)} - X^{(k-1)}\right\|_\infty \qquad (2.14)$$

$$\left\|X^{(k)} - X^*\right\|_\infty \le \frac{\mu^k}{1-\mu}\left\|X^{(1)} - X^{(0)}\right\|_\infty$$

trong đó

$$\|X\|_\infty = \max_i |x_i|,$$

$$\mu = \max_i \frac{q_i}{1-p_i},\ i = 1, 2, \ldots, n,$$

$$p_i = \sum_{j=1}^{i-1}|c_{ij}|;\ \ q_i = \sum_{j=i}^{n}|c_{ij}| \qquad (2.15)$$

2.3.3 Ví dụ

Giải hệ sau với sai số $\varepsilon = 0,05$ và M = 10.

$$\begin{cases} 5x_1 - x_2 + 2x_3 = 6 \\ x_1 - 4x_2 + x_3 = -2 \\ -2x_1 - x_2 + 4x_3 = 1 \end{cases}$$

Do

$$|5| > |-1| + |2|$$
$$|-4| > |1| + |1|$$
$$|4| > |-2| + |-1|$$

$$\Rightarrow \begin{cases} x_1 = 0,2x_2 - 0,4x_3 + 1,2 \\ x_2 = 0,25x_1 + 0,25x_3 + 0,5 \\ x_3 = 0,5x_1 + 0,25x_2 + 0,25 \end{cases}$$

nên A thoả mãn tính chất đường chéo trội, vì vậy phép lặp Seidel hội tụ

Bước 1: Đưa hệ đã cho về hệ tương đương

Có $p_1=0,\ q_1=0,6;\ q1/(1-p_1)=0,6$

$p_2=0,25;\ q_2=0,25;\ q_2/(1-p_2)=1/3$

$p_3=0,75;\ q_3=0;\ q_3/(1-p_3)=0$

Vậy $\mu=0{,}6 \Rightarrow \dfrac{\mu}{1-\mu} = 1{,}5$

Bước 2: Thay xấp xỉ ban đầu, chẳng hạn $X^{(0)} = (0 \quad 0 \quad 0)$ của nghiệm vào vế phải tìm xấp xỉ thứ nhất $X^{(1)} = (x_1^{(1)}, x_2^{(1)}, x_3^{(1)})$ của nghiệm theo (2.11) với n=3

$$x_1^{(1)} = 0*0{,}2 - 0*0{,}4 + 1{,}2 = 1{,}2$$
$$x_2^{(1)} = 0{,}25*1{,}2 + 0*0{,}25 + 0{,}5 = 0{,}8$$
$$x_3^{(1)} = 0{,}5*1{,}2 + 0{,}25*0{,}8 + 0{,}25 = 1{,}05$$

Bước 3: Tìm xấp xỉ thứ hai $X^{(2)} = (x_1^{(2)}, x_2^{(2)}, x_3^{(2)})$ của nghiệm

$$\max_i \left| x_i^{(1)} - x_i^{(0)} \right| = \max\,(1{,}2;\ 0{,}8;\ 1{,}05) = 1{,}2$$

Vậy $\left\| X^{(1)} - X^* \right\| \le 1{,}5 * 1{,}2 = 1{,}8$

$$x_1^{(2)} = 0{,}2*0{,}8 - 0{,}4*1{,}05 + 1{,}2 = 0{,}94$$
$$x_2^{(2)} = 0{,}25*0{,}94 + 0{,}25*1{,}05 + 0{,}5 = 0{,}998$$
$$x_3^{(2)} = 0{,}5*0{,}94 + 0{,}25*0{,}998 + 0{,}25 = 0{,}97$$

$$\max_i \left| x_i^{(2)} - x_i^{(1)} \right| = \max(0{,}26;\ 0{,}198;\ 0{,}08) = 0{,}26$$

Vậy $\left\| X^{(2)} - X^* \right\| \le 1{,}5 * 0{,}26 = 0{,}39$

Bước 4: Tìm xấp xỉ thứ ba $X^{(3)} = (x_1^{(3)}, x_2^{(3)}, x_3^{(3)})$ của nghiệm

$$x_1^{(3)} = 0{,}2*0{,}998 - 0{,}4*0{,}97 + 1{,}2 = 0{,}95$$

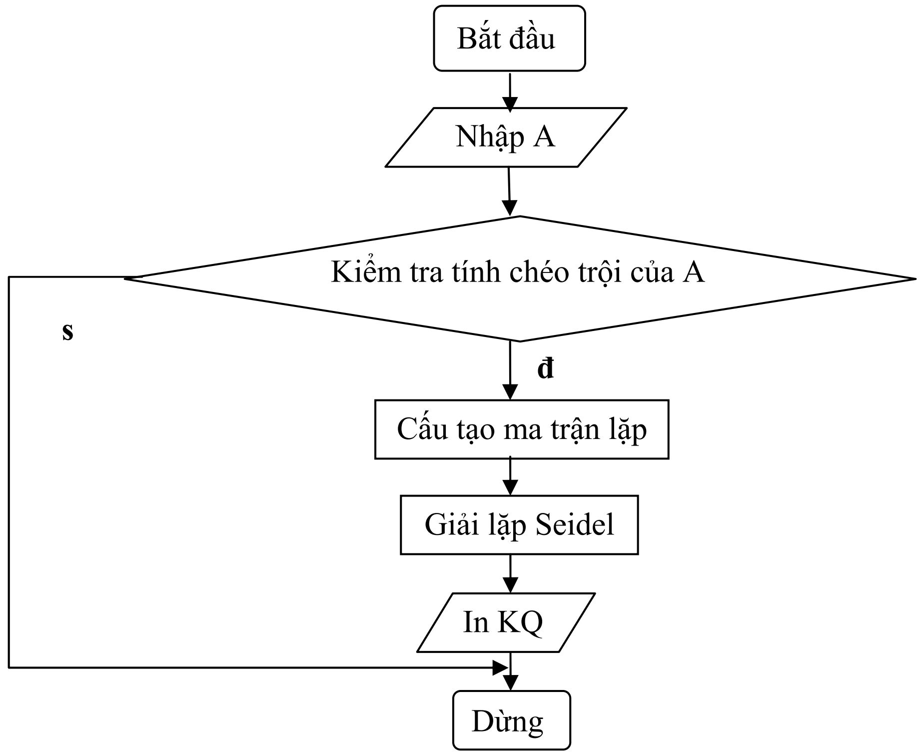
$$\max_i \left| x_i^{(3)} - x_i^{(2)} \right| = \max(0{,}01;\ 0{,}012; 0) = 0{,}012$$

Vậy $\left\| X^{(3)} - X^* \right\| \le 1{,}5 * 0{,}012 = 0{,}018 < \varepsilon$

$$x_2^{(3)} = 0{,}25*0{,}95 + 0{,}25*0{,}97 + 0{,}5 = 1{,}01$$
$$x_3^{(3)} = 0{,}5*0{,}95 + 0{,}25*1{,}01 + 0{,}25 = 0{,}97$$

Ta dừng ở xấp xỉ thứ ba của nghiệm vì sai số đã thỏa mãn độ chính xác ε cho trước

2.3.4 Sơ đồ khối và chương trình



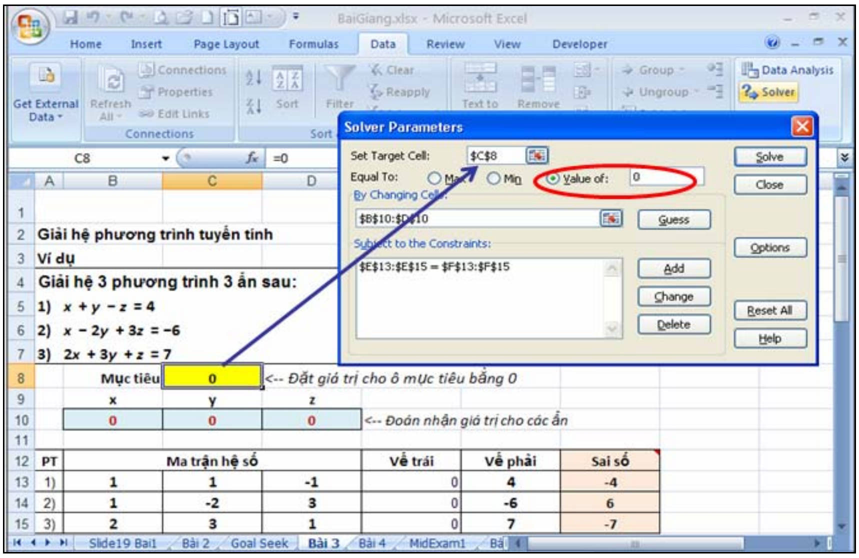

```
enum BOOLEAN  Hoitu (int  m,matran  a)
{
     int   i, j;
     float s;
     enum  BOOLEAN  TL, KT, KTH[nmax];
      TL=true;
      i = 0;
      do
      {
        KT = true;
        KTH[i] = true;
        if (a[i][i])
            {
            s = 0.;
            for (j=0;j <= m -1;j++)
                if((j!=i)) s = s + fabs(a[i][j]);
            if ((s/fabs(a[i][i])) >= 1.) KT = false;
            KTH[i] = KTH[i] && KT;
            }
        else KTH[i] = false;
        TL = TL && KTH[i];
        i++;
      }  while ((i <= m -1) && (TL==true));
      return TL;
 }
void  Matranlap (int m, matran a, matran b)
{
```

```
      int i, j;
      for (i = 0; i <= m -1; i++)
       {
      b[i][m] = a[i][m] / a[i][i];
      for (j = 0; j <= m -1; j++)
      if (i == j) b[i][j] = 0.;
      else b[i][j] = -a[i][j] / a[i][i];
       }
}
void LapSeidel(int m, int lanlap, float epsilon, matran b, mang1 xs)
{
       mang1  xt;
       int i, j, k;
       double s;
       enum BOOLEAN t;
       // Lay vec to ve phai lam vec to nghiem ban dau
       for (i = 0;i <= m -1; i++) xt[i] = b[i][m];
       k = 1;
       t = false ;
       do
       {
            for (i = 0; i <= m -1; i++)
                  {
                        xs[i] = 0;
                        for (j = 0; j <= m-1; j++)
                        if (j < i) xs[i] = xs[i] + b[i][j] * xs[j];
                        else xs[i] = xs[i] + b[i][j] * xt[j];
                        xs[i] = xs[i] + b[i][m];
                  }
            printf("    Nghiem lap thu %2d la:\n",k);
            for (i = 0; i <= m -1; i++)
                  {
                        printf("  x[%d] =%12.6f",i,xs[i]);
                        printf("\n");
                  }
            // Kiem tra dieu kien dung
            s = 0.0 ;
            for (i = 0; i < m -1; i++)  s += fabs(xs[i] - xt[i]) ;
            if (s <= epsilon)
               {
                 t = true ;
                 printf("\n");
                 printf("PHEP LAP HOI TU SAU %d BUOC\n" , k);
               }
            else
               printf("Sai so o lan lap thu %d la %12.2E" ,k,s);
            for (i = 0; i < m -1;i++)  xt[i] = xs[i] ;
            k++ ;
       } while((!t) && (k<=lanlap));
 }
```

2.3.5 Sử dụng Solver trong EXCEL giải hệ PTTT

- Nhập các giá trị đoán nhận ban đầu cho $x_1, x_2, \ldots, x_n$
- Nhập các vế trái của các phương trình $f_1, f_2, \ldots, f_n$ vào các ô riêng được liên kết với giá trị của các ẩn
- Chọn công cụ Solver (Data Tab → Solver)

  a. Đặt "Target Cell" cho ô có công thức = 0
  b. Chọn "Equal to Value of 0"
  c. Đối với "By Changing Cells," trỏ tới các ô chứa giá trị ban đầu của $x_1, x_2, \ldots, x_n$
  d. Thêm các ràng buộc (các phương trình)
  e. Nhấn nút "Options", chọn "Assume linear model"
  f. Nhấn nút "Solve"



2.4 TÍNH MA TRẬN NGHỊCH ĐẢO

2.4.1 Ứng dụng phương pháp Gauss tính ma trận nghịch đảo

Để tính ma trận nghịch đảo A^{-1} của ma trận A, ta dựa vào định nghĩa $A\,A^{-1} = E$ hay

$$\begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1n} \\ x_{21} & x_{22} & \ldots & x_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ x_{n1} & x_{n2} & \ldots & x_{nn} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & 1 \end{bmatrix} \qquad (2.16)$$

Để tính các cột của ma trận nghịch đảo ta giải Gauss – Jordan hệ n phương trình có cùng ma trận vế trái, còn các véc tơ cột vế phải lần lượt là các cột của ma trận đơn vị E.

Ví dụ: Tìm ma trận nghịch đảo của ma trận $A = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 3 & 1 \\ 1 & 2 & 1 \end{bmatrix}$

Các cột của A^{-1} lần lượt là nghiệm của các hệ phương trình ĐSTT sau đây với cùng ma trận vế phải A

$$\left[\begin{array}{ccc|ccc} 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & 3 & 1 & 0 & 1 & 0 \\ 1 & 2 & 1 & 0 & 0 & 1 \end{array}\right]$$

Giải ra ta được

$$A^{-1} = \begin{bmatrix} 1 & -1 & 1 \\ -1 & 0 & 1 \\ 1 & 1 & -2 \end{bmatrix}$$

2.4.2 Tính ma trận nghịch đảo A^{-1} bằng phương pháp lặp Newton

➢ *Đặt vấn đề*

Cho số a ≠ 0, cần tìm số x là số nghịch đảo của a. Đặt f(x) = a – 1/x, áp dụng phương pháp lặp Newton (tiếp tuyến) tìm nghiệm gần đúng của phương trình f(x) = 0

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - \frac{a - \frac{1}{x_n}}{\frac{1}{x_n^2}} = x_n\,(2 - a\,x_n) \qquad (2.17)$$

➢ *Thuật toán*

Áp dụng công thức lặp tương tự (2.17) để tìm ma trận nghịch đảo A^{-1} của ma trận không suy biến A. Gọi X_k là ma trận gần đúng của A^{-1} ở bước lặp thứ k, ta có

$$X_{n+1} = X_n\,(2E - AX_n)$$

(2.18) với E là ma trận đơn vị cùng cấp.

➢ *Sự hội tụ*

Đặt

$$\begin{aligned} G_k &= E - AX_k \text{ thay (2.18) vào} \\ &= E - AX_{k-1}(2E - AX_{k-1}) \\ &= E - 2AX_{k-1} + AX_{k-1}AX_{k-1} \\ &= (E - AX_{k-1})^2 = G^2_{k-1} \end{aligned}$$

$$\Rightarrow G_k = G_{k-1}^2 = G_{k-2}^4 = \ldots\ldots = G_0^{2k} \qquad (2.19)$$

Mặt khác do $A^{-1}A = E$ nên

$$A^{-1} - X_k = A^{-1}(E - AX_k) = A^{-1}G_k = A^{-1}\,G_0^{2k}$$

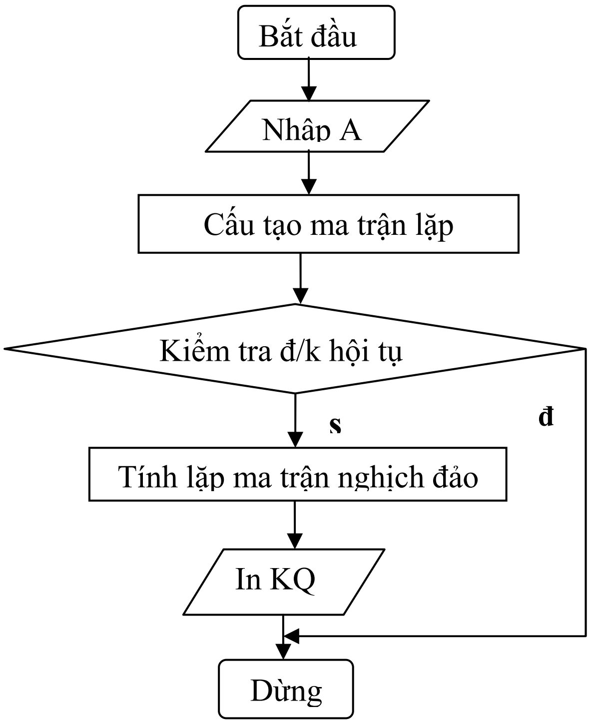
$$\Rightarrow \left\|A^{-1} - X_k\right\| \le \left\|A^{-1}\right\| \left\|G_0\right\|^{2k} \qquad (2.20)$$

Vậy nếu xấp xỉ đầu X_0 được chọn gần với A^{-1}, cụ thể là nếu

$$\boxed{\|E - AX_0\| = \|G_0\| < 1}$$

thì từ (2.20) dãy $\{X_k\}$ hội tụ về A^{-1} rất nhanh.

➢ *Sơ đồ khối và chương trình*

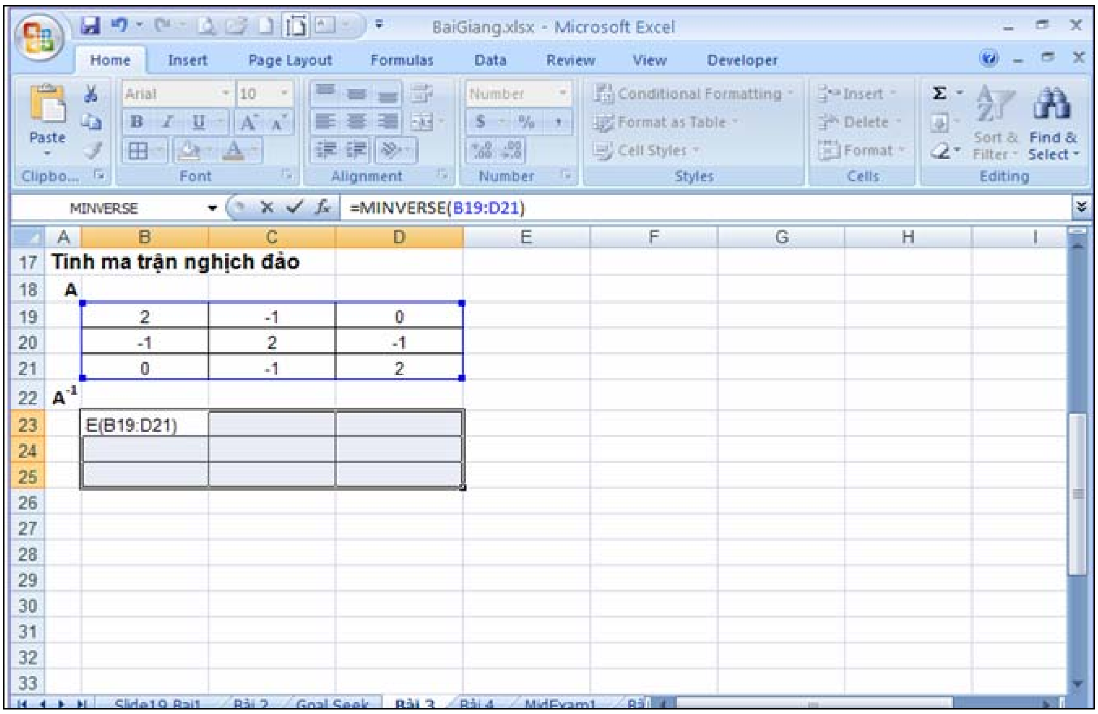


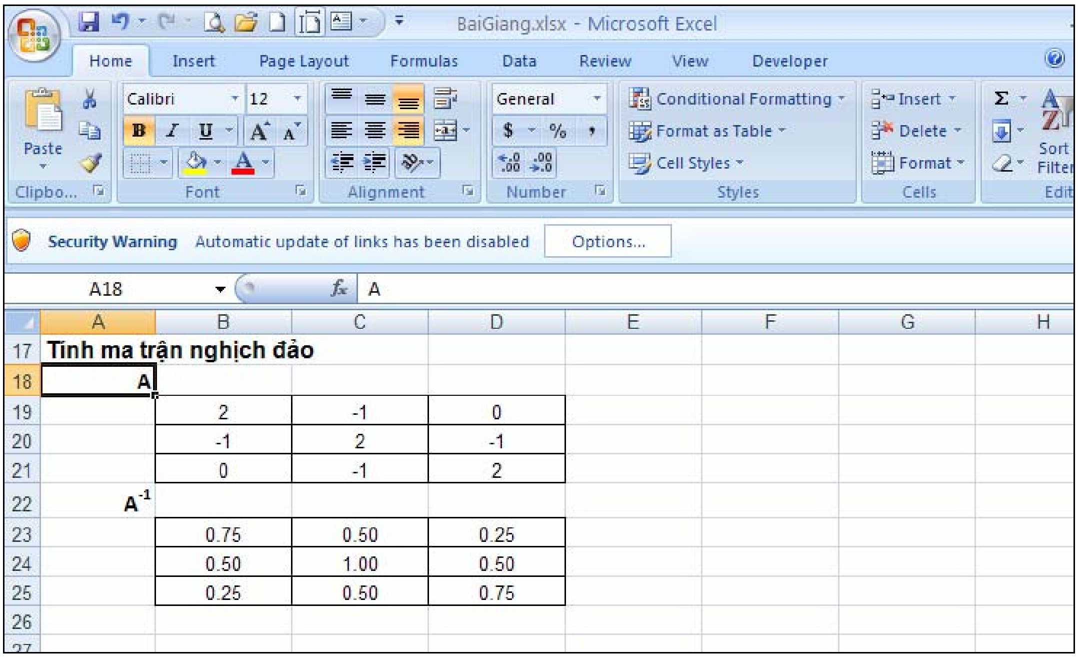

```
void Daomatran(int n,int lanlap,matrix a,matrix xt,matrix xs)
      {
      int i, j, k,dem;
      matrix g,y;
      double t;
      FILE *fp;
        // Tao ma tran lap
      for (i = 0; i <= n−1; i++)
          for (j = 0; j <= n −1; j++)
               for (k = 0; k <= n −1; k++)
                   {
                          if (i==j)
                                {
                                g[i][j] = 1. - a[i][k] * xt[k][j];
                                y[i][j]=g[i][j]+1;
                                }
                          else
                                {
                                g[i][j] = - a[i][k] * xt[k][j];
                                y[i][j]=g[i][j];
                                }
                   }
        // Kiem tra dieu kien hoi tu cua ma tran lap
        t=0.;
        for (i = 0; i <= n−1; i++)
```

```
        for (j = 0 ; j <= n -1 ; j++) t += g[i][j]*g[i][j];
    t = sqrt(t);
    if  (t >= 1)
      {
        printf("\tPhep lap khong hoi tu! \n");
        return;
      }
    else  //Tinh lap ma tran dao
      {
       dem = 1;
       do
         {
          for (i = 0; i <= n -1; i++)
                for (j = 0 ; j <= n -1 ; j++)
                      for (k = 0; k <= n -1; k++)
                            xs[i][j] = xt[i][k] * y[k][j];
          for (i = 0; i <= n -1; i++)
               for (j = 0 ; j <= n -1 ; j++) xt[i][j] = xs[i][j];
          dem++;
         } while (dem <= lanlap);
     }
}
```

2.4.3 Sử dụng hàm MINVERSE trong EXCEL tìm A^{-1}

1. Chọn mảng các ô chứa A^{-1}.
2. Nhập hàm MINVERSE tính A^{-1} với các tham số là các ô chứa ma trận A
3. Nhấn phím F2, rồi nhấn tiếp CTRL+SHIFT+ENTER




BaiGiang.xlsx - Microsoft Excel
Home
Insert
Page Layout
Formulas
Data
Review
View
Developer
Security Warning
Automatic update of links has been disabled
Options...
A18
A
Tính ma trận nghịch đảo
A
2 -1 0
-1 2 -1
0 -1 2
A-1
0.75 0.50 0.25
0.50 1.00 0.50
0.25 0.50 0.75

2.5 BÀI TẬP CHƯƠNG 2

1> Giải các hệ phương trình tuyến tính sau bằng phương pháp Gauss-Jordan

$$\begin{cases} 2x_1 + 5x_2 - 2x_3 = 3 \\ 3x_1 + 6x_2 + 9x_3 = 39 \\ x_1 + 3x_2 - x_3 = 2 \end{cases}$$

$$\begin{cases} x_1 + 2x_2 - x_3 = 2 \\ 2x_1 + 4x_2 + x_3 = 7 \\ 3x_1 + 6x_2 - 2x_3 = 7 \end{cases}$$

2> Sử dụng chương trình con **Gauss** để giải hệ phương trình tuyến tính sau

$$\begin{cases} 0{,}21\,x_1 + 0{,}32\,x_2 + 0{,}12\,x_3 + 0{,}31x_4 = 0{,}96 \\ 0{,}10\,x_1 + 0{,}15\,x_2 + 0{,}24\,x_3 + 0{,}22x_4 = 0{,}71 \\ 0{,}20\,x_1 + 0{,}24\,x_2 + 0{,}46\,x_3 + 0{,}36x_4 = 1{,}26 \\ 0{,}61\,x_1 + 0{,}40\,x_2 + 0{,}32\,x_3 + 0{,}20x_4 = 1{,}53 \end{cases}$$

Kiểm tra đáp án của bạn bằng cách thay lại vào hệ phương trình ban đầu, và đánh giá độ chính xác. Nghiệm đúng là $[1\ 1\ 1\ 1]^T$

3> Sử dụng chương trình con **Tridiagonal** để giải hệ phương trình tuyến tính sau

$$\begin{cases} -2(1 + h^2)x_1 + x_2 = 1 \\ -x_{i-1} - 2(1 + h^2)x_i + x_{i+1} = 0 \qquad i = 2, \ldots, n-1 \\ x_{n-1} - 2(1 + h^2)x_n = 1 \end{cases}$$

khi $n = 30$ và $h = 0{,}1$

4> Tìm ma trận nghịch đảo của ma trận sau bằng phương pháp Gauss-Jordan

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9{,}01 \end{pmatrix}$$

5> Viết chương trình con tính định thức của ma trận vuông cấp n, bằng cách sử dụng bước khử xuôi của phương pháp Gauss-Jordan đưa ma trận ban đầu về dạng tam giác trên.

Sử dụng chương trình này, tính định thức của ma trận sau

$$\begin{pmatrix} 0{,}217 & 0{,}732 & 0{,}414 \\ 0{,}508 & 0{,}809 & 0{,}376 \\ 0{,}795 & 0{,}886 & 0{,}338 \end{pmatrix}$$

3 CHƯƠNG 3: PHÉP NỘI SUY VÀ ĐƯỜNG CONG PHÙ HỢP

3.1 KHÁI QUÁT VỀ BÀI TOÁN NỘI SUY

3.1.1 Đặt vấn đề

Nhiều bài toán đòi hỏi tìm hàm f(x) từ tập các điểm (x_i, y_i) cho trước, chẳng hạn như việc thiết kế hệ thống tính toán điều khiển năng lượng cho một toà nhà cao tầng, đòi hỏi số liệu về sự biến đổi nhiệt độ ở đó trong mỗi ngày. Mẫu nhiệt độ thường được đo tại các thời điểm rời rạc. Tuy nhiên khi tính toán lại cần đến các giá trị nhiệt độ không có trong mẫu đo. Một trong các cách khắc phục vấn đề trên là tìm hàm đơn giản, thường là đa thức, đo đúng mẫu và nội suy những giá trị nằm giữa các điểm mẫu. Cũng có trường hợp biểu thức giải tích của f(x) đã biết nhưng quá phức tạp. Khi đó dùng phép nội suy ta có thể tính được giá trị của f(x) tại điểm x bất kì trên [a,b] mà độ chính xác không kém bao nhiêu.

Phép nội suy thường dùng trong các ứng dụng của khoa học, công nghệ và kinh tế. Cơ sở của mọi phép nội suy là sự phù hợp của kiểu đường cong hay hàm nội suy đối với tập các điểm mẫu cho dưới dạng bảng.

3.1.2 Đa thức nội suy

➢ *Bài toán*: Cho tập các điểm thực nghiệm (x_0,y_0), (x_1,y_1), ,(x_n,y_n), hãy xây dựng một đa thức

$$P_n(x) = a_0 + a_1x + \ldots + a_{n-1}x^{n-1} + a_nx^n \quad (3.1)$$

thoả mãn

$$P_n(x_i) = y_i \quad i = 0, 1, 2,\ldots, n \quad (3.2)$$

Các điểm x_i, y_i i = 0, 1, 2,…, n gọi là các mốc nội suy hay các điểm nút nội suy. Về mặt hình học, bài toán nội suy đa thức là tìm một đường cong y = $P_n(x)$ đi qua các điểm $M_0(x_0,y_0)$, $M_1(x_1,y_1)$, … , $M_n(x_n,y_n)$ cho trước

Sau đó dùng đa thức này để tính các giá trị $P_n(x)$ tại các giá trị x nằm giữa x_0 và x_n

➢ *Giải: n+1 hệ số a_i được xác định từ hệ phương trình ĐSTT*

$$\sum_{j=o}^{n} a_j x_i^j = y_i \quad i = \overline{0,n} \quad (3.3)$$

Định thức chính của hệ là

$$D = \begin{vmatrix} 1 & x_0 & x_0^2 & \ldots & x_0^n \\ 1 & x_1 & x_1^2 & \ldots & x_1^n \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & x_n & x_n^2 & \ldots & x_n^n \end{vmatrix} = \prod_{0 \le i < j \le n} (x_i - x_j) \neq 0$$

nên hệ tồn tại duy nhất nghiệm.

➢ *Nhận xét*: Phương pháp này có 2 nhược điểm

- Số phép tính cần thực hiện cỡ n $(\log n)^2$, chủ yếu là phép nhân.

- Nếu n lớn thì bậc của đa thức nội suy là quá cỡ!

3.1.3 Sơ đồ Horner tính giá trị của đa thức

- *Bài toán*: Cho đa thức (1), cần tính giá trị của đa thức tại x = c

  $$P_n(c) = a_0 + a_1c + \ldots + a_{n-1}c^{n-1} + a_nc^n \qquad (3.4)$$

- *Giải*: Cách tính $P_n(c)$ tiết kiệm nhất về số phép tính là viết lại (3.4) dưới dạng

  $$P_n(c) = (\ldots(((a_nc + a_{n-1})c + a_{n-2})c + a_{n-3})c + \ldots + a_1)c + a_0 \qquad (3.5)$$

Vậy để tính $P_n(c)$ ta chỉ cần tính lần lượt các số sau theo công thức truy hồi

$b_n = a_n$

$b_{n-1} = a_{n-1} + b_nc$

………………

$b_0 = a_0 + b_1c = P_n(c)$

hay lập thành sơ đồ Horner sau đây

	a_n	a_{n-1}	a_{n-2}	….	a_0	
+		b_nc	$b_{n-1}c$	…	b_1c	*c
	b_n	b_{n-1}	b_{n-2}		$b_0 = P_n(c)$	

- *Ví dụ*:

Tính giá trị của $P_3(x) = 3x^3 + 2x^2 - 5x + 7$ tại x = 3

Giải: Lập sơ đồ Horner

	3	2	-5	7	
+		9	33	84	*3
	3	11	28	91 = $P_3(3)$	

3.2 ĐA THỨC NỘI SUY LAGRANGE

3.2.1 Lập công thức

Với các mốc bất kỳ $x_0 < x_1 < \ldots < x_k < x_{k+1} < \ldots < x_N$ ta có các giá trị quan sát tương ứng $y_k = f(x_k)$., k=0, ..., N Tìm đa thức nội suy $L_N(x)$ bậc ≤ N sao cho $L_N(x_k) = y_k$, k=0,1, …, N

Đa thức nội suy Lagrange được cho như sau

$$f(x) \approx L_n(x) = \sum_{k=0}^{N} \frac{\prod_k(x)}{\prod_k(x_k)} y_k \qquad (3.6)$$

trong đó

$$\prod_k(x) = \prod_{j=0, j\neq k}^{N}(x - x_j) \qquad (3.7)$$

Kiểm tra điều kiện đường cong đa thức đi qua các điểm cho trước

$$L_n(x_i) = \sum_{k=0}^{N} \frac{\prod_k(x_i)}{\prod_k(x_k)} y_k \equiv y_i \ \forall\, i = \overline{0, N}$$

Chẳng hạn với N=3 ta có khai triển (3.6) như sau

$$f(x) \approx \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}y_0 + \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}y_1 + \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}y_2 + \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}y_3.$$

Ta có thể rút ra cách đơn giản như sau để triển khai (3.6):

- Vế phải có N+1 số hạng
- Trong số hạng thứ k, (k = 0,..,N) có 2 thừa số:

  Thừa số thứ hai là y_k

  Thừa số thứ nhất là phân số mà tử số có N thừa số không có x_k, mẫu số cũng có N thừa số như tử số khi thay x bởi x_k.

3.2.2 Ví dụ: Tìm giá trị gần đúng của f(2,6) từ bảng số liệu

x_k	–4	–1	2	5	8	11
y_k	–1	3	5	8	16	25

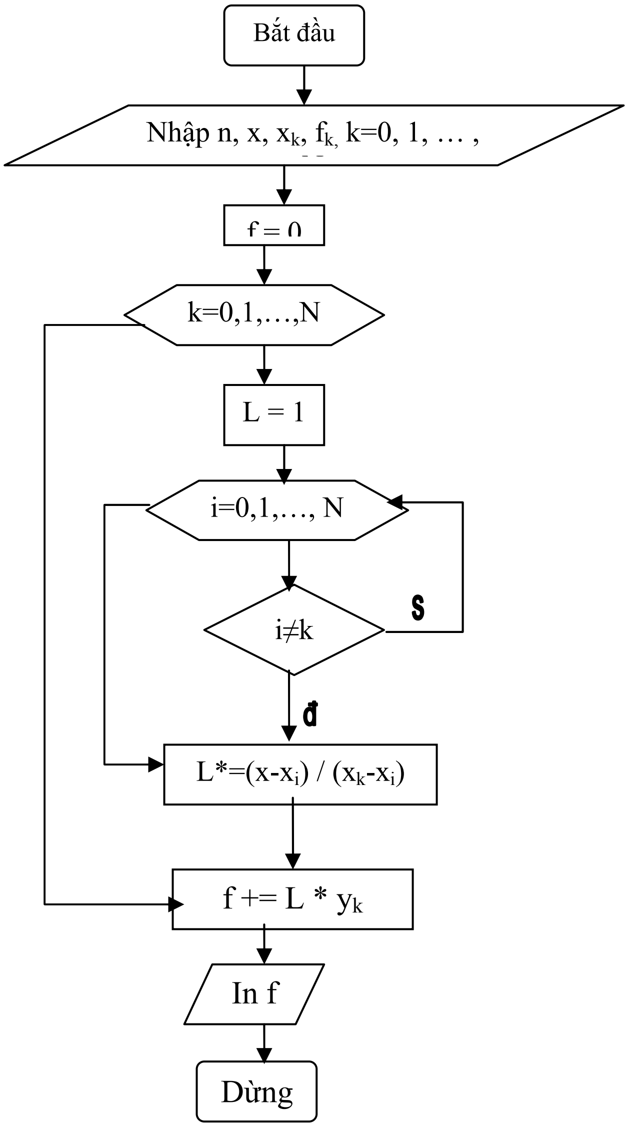
Áp dụng công thức (3.6) với N = 5, $x_0 = -4$, ..., $x_5 = 11$, x = 2,6:

$$f(2,6) \approx \frac{(2,6+1)(2,6-2)(2,6-5)(2,6-8)(2,6-11)}{(-4+1)(-4-2)(-4-5)(-4-8)(-4-11)}(-1) + \frac{(2,6+4)(2,6-2)(2,6-5)(2,6-8)(2,6-11)}{(-1+4)(-1-2)(-1-5)(-1-8)(-1-11)}3 + \frac{(2,6+4)(2,6+1)(2,6-5)(2,6-8)(2,6-11)}{(2+4)(2+1)(2-5)(2-8)(2-11)}5 + \frac{(2,6+4)(2,6+1)(2,6-2)(2,6-8)(2,6-11)}{(5+4)(5+1)(5-2)(5-8)(5-11)}8 + \frac{(2,6+4)(2,6+1)(2,6-2)(2,6-5)(2,6-11)}{(8+4)(8+1)(8-2)(8-5)(8-11)}16 + \frac{(2,6+4)(2,6+1)(2,6-2)(2,6-5)(2,6-8)}{(11+4)(11+1)(11-2)(11-5)(11-8)}25 = 5,3494$$

3.2.3 Sai số: Người ta đã chứng minh rằng nếu hàm f(x) khả vi liên tục đến cấp N+1 trên đoạn [a,b] chứa tất cả các mốc nội suy x_k, k = 0, ..., N thì sai số của nội suy Lagrange là

$$R_n(x) = |f(x) - L_N(x)| = \left|\frac{f^{(N+1)}(\xi)}{(N+1)!}\prod_{k=0}^{N}(x-x_k)\right|, x_0 < \xi < x_N \quad (3.8)$$

3.2.4 Sơ đồ khối và chương trình




```
…
printf("DOC DIEM CAN TINH \n " );
scanf("%f" , &x)
printf("TINH y = f(%6.2f) \n",x);
  y = 0 ;
  for (k = 0 ; k <= n ; k++)
  {
        L = 1;
        for (i = 0 ; i <= n ; i++)
              if (i != k)  L *= (x- x0[i]) / (x0[k] - x0[i]) ;
        y += L*y0[k];
  }
…
```

Nhận xét:

- Thuật toán có 2N(N+1) phép nhân và chia

- Khi các mốc nội suy cách đều, tức là $x_{k+1} = x_k + h = \ldots = x_0 + kh$, h là hằng số, đặt

$$u = (x - x_0) / h \text{ với } u = 0, 1, \ldots, N$$

thì công thức nội suy Lagrange (3.6) có dạng

$$\boxed{f(x) \approx L_N(x) = L_N(x_0 + uh) = \sum_{k=0}^{N} \frac{(-1)^{N-k} u(u-1)\ldots(u-k+1)(u-k-1)\ldots(u-N)}{k!(N-k)!} f_k} \quad (3.9)$$

- Nhược điểm: Khi tăng hoặc giảm số mốc nội suy, ta phải tính lại tất cả các số hạng của đa thức mới chứ không kế thừa được các kết quả cũ.

3.3 ĐA THỨC NỘI SUY NEWTON VỚI BƯỚC CÁCH ĐỀU

3.3.1 Bảng sai phân hữu hạn

Với các mốc quan sát $x_k = x_0 + k.h$, h là hằng số, ta có các gía trị quan sát tương ứng $f_k = f(x_k)$, $k = -m, \ldots, n$. Sau đây ta sẽ trình bày khái niệm về sai phân hữu hạn.

➢ *Sai phânhữu hạn*

Sai phân hữu hạn bậc một $\Delta f_k = f_{k+1} - f_k$.

Sai phân hữu hạn bậc hai $\Delta^2 f_k = \Delta f_{k+1} - \Delta f_k$.

Sai phân hữu hạn bậc ba $\Delta^3 f_k = \Delta^2 f_{k+1} - \Delta^2 f_k$

....

Bảng sai phân hữu hạn

$x_k = x_0 + k.h$	f_k	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$	$\Delta^4 f_k$	...
x_{-1}	f_{-1}					
		Δf_{-1}				
x_0	$\boxed{f_0}$		$\Delta^2 f_{-1}$			
		$\boxed{\Delta f_0}$		$\Delta^3 f_{-1}$		
x_1	f_1		$\boxed{\Delta^2 f_0}$		$\Delta^4 f_{-1}$	
		Δf_1		$\boxed{\Delta^3 f_0}$		
x_2	f_2		$\Delta^2 f_1$		$\boxed{\Delta^4 f_0}$	
		Δf_2		$\Delta^3 f_1$		
x_3	f_3		$\Delta^2 f_2$			
		Δf_3				
x_4	f_4					
...						

Chú ý:

* Để thuận tiện cho cho việc tra cứu các số liệu, các giá trị chuẩn đặt trong khung

* Giá trị cột thứ 3 trở đi được tính bằng cách lấy giá trị dưới trừ giá trị trên tương ứng của cột trước.

➢ *Ví dụ*

Bảng sai phân hữu hạn cho hàm $f(x) = x^3$ với các mốc quan sát $x = 0 + 2k$, $k = -2, ..., 4$

$x_k = 0 + 2k$	f_k	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$	$\Delta^4 f_k$	$\Delta^5 f_k$
$x_{-2} = -4$	-64					
		56				
$x_{-1} = -2$	-8		-48			
		8		48		
$x_0 = 0$	$\boxed{0}$		0		0	
		$\boxed{8}$		48		0
$x_1 = 2$	8		$\boxed{48}$		0	
		56		$\boxed{48}$		
$x_2 = 4$	64		96		$\boxed{0}$	
		152		48		
$x_3 = 6$	216		144			
		296				
$x_4 = 8$	512					

Từ bảng này ta có thể tìm $f_{-1} = -8$, $\Delta f_{-2} = 56$, $\Delta^2 f_2 = 144$, ...

3.3.2 Đa thức nội suy Newton tiến

➢ *Bài toán*

Với các mốc quan sát cách đều $x_k = x_0 + k.h$, $k = 0,...,n$, h là hằng số và giá trị quan sát tương ứng $f_k = f(x_k)$ ta cần tìm đa thức nội suy dưới dạng

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)\ldots(x - x_{n-1}) \qquad (3.10)$$

➢ *Công thức nội suy Newton tiến*

Cho $x = x_0$, $a_0 = f_0$; $x = x_1$, $a_1 = \Delta f_0 / h$; … , $x = x_i$, $a_i = \Delta^i f_0 / (i!\, h^i)$

Đổi biến $u = (x - x_0) / h \Rightarrow x = x_0 + uh$

$$f(x_0 + uh) \approx f_0 + \frac{\Delta f_0}{1!}u + \frac{\Delta^2 f_0}{2!}u(u-1) + \frac{\Delta^3 f_0}{3!}u(u-1)(u-2) + \ldots + \frac{\Delta^n f_0}{n!}u(u-1)\ldots(u-n+1) \qquad (3.11)$$

Nhận xét: Công thức này cho xấp xỉ tốt nhất của f(x) khi x ở đầu bảng, vì vậy người ta thường áp dụng với x_0 ở đầu bảng.

➢ *Sai số*

Người ta đã chứng minh rằng

$$R_n(x) = \frac{h^{n+1}\left|f^{(n+1)}(\xi)\right|}{(n+1)!}u(u-1)\dots(u-n),\ x_0 < \xi < x_1 \qquad (3.12)$$

➢ *Ví dụ* Từ bảng số liệu

x_k	–4	–2	0	2	4	6
f_k	–64	–8	0	8	64	216

hãy tính gần đúng giá trị f(–3). Ở đây ta thấy rằng x_k là các mốc cách đều với h = 2 và x = –3 ở đầu bảng giá tri x_k nên ta chọn x_0 = –4. Vì vậy u = (–3 – (–4)) / 2 = 0,5

Bảng sai phân hữu hạn tiến tương ứng

$x_k = -4 + 2k$	f_k	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$	$\Delta^4 f_k$	$\Delta^5 f_k$
$x_0 = -4$	$\boxed{-64}$					
		$\boxed{56}$				
$x_1 = -2$	–8		$\boxed{-48}$			
		8		$\boxed{48}$		
$x_2 = 0$	0		0		$\boxed{0}$	
		8		48		$\boxed{0}$
$x_3 = 2$	8		48		0	
		56		48		
$x_4 = 4$	64		96			
		152				
$x_5 = 6$	216					

Theo công thức (3.11) ta tính được

$$f(-3) \approx -64 + 56.0,5 / 1! - 48.0,5.(0,5-1) / 2! + 48.\ 0,5.(0,5-1)(0,5-2) / 3!$$
$$+\ 0.0,5.(0,5-1)(0,5-2)(0,5-3) / 4!$$
$$+\ 0.0,5.(0,5-1)(0,5-2)(0,5-3)(0,5-4) / 5! = -27$$

Lưu ý: Đây là bảng số liệu của $f(x) = x^3$ cho nên giá trị đúng f(-3) = –27.
Theo công thức (3.12) thì $f^{(6)}(x) = 0$ vì vậy quả là sai số $R_5(-3) = 0$.

3.3.3 Đa thức nội suy Newton lùi

Trường hợp mốc quan sát cách đều xếp theo thứ tự giảm dần $x_n > x_{n-1} > \dots > x_0$, $x_k = x_0 + kh$, h là hằng số, các giá trị quan sát tương ứng $f_k = f(x_k)$, k = 0, ..., n

➢ *Lập công thức*

Đặt $x = x_n + uh$, tìm đa thức nội suy dưới dạng

$$P_n(x) = a_0 + a_1(x - x_n) + a_2(x - x_n)(x - x_{n-1}) + \dots + a_n(x - x_n)\dots(x - x_1) \qquad (3.13)$$

Cho $x = x_n$ ta được $a_0 = f_n$;
$x = x_{n-1}$ ta được $a_1 = \Delta f_{n-1} / h$;
…

$x = x_i$ ta được $a_i = \Delta^i f_{n-I} / (i!\, h^i)$

Đổi biến $u = (x - x_n) / h$ ta được

$$f(x_n + uh) \approx f_n + \frac{\Delta f_{n-1}}{1!}u + \frac{\Delta^2 f_{n-2}}{2!}u(u+1) + \frac{\Delta^3 f_{n-3}}{3!}u(u+1)(u+2) + \ldots$$

$$+ \frac{\Delta^n f_o}{n!}u(u+1)\ldots(u+n-1) \qquad (3.14)$$

Nhận xét: Công thức này cho xấp xỉ tốt nhất của f(x) khi x ở cuối bảng dữ liệu, vì vậy người ta thường áp dụng với x_n ở cuối bảng.

- *Sai số*:

  Người ta đã chứng minh rằng

$$R_n(x) = \frac{h^{n+1}\left|f^{(n+1)}(\xi)\right|}{(n+1)!}u(u+1)\ldots(u+n), \quad x_{n-1} < \xi < x_n \qquad (3.15)$$

- *Ví dụ*: Từ bảng số liệu

x_k	–2	–1	0	1	2	3
f_k	16	1	0	1	16	81

hãy tính gần đúng giá trị f(2,2). Ở đây ta thấy rằng x_k là các mốc cách đều với h = 1 và x = 2,2 cuối bảng giá tri x_k nên ta chọn $x_n = 3$

$\Longrightarrow$ u = (2,2 – 3) /1 = –0,8.

BẢNG SAI PHÂN HỮU HẠN LÙI TƯƠNG ỨNG

$x_k = x_0 + kh$	f_k	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$	$\Delta^4 f_k$	$\Delta^5 f_k$
x_0 = –2	16					
		–15				
x_1 = 1	1		14			
		–1		–12		
x_2 = 0	0		2		24	
		1		12		$\boxed{0}$
x_3 = 1	1		14		$\boxed{24}$	
		15		$\boxed{36}$		
x_4 = 2	16		$\boxed{50}$			
		$\boxed{65}$				
x_5 = 3	$\boxed{81}$					

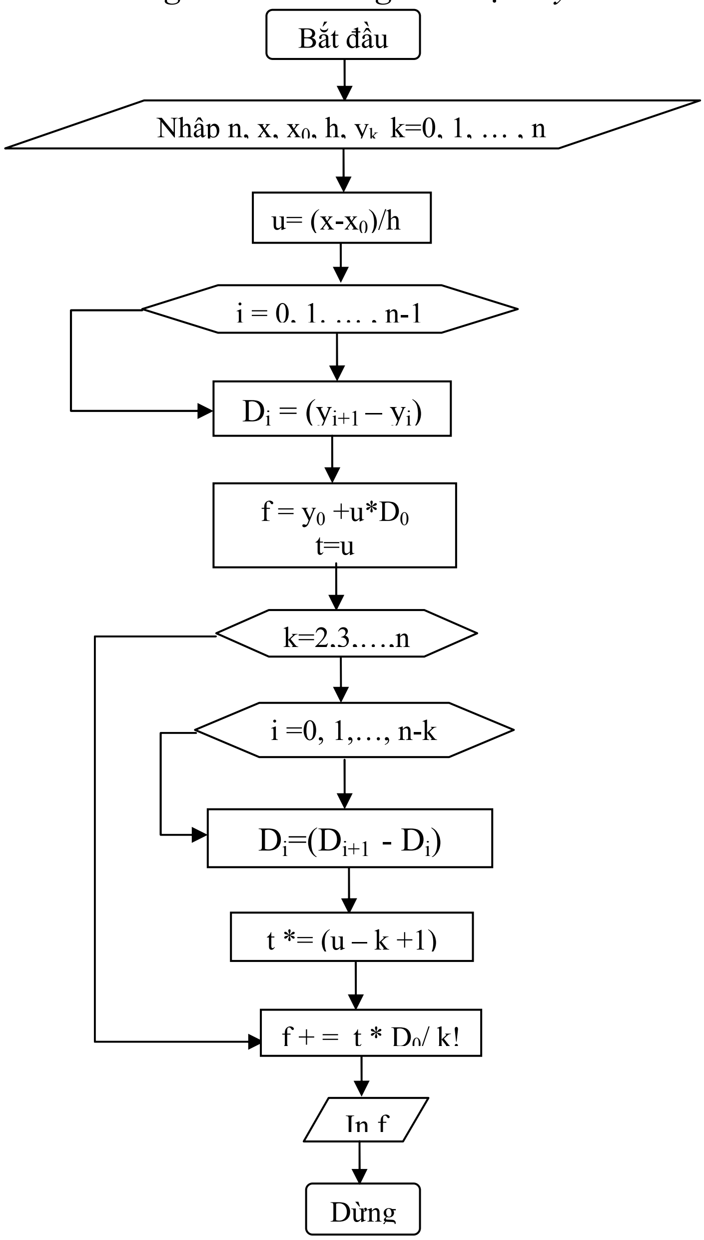
Theo công thức (3.14) ta tính được

f(2,2) ≈ 81 + 65.(–0,8/1! + 50.(–0,8).(–0,8 + 1) / 2!
+36.(–0,8).(–0,8 + 1)(–0,8 + 2) / 3!
+ 24.(–0,8).(–0,8 – 1)(–0,8 + 2)(–0,8 + 3) / 4!
+ 0.(–0,8).(–0,8 + 1)(–0,8 + 2)(–0,5 + 3)(–0,8 + 4) / 5! = 23,4256

Chú ý:

- Công thức nội suy Newton tiến (lùi) chỉ là cách viết khác của công thức Lagrange với mốc cách đều (3.9)
- Công thức nội suy Newton dựa trên các hệ số thứ k là các sai phân bậc k của f(x), các sai phân này có tính chất đều (đến bậc thứ $k \le n$ nào đó thì $\Delta^k f_i = 0$ với mọi i), do đó bậc của đa thức nội suy sẽ thấp đi
- Dùng công thức nội suy Newton tiến (lùi) không phải tính lại từ đầu nếu thêm mốc nội suy mới.

➢ *Sơ đồ khối và chương trình cho công thức nội suy Newton tiến*




```
…
 u = (x1 - x) / h; // x la moc noi suy dau tien
 for (i = 0; i <= n - 1; i++) D[i] = y[i+1] - y[i];
 f = y[0] + u * D[0]; t = u;
 for (k = 2; k <= n; k++)
     {
```

```
        for (i = 0 ; i <= n - k ; i++)  D[i] = D[i+1] - D[i];
        printf("\nSai phan cap %d = %8.6f",k,D[0]);
        t *= (u - k + 1);
        f += t * D[0] / GiaiThua(k);
    }
printf("\n Noi suy Newton tien f(%8.6f) = %8.6f",x1,f);
…
```

3.3.4 Công thức nội suy Newton với mốc quan sát bất kỳ

➢ *Công thức*:

Với các mốc quan sát bất kỳ $x_0 < x_1 < \ldots < x_k < x_{k+1} < \ldots < x_n$ và các giá trị quan sát $f_k = f(x_k)$, k=0,..., n tương ứng ta có công thức sau đây

$$f(x) \approx D_{00} + D_{11}(x-x_0) + D_{22}(x-x_0)(x-x_1) + \ldots + D_{nn}(x-x_0)(x-x_1)\ldots(x-x_{n-1}) \qquad (3.16)$$

trong đó các tỉ sai phân D_{kj} được định nghĩa như sau

$$D_{k0} = f_k,\ k = 0,\ldots,n \qquad (3.17)$$

$$D_{kj} = \frac{D_{k\,j-1} - D_{k-1\,j-1}}{x_k - x_{k-j}},\ j = \overline{1,n};\ k = \overline{j,n} \qquad (3.18)$$

➢ *Sai số*

$$R_n(x) = \left|\frac{f^{(n+1)}(\xi)}{(n+1)!}\prod_{k=0}^{n}(x - x_k)\right|,\ x_0 < \xi < x_n \qquad (3.19)$$

BẢNG TỶ SAI PHÂN

x_k	$D_{k0} = f_k$	D_{k1}	D_{k2}	D_{k3}	D_{k4}
x_0	$\boxed{D_{00} = f_0}$				
x_1	$D_{10} = f_1$	$\boxed{D_{11}}$			
x_2	$D_{20} = f_2$	D_{21}	$\boxed{D_{22}}$		
x_3	$D_{30} = f_3$	D_{31}	D_{32}	$\boxed{D_{33}}$	
x_4	$D_{40} = f_4$	D_{41}	D_{42}	D_{43}	$\boxed{D_{44}}$
...					

➢ *Ví dụ:* Tính gần đúng f(2,6) từ bảng số liệu với 4 chữ số sau dấu phẩy

x_k	–4	–1	2	5	8	11
f_k	–1	3	5	8	16	25

Theo công thức (3.17):ta có $\mathbf{D_{00} = -1}$, $D_{10} = 3$, $D_{20} = 5$, $D_{30} = 8$, $D_{40} = 16$, $D_{50} = 25$

Theo công thức (3.18) :

$\mathbf{D_{11}}= (D_{10} – D_{00}) / (x_1 – x_0) = (3 + 1) / (–1 + 4) =$ **1,3333**

$D_{21}= (D_{20} – D_{10}) / (x_2 – x_1) = (5 – 3) / (2 + 1) = 0,6667$

$D_{31}= (D_{30} – D_{20}) / (x_3 – x_2) = (8 – 5) / (5 – 2) = 1$

$D_{41}= (D_{40} – D_{30}) / (x_4 – x_3) = (16 – 8) / (8 – 5) = 2,6667$

$D_{51}= (D_{50} – D_{40}) / (x_5 – x_4) = (25 – 16) / (11 – 8) = 3$

$\mathbf{D_{22}}= (D_{21} – D_{11}) / (x_2 – x_0) = (0,6667 – 1,3333) / (2 + 4) =$ **–0,1111**

$D_{32}= (D_{31} – D_{21}) / (x_3 – x_1) = (1 – 0,6667) / (5 + 1) = 0,0556$

$D_{42}= (D_{41} – D_{31}) / (x_4 – x_2) = (2,6667 – 1) / (8 – 2) = 0,2778$

$D_{52}= (D_{51} – D_{41}) / (x_5 – x_3) = (3 – 2,6667) / (11–5) = 0,0556$

$\mathbf{D_{33}}= (D_{32} – D_{22}) / (x_3 – x_0) = (0,0556 + 0,1111) / (5 + 4) =$ **0,0185**

$D_{43}= (D_{42} – D_{32}) / (x_4 – x_1) = (0,2778 – 0,0556) / (8 + 1) = 0,0247$

$D_{53}= (D_{52} – D_{42}) / (x_5 – x_2) = (0,0556 – 0,2778) / (11 – 2) = –0,0247$

$\mathbf{D_{44}}= (D_{43} – D_{33}) / (x_4 – x_0) = (0,0247 – 0,0185) / (8 + 4) =$ **0,0005**

$D_{54}= (D_{53} – D_{43}) / (x_5 – x_1) = (–0,0247 – 0,0247) / (11 + 1) = –0,0041$

$\mathbf{D_{55}}= (D_{54} – D_{44}) / (x_5 – x_0) = (–0,0041 – 0,0005) / (11 + 4) =$ **–0,0003**

BẢNG TỶ SAI PHÂN

x_k	$D_{k0} = f_k$	D_{k1}	D_{k2}	D_{k3}	D_{k4}	D_{k5}
–4	$\boxed{-1}$					
–1	3	$\boxed{1,3333}$				
2	5	0,6667	$\boxed{-0,1111}$			
5	8	1	0,0556	$\boxed{0,0185}$		
8	16	2,6667	0,2778	0,0247	$\boxed{0,0005}$	
11	25	3	0,0556	–0,0247	–0,0041	$\boxed{-0,0003}$

Theo công thức (3.16):

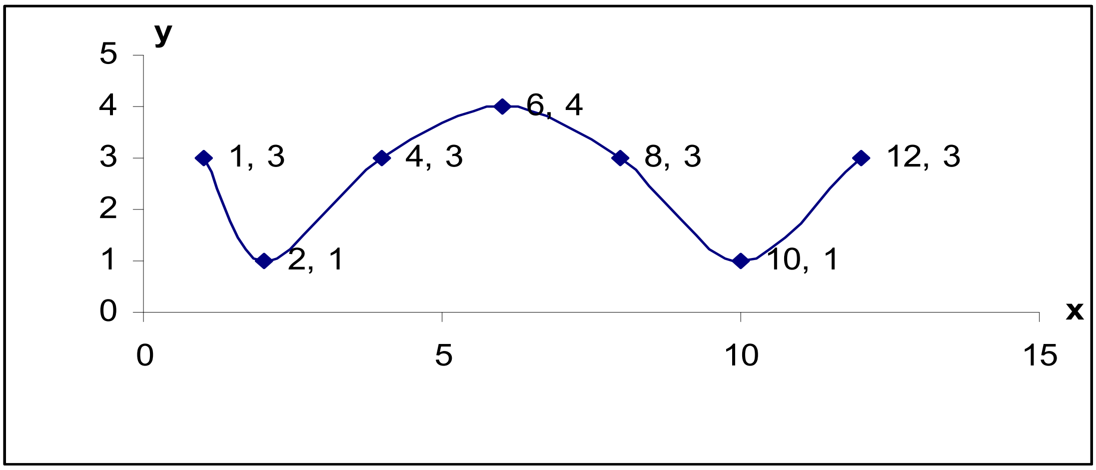
$$f(x) \approx \mathbf{-1} + \mathbf{1,3333}(x + 4) - \mathbf{0,1111}(x + 4)(x + 1) + \mathbf{0,0185}(x + 4)(x + 1)(x - 2)$$
$$+ \mathbf{0,0005}(x + 4)(x + 1)(x - 2)(x - 5)$$
$$\mathbf{- 0,0003}(x + 4)(x + 1)(x - 2)(x - 5)(x - 8)$$

Thay x = 2,6 ta được $f(2,6) \approx 5,3859$

3.4 NỘI SUY SPLINE

3.4.1 Đặt vấn đề

Các đa thức bậc thấp là những đường cong đơn giản mà người ta muốn dùng cho việc nội suy. Ý tưởng của phương pháp nội suy Spline (hàm ghép trơn) là thay vì tìm một đường cong đa thức đi qua mọi điểm nội suy, người ta tìm những đường cong bậc ba khác nhau nối các điểm kề nhau sao cho chúng trơn ở các điểm nối $M_i(x_i,y_i)$ đã cho, i = 0, 1,…, n.



3.4.2 Bài toán

Giả sử có hàm f(x) xác định trên đoạn $[x_0, x_n]$ và biết được giá tri của nó tại các điểm $x_0 < x_1 < \ldots < x_n$ là $y_0, y_1, \ldots, y_n$ tương ứng.

Hãy **tìm n đường ghép trơn bậc ba $S_i(x) = a_ix^3 + b_ix^2 + c_ix + d_i$** (3.20)

xác định trên $[x_{i-1}, x_i]$, i = 1, 2 ,… n sao cho

$$\begin{cases} S_i(x_{i-1}) = y_{i-1}, i = \overline{1,n} & (3.21) \\ S_i(x_i) = y_i, i = \overline{1,n} & (3.22) \\ S_i'(x_i) = S_{i+1}'(x_i), i = \overline{1,n-1} & (3.23) \\ S_i''(x_i) = S_{i+1}''(x_i), i = \overline{1,n-1} & (3.24) \\ S_1''(x_0) = 0 \\ S_n''(x_n) = 0 \end{cases}$$

3.4.3 Xây dựng công thức

Ta thấy hệ phương trình điều kiện có đủ 4n phương trình để tìm 4n ẩn a_i, b_i, c_i, d_i (i = 1, …, n)

Đặt

$$\boxed{\mathbf{S_i''(x_i) = p_i, i = \overline{1,n-1}, p_0 = S_1''(x_0) = 0, p_n = S_n''(x_n) = 0}} \quad (3.25)$$

Do $S_i''(x)$ là đa thức bậc nhất trên $[x_{i-1}, x_i]$ nên có thể tìm nó dưới dạng

$$\mathbf{S_i''(x) = \alpha_i(x_i - x) + \beta_i(x - x_{i-1})}.$$

Cho $x = x_{i-1}$ và $x = x_i$ ta có

$$\alpha_i = \frac{p_{i-1}}{x_i - x_{i-1}} = \frac{p_{i-1}}{h_i}, \ \beta_i = \frac{p_i}{x_i - x_{i-1}} = \frac{p_i}{h_i}, \ \ h_i = x_i - x_{i-1}$$

Khi đó lấy tích phân xác định 2 lần hàm $S_i''(x)$ trên $[x_{i-1}, x_i]$ ta được

$$S_i(x) = \frac{p_{i-1}}{6h_i}(x_i - x)^3 + \frac{p_i}{6h_i}(x - x_{i-1})^3 + \tilde{\alpha}_i(x_i - x) + \tilde{\beta}_i(x - x_{i-1}) \tag{3.26}$$

Lần lượt cho $x = x_i$ và $x = x_{i-1}$ ta tính được

$$\tilde{\beta}_i = \frac{1}{h_i}\left(y_i - \frac{p_i h_i^2}{6}\right), \tilde{\alpha}_i = \frac{1}{h_i}\left(y_{i-1} - \frac{p_{i-1} h_i^2}{6}\right)$$

Đổi biến

$$\boxed{t = \frac{x - x_{i-1}}{h_i}}$$

thay vào (3.26) ta có

$$S_i(t) = t y_i + (1-t) y_{i-1} + \frac{h_i^2\{(t^3 - t)p_i + [(1-t)^3 - (1-t)]p_{i-1}\}}{6} \tag{3.27}$$

Khi x □ $[x_i, x_{i+1}]$ thì t □ [0, 1], i = 1,…, n-1 và các điều kiện từ (3.21) đến (3.24) trở thành

$$S_i(1) = S_{i+1}(0) = y_i, \ i = 1, \dots, n-1$$

$$S'_i(1) = S'_{i+1}(0), \ i = 1, \dots, n-1$$

$$S''_i(1) = S''_{i+1}(0) = p_i, \ i = 1, \dots, n-1$$

Các p_i được xác định từ điều kiện (3.25) bằng cách lấy đạo hàm cấp một theo x của (3.20) viết theo t:

$$[S'_i(t)]_x = [S'_i(t)]_t \ t'_x = [S'_i(t)]_t \frac{1}{h_i} = z_i + \frac{h_i\{(3t^2 - 1)p_i - [3(1-t)^2 - 1]p_{i-1}\}}{6}, \ z_i = \frac{y_i - y_{i-1}}{h_i}$$

Do

$$\mathbf{S'_i(1) = S'_{i+1}(0)} \ \ \forall \mathbf{i} = \overline{\mathbf{1, n-1}}$$

ta có n-1 phương trình với n-1 ẩn $p_1, p_2, \dots, p_{n-1}$

$$\boxed{\mathbf{h_i\, p_{i-1} + 2\,(h_i + h_{i+1})\, p_i + h_{i+1}\, p_{i+1} = 6\,(z_{i+1} - z_i)},\ i = 1, \dots, n-1} \tag{3.28}$$

với ma trận hệ số là

$$\left(\begin{array}{cccccccc|c}
2(h_1 + h_2) & h_2 & 0 & 0 & \dots & 0 & 0 & 0 & w_1 \\
h_2 & 2(h_2 + h_3) & h_3 & 0 & \dots & 0 & 0 & 0 & w_2 \\
0 & h_3 & 2(h_3 + h_4) & h_4 & \dots & 0 & 0 & 0 & w_3 \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & 0 & 0 & 0 & 0 & h_{n-2} & 2(h_{n-2} + h_{n-1}) & h_{n-1} & w_{n-2} \\
0 & 0 & 0 & 0 & 0 & 0 & h_{n-1} & 2(h_{n-1} + h_n) & w_{n-1}
\end{array}\right),$$

$$w_i = 6(z_{i+1} - z_i)$$

Đây là hệ n – 1 hệ phương trình tuyến tính n – 1 ẩn dạng 3 đường chéo, đối xứng qua đường chéo chính và đường chéo chính là trội, nên luôn tồn tại duy nhất nghiệm.

3.4.4 Các bước giải bài toán nội suy Spline bậc ba

Nhập ma trận hệ số của hệ phương trình tuyến tính

Giải hệ phương trình tuyến tính tìm các giá trị p_i , i = 1, …, n – 1. Kiểm tra giá trị x cần nội suy nằm trên khoảng $[x_{i-1}, x_i]$ nào rồi tính

$$t = \frac{x - x_{i-1}}{h_i}$$

Sử dụng công thức (3.26) tính được $S_i(t)$

3.4.5 Ví dụ

Cho bảng các giá trị của hàm f(x). Hãy tính các hệ số p_i của đa thức ghép trơn Spline bậc ba trên các đoạn đã cho rồi tính gần đúng f(6,46).

i	x_i	h_i	y_i	$y_i - y_{i-1}$	$z_i = (y_i - y_{i-1})/h_i$	$w_i = 6(z_{i+1}-z_i)$
0	1		2			
1	2	1	1,5	–0,5	–0,5	2,25
2	4	2	1,25	–0,25	–0,125	0,45
3	5	1	1,2	–0,05	–0,05	0,15
4	8	3	1,125	–0,075	–0,025	0,075
5	10	2	1,1	–0,025	–0,0125	

Giải: Các hệ số p_1, p_2, p_3, p_4 là nghiệm của hệ 3 đường chéo sau

$$\left(\begin{array}{cccc|c} 6 & 2 & 0 & 0 & 2,250 \\ 2 & 6 & 1 & 0 & 0,450 \\ 0 & 1 & 8 & 3 & 0,150 \\ 0 & 0 & 3 & 10 & 0,075 \end{array}\right)$$

Giải hệ trên ta tìm được p_1= 0,39541; p_2 = –0,06123; p_3 = 0,02658; p_4 = –0,00048 ($p_0 = p_5 = 0$)

Do x = 6,46 thuộc vào đoạn thứ i = 4, cụ thể là $x \in [5; 8]$ nên

$$t = (6,46 - 5) / (8 - 5) \approx 0,48667$$

Thay giá trị của t và các giá trị $p_{i-1} = p_3 = 0,02658$ và $p_i = p_4 = -0,00048$ vào (3.27) ta tính được f(6,46) ≈ 1,14869. So với giá trị đúng của f(x) = 1 + 1/x tại x = 6,46 là 1,15480 chỉ sai khác 0,6%

3.4.6 Chương trình tính

```
float  Spline (int n, mang x, mang y, float v)
{
    mang  h, h1, d, w, p;
    int         i;
    float       t, s;
    for (i = 1; i <= n ; i++)
    {
     h[i] = x[i] - x[i-1];
```

```
      printf("  h[%d] = %10.6f \n",i,h[i]);
    }
    for (i = 1; i <= n -1 ; i++) d[i] = 2 * (h[i+1] + h[i]);
    for (i = 1; i <= n-1 ; i++) h1[i] = h[i+1];
    for (i = 1; i <= n - 1; i++)
         {
          w[i] = 6*((y[i+1]-y[i])/h[i+1]-(y[i]-y[i-1])/h[i]);
          printf("  w[%d] = %10.6f \n",i,w[i]);
         }
    p[0] = 0.; p[n] = 0.;
    Tridiagonal (n-1, h, d, h1, w, p);
    i = 1;
    while (v > x[i]) i++;
    t=(v-x[i-1])/h[i];
    s=t*y[i]+(1-t)*y[i-1]+h[i]*h[i]*(Tich(t)*p[i]+Tich(1-t)*p[i-1])/6;
    return s;
}
Hàm Tich(t) được định nghĩa như sau:
float Tich(float t)
   {
     return t*t*t - t;
   }
```

Nhận xét: thuật toán có độ phức tạp thời gian tuyến tính

3.5 PHƯƠNG PHÁP BÌNH PHƯƠNG BÉ NHẤT LÀM KHỚP DỮ LIỆU

3.5.1 Đặt vấn đề:

Phương pháp bình phương bé nhất thường được dùng để lập công thức thực nghiệm. Giả sử cần tìm quan hệ hàm giữa 2 đại lượng x và y dựa vào bảng số liệu thực nghiệm

x	x_1	x_2	…	x_n
y	y_1	y_2	…	y_n

Bên cạnh đó, ta còn *biết dạng của hàm phải tìm*. Hàm này phụ thuộc vào một số tham số, chẳng hạn $f(x) = f(c_1, c_2, \dots, c_m; x)$

Việc chọn hàm xấp xỉ tốt nhất trở thành việc *chọn các tham số phù hợp nhất với các điểm thực nghiệm.* Tiêu chuẩn để chọn là *tổng các bình phương của sai số ở các điểm thực nghiệm là bé nhất*, tức là bộ tham số phù hợp nhất phải là bộ số $(c_1^*, c_2^*, \dots, c_m^*)$ làm cực tiểu hàm m biến

$$E(c_1, c_2, \dots, c_m) = \sum_{i=1}^{n}[f(c_1, c_2, \dots, c_m; x_i) - y_i]^2 \tag{3.28}$$

Trong phạm vi giáo trình này, ta chỉ xét trường hợp dạng tham số tuyến tính tức là hàm thực nghiệm là tổ hợp tuyến tính của các hàm đơn giản hơn với các hệ số của tổ hợp chính là các tham số phải tìm

$$f(x) = c_1f_1(x) + c_2f_2(x) + \dots + c_mf_m(x) \tag{3.29}$$

chẳng hạn

$f(x) = a + bx$;
$f(x) = a + bx + cx^2$;
$f(x) = a + b/x$;
$f(x) = a + b\cos x + c\sin x$;
$f(x) = a + b\ln x$

hoặc có thể biến đổi về dạng tham số tuyến tính như

$f(x) = ae^{bx}$; $(a > 0)$
$f(x) = ax^b$; $(a > 0, b > 0)$

3.5.2 Lập công thức

Điểm $(c_1^*, c_2^*, ..., c_m^*)$ làm cực tiểu hàm $E(c_1,c_2,\ldots,c_m)$ phải là điểm dừng của hàm này, tức là phải thoả mãn hệ m phương trình

$$\frac{\partial E}{\partial c_j} = 2\sum_{i=1}^{n}[f(c_1,c_2,...c_m;x_i) - y_i]\frac{\partial f(c_1,c_2,...,c_m;x_i)}{\partial c_j} = 0,\ \forall j = \overline{1,m}$$

Khi $f(c_1,c_2,\ldots,c_m;x_i) = c_1f_1(x_i) + c_2f_2(x_i) + \ldots + c_mf_m(x_i)$, thay vào hệ trên ta được hệ đối xứng

$$\begin{cases} c_1(f_1,f_1) + c_2(f_1,f_2) + ... + c_m(f_1,f_m) = (f_1,y) \\ c_1(f_2,f_1) + c_2(f_2,f_2) + ... + c_m(f_2,f_m) = (f_2,y) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ c_1(f_m,f_1) + c_2(f_m,f_2) + ... + c_m(f_m,f_m) = (f_m,y) \end{cases} \quad (3.30)$$

trong đó $\mathbf{f_i} = (f_i(x_1), \ldots, f_i(x_n))$, $i = 1, \ldots, m$ và $\mathbf{y} = (y_1, \ldots, y_n)$, còn (f_i, f_j) là tích vô hướng giữa 2 véc tơ f_i và f_j., cụ thể

$$(f_i,f_j) = \sum_{k=1}^{n} f_i(x_k)f_j(x_k)\ ,\ (f_i\ ,y) = \sum_{k=1}^{n} f_i(x_k)y_k$$

Hệ phương trình tuyến tính(3.30) là đối xứng, giải được bằng phương pháp Gauss-Joordan.

3.5.3 Các ví dụ:

Ví dụ 1: Giả sử có 11 điểm thực nghiệm

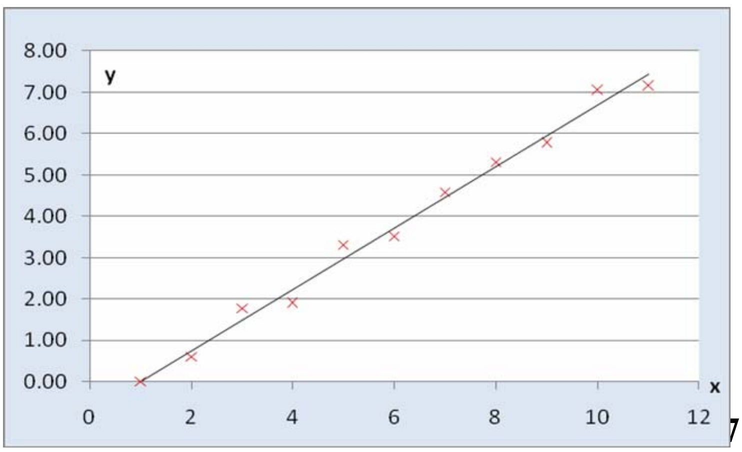
x	1	2	3	4	5	6	7	8	9	10	11
y	0	0.6	1.77	1.92	3.31	3.52	4.59	5.31	5.79	7.06	7.17

Tìm đường thẳng

$y = f(x) = c_1 + c_2x$

xấp xỉ tốt nhất tập điểm trên theo tiêu chuẩn bình phương bé nhất.

Giải: Trong trường hợp này $f_1(x) = 1$, $f_2(x) = x$



Lập bảng

i	$x = f_2$	y	f_1	(f_1,f_1)	(f_1,f_2)	(f_1,y)	(f_2,f_2)	(f_2,y)
1	1	0,00	1	1	1	0	1	0
2	2	0,60	1	1	2	0,6	4	1,2
3	3	1,77	1	1	3	1,77	9	5,31
4	4	1,92	1	1	4	1,92	16	7,68
5	5	3,31	1	1	5	3,31	25	16,55
6	6	3,52	1	1	6	3,52	36	21,12
7	7	4,59	1	1	7	4,59	49	32,13
8	8	5,31	1	1	8	5,31	64	42,48
9	9	5,79	1	1	9	5,79	81	52,11
10	10	7,06	1	1	10	7.06	100	70,60
11	11	7,17	1	1	11	7,17	121	78,87
			Tổng	**11**	**66**	**41,04**	**506**	**328,05**

Thay vào (3.30) ta có hệ 2 phương trình đối xứng của 2 ẩn c_1 và c_2 là

$$\left(\begin{array}{cc|c} 11 & 66 & 41.04 \\ 66 & 506 & 328.05 \end{array}\right)$$

Giải ra ta được $c_1 = -0,7314$ và $c_2 = 0,7437$. Vậy **$f(x) = -0,7314 + 0,7437x$**

Ví dụ 2: Giả sử có 6 điểm thực nghiệm

x	1	2	4	5	8	10
y	2,0	1,5	1,25	1,2	1,125	1,1

có thể làm khớp bởi đường cong $f(x) = c_1 + c_2 / x$. Hãy xác định c_1, c_2 bằng phương pháp bình phương bé nhất

Giải: Trong trường hợp này $f_1(x) = 1$, $f_2(x)=1/x$. Ta tính được

Lập bảng

i	x	y	$f_1 = 1$	$f_2 = 1/x$	(f_1,f_1)	(f_1,f_2)	(f_1,y)	(f_2,f_2)	(f_2,y)
1	1	2,000	1	1,000	1	1,000	2,000	1,000	2,000
2	2	1,500	1	0,500	1	0,500	1,500	0,250	0,750
3	4	1,250	1	0,250	1	0,250	1,250	0,063	0,313
4	5	1,200	1	0,200	1	0,200	1,200	0,040	0,240
5	8	1,125	1	0,125	1	0,125	1,125	0,016	0,141
6	10	1.100	1	0.100	1	0,100	1,100	0,010	0,110
				Tổng	**6**	**2,175**	**8,175**	**1,378**	**3,553**

Thay vào (3.30) ta có hệ 2 phương trình đối xứng của 2 ẩn c_1 và c_2 là

$$\left(\begin{array}{cc|c} 6 & 2,175 & 8,175 \\ 2,175 & 1,378 & 3,553 \end{array}\right)$$

Giải ra ta được $c_1 = 1$ và $c_2 = 1$ hay $\mathbf{f(x) = 1 + 1/x}$

3.5.4 Các bước giải và chương trình

- Nhập các véc tơ $\mathbf{x} = (x_1, \dots , x_n)$ và $\mathbf{y} = (y_1, \dots , y_n)$
- Lập ma trận $\mathbf{F_{mxn}}$ các vectơ $\mathbf{f_i} = (f_i(x_1), \dots ,f_i(x_n))$, $i = 1, \dots, m$
- Lập ma trận các hệ số của phương trình $\mathbf{AX = b}$ với

  $\mathbf{a_{ij}} = (\mathbf{f_i, f_j})$; $i, j = 1, \dots, m$ và $\mathbf{b = a_{i,\, m+1}} = (\mathbf{f_i, y})$, $i = 1, \dots, m$
- Giải hệ $\mathbf{AX = b}$ ta được bộ tham số $(c_1^*, c_2^*, \dots, c_m^*)$ tốt nhất của hàm $f(c_1, c_2,\dots,c_m; x)$ đã cho

```
…
for (i = 1; i <= m; i++)
     for (j = 1; j <= m+1; j++)
          {
           t = 0. ;
           for (k = 1; k <=n; k++) t += f[i][k] * f[j][k];
           a[i][j] = t;
          }
for (i = 1; i <= m; i++)
          {
           t = 0. ;
           for (k = 1; k <= n; k++)  t += f[i][k]*y[k];
           a[i][m+1] = t;
          }
Gauss(m, a, c);
…
```

3.6 BÀI TẬP CHƯƠNG 3

1> Từ bảng các logarit chúng ta nhận được các giá trị sau đây của log x tại các điểm ấn định của bảng

x	log x
1,0	0.0
1,5	0,17069
2,0	0,30103
3,0	0,47712
3,5	0,54407
4,0	0,60206

nội suy các giá trị sau đây: log2,5 và log1,25 sử dụng đa thức nội suy Newton.

2> Đối với bảng dữ liệu

x	–2	–1	0	1	2
y	–4	1	1	2	10

Tính các giá trị của đa thức nội suy Newton $P_5(-0,5)$ và $P_5(1,5)$

3> Tạo bảng sai phân cho dữ liệu dưới đây, và đánh giá bậc của đa thức nội suy Newton cần thiết để sinh ra các giá trị nội suy đúng cho số chữ số có nghĩa đã biết.

X	f(x)
1	1,5709
2	1,5713
3	1,5719
4	1,5727
5	1,5738
6	1,5751
7	1,5767
8	1,5785
9	1,5805

4> Đối với bảng dữ liệu dưới đây, sử dụng đa thức ghép trơn spline bậc ba S(x) để tính nội suy tại các điểm x=10 và x=35

x	5	15	30	50	70
y	1,201	1,824	2,624	2,787	2,210

5> Tìm các hệ số a*, b*, c* sao cho hàm $f(x) = ax^2+bx+c$ khớp tốt nhất đối với bảng dữ liệu cho trong bài tập 4 theo tiêu chuẩn bình phương bé nhất.

4 CHƯƠNG 4: TÍNH ĐẠO HÀM VÀ TÍCH PHÂN XÁC ĐỊNH

4.1 TÍNH GẦN ĐÚNG ĐẠO HÀM

4.1.1 Xấp xỉ giá trị đạo hàm dựa vào bảng sai phân

Với các mốc quan sát $x_k = x_0 + k.h$, h là hằng số, ta có các gía trị quan sát tương ứng $f_k = f(x_k)$, $k = -m, ..., n$. Các giá trị của cột Δf_k trong bảng số gia hữu hạn.được dùng để ước lượng đạo hàm bậc nhất, các giá trị của các cột $\Delta^i f_k$ được dùng để ước lượng các đạo hàm bậc i tương ứng của hàm f(x) tại x_k.

BẢNG SAI PHÂN HỮU HẠN

	f_k	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$	$\Delta^4 f_k$	...
x_{-1}	$\boxed{f_{-1}}$					
		$\boxed{\Delta f_{-1}}$				
x_0	$\boxed{f_0}$		$\boxed{\Delta^2 f_{-1}}$			
		$\boxed{\Delta f_0}$		$\boxed{\Delta^3 f_{-1}}$		
x_1	f_1		$\boxed{\Delta^2 f_0}$		$\boxed{\Delta^4 f_{-1}}$	
		Δf_1		$\boxed{\Delta^3 f_0}$		
x_2	f_2		$\Delta^2 f_1$		$\boxed{\Delta^4 f_0}$	
		Δf_2		$\Delta^3 f_1$		
x_3	f_3		$\Delta^2 f_2$			
		Δf_3				
x_4	f_4					
...						

Dựa vào định nghĩa

$$f'(x) = \lim_{\Delta x \to 0} \frac{\Delta f}{\Delta x}$$

ta có 2 giá trị gần đúng sau đây với Δx là hằng số dương:

$$f'(x_k) \approx \frac{\Delta f_k}{\Delta x}, \quad f'(x_k) \approx \frac{\Delta f_{k-1}}{\Delta x}$$

lấy trung bình cộng của 2 công thức trên ta có công thức xấp xỉ:

$$\boxed{f'(x_k) \approx \frac{\Delta f_k + \Delta f_{k-1}}{2\Delta x}} \qquad (4.1)$$

với sai số là:

$$\varepsilon_1 = \frac{\Delta x^2}{6}\left|f^{(3)}(\xi)\right|, x - \Delta x < \xi < x + \Delta x \qquad (4.2)$$

Các xấp xỉ đạo hàm cấp cao được tính tương tự như công thức (4.1)

Ví dụ: Tính gần đúng $f'(1,06)$ từ bảng số liệu sau đây với $\Delta x = 0,01$

$\mathbf{x_k}$	1,03	1,04	1,05	1,06	1,07	1,08	1,09	1,10

f_k	0,012837	0,017033	0,021189	0,025306	0,029384	0,033424	0,037426	0,041393

Theo công thức (4.1):

$$f'(1{,}06) \approx \frac{\Delta f(1{,}05) + \Delta f(1{,}06)}{2.0{,}01} = \frac{0{,}004117 + 0{,}004078}{0{,}02} = 0.40975$$

4.1.2 Xấp xỉ đạo hàm bằng công thức nội suy

Ta có thể sử dụng một trong các công thức nội suy của chương III, thay f(x) bằng đa thức nội suy của nó: $f(x) = P(x) + R_n(x)$. Khi đó

$$f'(x) = P'(x) + R_n'(x)$$

Nhận xét: Phép tính gần đúng đạo hàm như trên không chính xác bằng phép nội suy trực tiếp vì từ hệ thức $f(x_i) = P(x_i)$ nói chung không suy ra được $f'(x_i) = P'(x_i)$

- ➢ *Trường hợp các mốc nội suy cách đều*
  - Công thức nội suy Newton tiến: Sử dụng công thức

$$f(x) \approx f_0 + \frac{\Delta f_0}{1!}u + \frac{\Delta^2 f_0}{2!}u(u-1) + \ldots + \frac{\Delta^N f_0}{N!}u(u-1)\ldots(u-N+1)$$

với $u = (x-x_0)/h$, $f_0 = f(x_0)$ để tính gần đúng giá trị của đạo hàm $f'(x)$ khi $x = x_0$:

$$f'(x_0) \approx \left[\frac{\Delta f_0}{1!}u + \frac{\Delta^2 f_0}{2!}u(u-1) + \ldots + \frac{\Delta^N f_0}{N!}u(u-1)\ldots(u-N+1)\right]\Bigg|_u' u_x'(x_0) \qquad (4.3)$$

Sau một số biến đổi ta được $\boxed{f'(x_0) \approx \frac{1}{h}\sum_{k=1}^{N}\frac{(-1)^{k-1}\Delta^k f_0}{k}}$ (4.4)

với sai số là

$$r_n(x_0) = R_n'(x_0) = \frac{h^N\left|f^{(N+1)}(\xi)\right|}{N+1} \approx \frac{h^N\left|\Delta^{N+1}f_0\right|}{(N+1)h^{N+1}} = \frac{\left|\Delta^{N+1}f_0\right|}{(N+1)h}, x < \xi < x_0 \qquad (4.5)$$

Ví dụ: Tính gần đúng giá trị của f '(1,06) theo công thức nội suy Newton tiến từ bảng số liệu trong ví dụ trên

Ở đây ta cần tính giá trị đạo hàm tại x = 1,06 vì vậy ta có thể chọn $x_0 = 1{,}06$ và lập bảng số gia hữu hạn tiến tương ứng

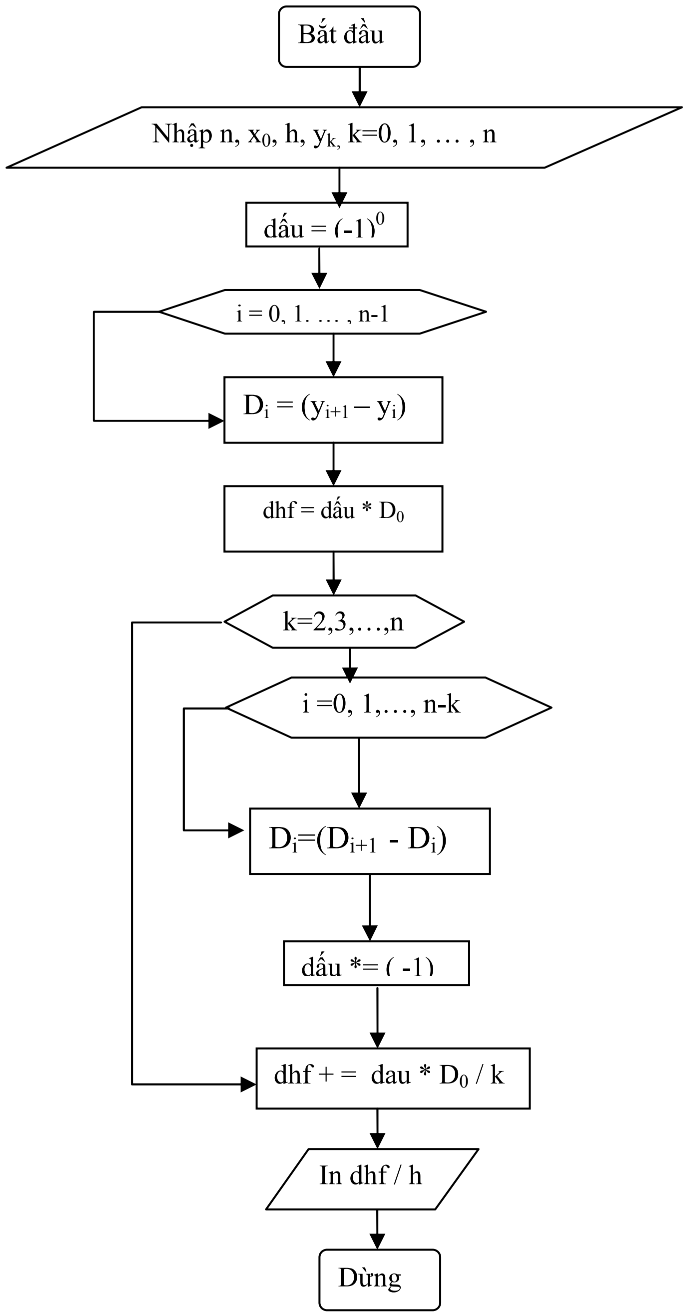
$x_k = 1{,}06 + 0{,}01k$	f_k	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$	$\Delta^4 f_k$
$x_0 = 1{,}06$	$\boxed{0{,}025306}$				
		$\boxed{0{,}004078}$			
$x_1 = 1{,}07$	0,029384		$\boxed{-0{,}000038}$		
		0,004040		$\boxed{0}$	
$x_2 = 1{,}08$	0,033424		–0,000038		$\boxed{0{,}000003}$
		0,004002		0,000003	
$x_3 = 1{,}09$	0,037426		–0,000035		
		0,003967			
$x_4 = 1{,}10$	0,041393				

Áp dụng công thức (4.4) với N = 4, h = 0,01 ta tìm được

$$f'(1{,}06) \approx \frac{1}{h}\left[\frac{\Delta f_0}{1} - \frac{\Delta^2 f_0}{2} + \frac{\Delta^3 f_0}{3} - \frac{\Delta^4 f_0}{4}\right]$$

$$= (1/0{,}01).(0{,}004078 + 0{,}000038/2 + 0 - 0{,}000003/4) = 0{,}4096$$

- Công thức nội suy Newton lùi: cách tính hoàn toàn tương tự
- Sơ đồ khối và chương trình tính đạo hàm bằng công thức nội suy Newton tiến




```
…
dau = (-1)*( -1);
```

```
for (i = 0 ; i <= n-1 ; i++) D[i] = y[i+1] - y[i];
dhf = dau * D[0];
for (k = 2 ; k <= n; k++)
      {
          for (i = 0 ; i <= n -k ; i++) D[i] = D[i+1] - D[i];
          printf("\nSai phan cap %d = %8.6f",k,D[0]);
          dau *= (-1);
          dhf += dau * D[0] / k;
         }
printf("\nDao ham tai x = %8.6f la f = %8.6f",x[0],dhf/h);
```

…

Nhận xét: Thuật toán có độ phức tạp thời gian cỡ $O(n^2)$

➢ *Trường hợp các mốc nội suy không cách đều – Phương pháp Spline bậc ba*

Dùng Spline bậc ba là là một trong những phương pháp tiện lợi nhất hiện nay. Áp dụng công thức

$$\boxed{S_i''(x) = \frac{p_{i-1}}{h_i}(x_i - x) + \frac{p_i}{h_i}(x - x_{i-1}), h_i = x_i - x_{i-1}, \forall i = \overline{1,n}} \qquad (4.6)$$

Lấy tích phân xác định hàm $S_i''(x)$ trên $[x_{i-1}, x_i]$ ta được

$$\boxed{S_i'(x) = -\frac{p_{i-1}}{2h_i}(x_i - x)^2 + \frac{p_i}{2h_i}(x - x_{i-1})^2 - \frac{1}{h_i}(y_{i-1} - \frac{p_{i-1} h_i^2}{6}) + \frac{1}{h_i}(y_i - \frac{p_i h_i^2}{6})} \qquad (4.7)$$

ta tính được $f'(x) \approx S_i'(x)$, $f''(x) \approx S_i''(x)$ khi $x_{i-1} \le x \le x_i$

Ví dụ: Cho bảng các giá trị của hàm f(x):
Hãy tính các hệ số p_i của Spline bậc ba trên các đoạn đã cho rồi tính gần đúng f ’(6,46)

x	1.0	2.0	4.0	5.0	8.0	10.0
y	2.0	1.5	1.25	1.2	1.125	1.1

Giải: Các hệ số p_i là nghiệm của hệ 3 đường chéo sau

$$\left(\begin{array}{cccc|c} 6 & 2 & 0 & 0 & 2,250 \\ 2 & 6 & 1 & 0 & 0,450 \\ 0 & 1 & 8 & 3 & 0,150 \\ 0 & 0 & 3 & 10 & 0,075 \end{array}\right)$$

Giải hệ trên ta tìm được p_1=0,39541; p_2=–0,06123; p_3=0,02658; p_4 –0,00047(p_0=p_5 =0)

Do x = 6,46 thuộc vào đoạn thứ i = 4, cụ thể là $x \in [5, 8]$ nên $p_{i-1} = p_3 = 0,02658$ và $p_i = p_4 = -0,00047$. Thay vào (4.7) ta có:

$$f'(6,46) \approx S_4'(6,46) =$$

$$= -\frac{p_3}{2h_4}(x_4 - x)^2 + \frac{p_4}{2h_4}(x - x_3)^2 - \frac{1}{h_4}(y_3 - \frac{p_3\, h_4^2}{6}) + \frac{1}{h_4}(y_4 - \frac{p_4\, h_4^2}{6})$$

$$= -\frac{0,02658}{2.3}(8 - 6,46)^2 + \frac{-0,00047}{2.3}(6,46 - 5)^2 - \frac{1}{3}(1,2 - \frac{0,026589}{6}) + \frac{1}{3}(1,125 - \frac{-0,000479}{6})$$

$$= \frac{-0,02658}{6}2,3716 - \frac{0,00047}{6}2,1316 - \frac{1}{3}(1,2 - \frac{0,23922}{6}) + \frac{1}{3}(1,125 + \frac{0,00423}{6})$$

$$= -0,01051 - 0,00017 - 0,38671 + 0,37524 = -0,02215$$

Nhận xét:

- So với kết quả lấy đạo hàm của f(x) = 1 + 1/x, f ’(6,46) = – 0,02396 chỉ sai khác 0,2%.
- Chương trình tính Spline bậc ba chỉ cần bổ sung thêm công thức (4.7) và (4.6) để tính gần đúng đạo hàm cấp một và cấp hai của f(x) tại x bất kì trên đoạn $[x_0, x_n]$

4.2 TÍNH GẦN ĐÚNG TÍCH PHÂN XÁC ĐỊNH

Trong nhiều trường hợp ta cần tính giá trị của tích phân xác định $I = \int_a^b f(x)dx$ nhưng không thể biểu diễn nguyên hàm dưới dạng hàm sơ cấp hoặc dù có thể tìm nguyên hàm dưới dạng hàm sơ cấp nhưng đó là biểu thức phức tạp. Khi đó ta phải áp dụng các cách tính gần đúng giá trị của tích phân xác định mà các công thức ban đầu thường được thiết lập từ ý nghĩa hình học của tích phân xác định.

4.2.1 Lập công thức chung sử dụng đa thức nội suy Newton tiến

Cho f(x) khả tích trên [a, b], hãy tính gần đúng tích phân I

- *Lập công thức*: Chia [a, b] thành n đoạn nhỏ đều nhau bởi các điểm chia

$a = x_0 < x_1 < x_2 < \dots < x_n = b$ ta tính được các giá trị $y_i = f(x_i)$ tương ứng tại các mốc nội suy: $f_0, f_1, f_2, \dots, f_n$. và lập được bảng số gia hữu hạn:

$x_k = x_0 + k.h$	f_k	$\square f_k$	$\square^2 f_k$	$\square^3 f_k$	$\square^4 f_k$	...
x_{-1}	f_{-1}					
		$\square f_{-1}$				
x_0	$\boxed{f_0}$		$\square^2 f_{-1}$			
		$\square\boxed{f_0}$		$\square^3 f_{-1}$		
x_1	f_1		$\boxed{\square^2 f_0}$		$\square^4 f_{-1}$	
		$\square f_1$		$\boxed{\square^3 f_0}$		
x_2	f_2		$\square^2 f_1$		$\boxed{\square^4 f_0}$	
		$\square f_2$		$\square^3 f_1$		
x_3	f_3		$\square^2 f_2$			
		$\square f_3$				
x_4	f_4					

Sử dụng đa thức nội suy Newton tiến với h = (b-a) / n, $u = (x - x_0) / h \Rightarrow x = x_0 + uh$:

$$f(x) = f(x_0 + uh) \approx f_0 + \frac{\Delta f_0}{1!}u + \frac{\Delta^2 f_0}{2!}u(u-1) + \frac{\Delta^3 f_0}{3!}u(u-1)(u-2)$$

$$+ \ldots + \frac{\Delta^n f_0}{n!}u(u-1)\ldots(u-n+1)$$

$$\int_a^b f(x)dx = \int_{x_0}^{x_n} f(x)dx \approx h\int_0^n \left[f_0 + \frac{\Delta f_0}{1!}u + \frac{\Delta^2 f_0}{2!}u(u-1) + \frac{\Delta^3 f_0}{3!}u(u-1)(u-2) \right.$$

$$\left. + \ldots + \frac{\Delta^n f_0}{n!}u(u-1)\ldots(u-n+1) \right] du$$

$$= h\left[nf_0 + \frac{\Delta f_0 n^2}{2.1!} + \frac{\Delta^2 f_0}{2!}\int_0^n u(u-1)du + \frac{\Delta^3 f_0}{3!}\int_0^n u(u-1)(u-2)du \right.$$

$$\left. + \ldots + \frac{\Delta^n f_0}{n!}\int_0^n u(u-1)\ldots(u-n+1)du \right] \quad (4.8)$$

➢ *Công thức hình thang*

- Khi n=1: Thay vào (4.8) ta có công thức nội suy Newton tuyến tính

$$\boxed{\int_a^b f(x)dx \approx h\left[f_0 + \frac{\Delta f_0}{2.1!} \right] = h\,\frac{f_0 + f_1}{2}} \quad (4.9)$$

Về mặt hình học, vế phải của (4.9) là diện tích hình thang vuông có đường cao h và hai đáy f_0, f_1

- Khi n tùy ý: ta tách tích phân ban đầu thành tổng của n tích phân trên các đoạn con. Mỗi tích phân này được tính gần đúng bằng công thức hình thang (4.9)

$$\boxed{\int_a^b f(x)dx = \sum_{i=1}^{n}\int_{x_{i-1}}^{x_i} f(x)dx \approx h\sum_{i=1}^{n}\frac{f_{i-1} + f_i}{2} = \frac{h}{2}[f_0 + 2f_1 + \ldots 2f_{n-1} + f_n]} \quad (4.10)$$

Công thức (4.10) là công thức hình thang tổng quát

Sai số: Nếu hàm f(x) có đạo hàm cấp hai liên tục trên đoạn [a, b] thì sai số của công thức (4.10) là

$$\varepsilon = \frac{h^2|f''(\xi)|}{12}(b-a), a < \xi < b \quad (4.11)$$

Ví dụ:

Dùng công thức hình thang tổng quát, tính gần đúng

$$I = \int_0^1 \frac{dx}{1+x}$$

với các điểm chia $x_k = 0,1 \times k$, k = 0, ... , 10 và tìm sai số.

i	x_i	y_i
0	0	1
1	0,1	0.90909
2	0,2	0,83333
3	0,3	0,76923
4	0,4	0,71429
5	0,5	0,66667
6	0,6	0,62500
7	0,7	0,58824
8	0,8	0,55556
9	0,9	0,52632
10	1	0,5

Giải: n = 10, a = 0, b = 1, h = 1/10 = 0,1 các giá trị được tính theo bảng bên:

Theo công thức (4.10) ta có:

$$I = \int_0^1 \frac{dx}{1+x}$$

$$\approx \left[\frac{1+0{,}5}{2} + 0{,}90909 + 0{,}83333 + 0{,}76923 + 0{,}71429 + 0{,}66667 + 0{,}625 + 0{,}58824 + 0{,}55556 + 0{,}52632\right]0{,}1$$

$$= 0{,}69377$$

Sai số ε được đánh giá từ

$$f''(x) = \frac{2}{(1+x)^3} \Rightarrow \max_{0<x<1} |f''(x)| = 2 \Rightarrow \varepsilon \le \frac{2.0{,}1^2}{12} = 0{,}00167 \approx 0{,}002$$

➢ *Chương trình tính*:

```
#include <stdio.h>
#include <conio.h>
#include <math.h>

double f(float x)
{
      double y;
      y = 1/(1+x) ;
      return y ;
}

void main()
{
 int i , n;
 double h, a, b, s;
 clrscr();
 printf("TICH PHAN BANG PHUONG PHAP HINH THANG CUA HAM
             y=1/(1+x)\n");
 printf("Nhap cac can a va b: ");
 scanf("%lf %lf",&a,&b);
 printf("Nhap so cac khoang chia n: ");
 scanf("%d",&n);
 s = (f(a) + f(b)) / 2 ;
 h = (b-a) / n;
 for (i = 1 ; i <= n-1 ;i++)    s += f(a+i*h) ;
 printf("TICH PHAN I= %10.3lf tren [%lf,%lf] voi n=%d\n",s*h,a,b,n);
 getch();
}
```

Nhận xét: Thuật toán có độ phức tạp thời gian tuyến tính O(n)

➢ *Công thức Simpson (công thức parabol)*

- Khi n = 2: Thay vào (4.8) ta có công thức nội suy Newton với ba mốc cách đều:

$$I = \int_a^b f(x)dx \approx h[nf_0 + \frac{\Delta f_0 n^2}{2.1!} + \frac{\Delta^2 f_0}{2!}\int_0^2 u(u-1)du] = \frac{h}{3}(f_0 + 4f_1 + f_2) \qquad (4.12)$$

Về mặt hình học, vế phải của (4.12) là diện tích của hình thang cong mà cạnh cong là cung parabol đi qua ba điểm $A(x_0, y_0)$, $B(x_1, y_1)$ và $C(x_2, y_2)$ của đường cong $y = f(x)$ nên (4.12) vừa được gọi là công thức Simpson vừa được gọi là công thức parabol.

- Khi n = 2k với k tùy ý: ta tách tích phân ban đầu thành tổng của n tích phân trên các đoạn con chứa ba mốc nội suy liên tiếp(x_0, x_1, x_2), …,(x_{2i-2}, x_{2i-1}, x_{2i}), …, (x_{2n-2}, x_{2n-1}, x_{2n}). Mỗi tích phân này được tính gần đúng bằng công thức Simpson (4.12)

$$\boxed{\begin{aligned}\int_a^b f(x)dx &= \sum_{i=1}^{k}\int_{x_{2i-2}}^{x_{2i}} f(x)dx \approx \frac{h}{3}\sum_{i=1}^{k}(f_{2i-2} + 4f_{2i-1} + f_{2i}) \\ &= \frac{h}{3}[(f_0 + f_{2k}) + 4(f_1 + f_3 + ... + f_{2k-1}) + 2(f_2 + f_4 + ... + f_{2k-2})]\end{aligned}} \quad (4.13)$$

Công thức (4.13) là công thức Simpson tổng quát

Sai số: Nếu hàm f(x) có đạo hàm cấp bốn liên tục trên đoạn [a, b] thì sai số của công thức (4.13) là

$$\varepsilon = \frac{h^4\left|f^{(4)}(\xi)\right|}{180}(b-a), a < \xi < b \qquad \text{với } h = \frac{b-a}{2k} \quad (4.14)$$

Ví dụ: Dùng công thức Simpson tổng quát, tính

$$I = \int_0^1 \frac{dx}{1+x}$$

với các điểm chia $x_i = 0{,}1 \times i$, i = 0,..., 10 và tìm sai số.

Giải: với n = 2k = 10, a = 0, b = 1, h = 1/10 = 0,1

các giá trị được tính theo bảng bên.

i	x_i	y_{2i-1}	y_{2i}
0	0		
1	0,1	0.90909	
2	0,2		0,83333
3	0,3	0,76923	
4	0,4		0,71429
5	0,5	0,66667	
6	0,6		0,62500
7	0,7	0,58824	
8	0,8		0,55556
9	0,9	0,52632	
10	1		
	∑	3,45995	2,72818

Theo công thức (4.13) ta có:

$$I = \int_0^1 \frac{dx}{1+x} \approx \frac{0{,}1}{3}[1 + 0{,}0467 + 4.3{,}45955 + 2.2{,}272818] = 0{,}69315$$

Sai số ε được đánh giá từ (4.14)

$$f^{(4)}(x) = \frac{24}{(1+x)^5} \Rightarrow \max_{0<x<1} |f^{(4)}(x)| = 24 \Rightarrow \varepsilon \le \frac{24.0{,}1^4}{180} = 0{,}000013$$

Chương trình tính: lập trình tương tự như đối với công thức hình thang

Nhận xét:

- Từ (4.11) và (4.14) ta thấy *công thức Simpson có độ chính xác cao hơn công thức hình thang với cùng số điểm chia n*
- Việc đánh giá sai số của (4.10) và (4.13) phụ thuộc vào f(x) và gặp khó khăn khi chỉ biết f(x) dưới dạng bảng

4.2.2 Quy tắc làm tăng độ chính xác của việc tính tích phân

Đặt vấn đề: Cần tìm số điểm chia [a,b] để đảm bảo độ chính xác cho trước hoặc với số lần lặp đủ lớn. Muốn vậy ta dựa vào quy tắc như sau

Tiến hành tính gần đúng các tích phân

$$S_j \approx \int_a^b f(x)dx$$

với các bước chia lần lượt là $h_j = (b-a) / 2^j$, j = 1,2,... bằng thuật toán lặp với điều kiện dừng ở bước lặp thứ j là

- với $\varepsilon > 0$ cho trước $|S_j - S_{j-1}| < \varepsilon$ (4.15)
- hoặc j = M (số lần lặp cho trước)

Như vậy công thức hình thang được tính lặp

$$ST_j = \frac{h_j}{2}[f_0 + 2f_1 + ... + 2f_{2^j-1} + f_{2^j}] \tag{4.16}$$

và công thức Simpson được tính lặp

$$SS_j = \frac{h_j}{3}[f_0 + f_{2^j} + 4(f_1 + f_3 + ... + f_{2^j-1}) + 2(f_2 + f_4 + ... + f_{2^j-2})] \tag{4.17}$$

Ví dụ **:**

Với ε = 0,001 và số lần lặp tối đa M = 10 hãy tính gần đúng giá trị của tích phân sau

$$I = \int_{-1}^{1} \sqrt[3]{2x^4 + 1}\, dx$$

theo quy tắc hình thang .

Chương trình sau đây cho giá trị gần đúng I = 0,693 đạt độ chính xác đã cho sau 4 lần lặp

```
void main()
{
int i , j, n, lanlap;
double h, a, b, epsilon, st, ss;
clrscr();
printf("TICH PHAN BANG PP HINH THANG HAM y=1/(1+x)\n");
printf("Nhap can duoi a va can tren b: ");
scanf("%lf %lf",&a,&b);
printf("Nhap so epsilon: "); scanf("%lf",&epsilon);
printf("Nhap so lan lap: "); scanf("%d",&lanlap);
```

```
ss = (f(a) + f(b)) * (b - a) / 2 ;
j = 1; n = 1;
do
      {
       st = ss;
       n *= 2;
       h = (b-a) / n;
       ss = (f(a) + f(b)) / 2 ;
       for (i = 1 ; i <= n-1 ; i++)  ss += f(a+i*h);
       ss *= h;
       j ++;
      } while ((fabs(st - ss) > epsilon) && (j <= lanlap));
printf("I = %10.6lf tren [%lf,%lf] voi n=%d \n",ss,a,b,n);
getch();
}
```

4.3 BÀI TẬP CHƯƠNG 4

1> Hãy sử dụng công thức Simpson tổng hợp tính các tích phân sau đây với h=0,1

$$\int_0^1 e^x dx,\quad \int_1^2 \frac{1}{(3x-2)^2}dx,\quad \int_0^1 \frac{x^{1/7}}{1+x^2}dx$$

2> Hãy sử dụng công thức hình thang tổng hợp để tính tích phân sau với bước h=0,1

$$\int_0^1 \frac{1}{4+\sin(20x)}dx$$

Bước h này đã đủ nhỏ để kết quả tính toán đạt được độ chính xác đến 3 chữ số thập phân?

3> Phải chia khoảng [0; 1] ít nhất thành bao nhiêu phần bằng nhau để việc tính tích phân trong bài tập 4.1 bằng công thức Simpson tổng hợp đạt được độ chính xác đến 4 chữ số thập phân?

Tính tích phân đó với n tìm được.

4> Viết chương trình tính gần đúng tích phân xác định bằng công thức Simpson tổng hợp có tự động làm tăng độ chính xác theo yêu cầu. Kiểm tra chương trình bằng cách tính các tích phân sau với độ chính xác 10^{-6}

$$\int_0^1 e^x dx,\quad \int_1^2 \frac{1}{(3x-2)^2}dx,\quad \int_0^1 \frac{x^{1/7}}{1+x^2}dx$$

5 CHƯƠNG 5: GIẢI PHƯƠNG TRÌNH f(x) = 0

5.1 ĐẶT VẤN ĐỀ

5.1.1 Bài toán

Cho hàm f(x) liên tục trên đoạn [a, b] hoặc trên khoảng vô hạn và phương trình

$$f(x) = 0 \tag{5.1}$$

chỉ có nghiệm thực cô lập, tức là với mỗi nghiệm thực tồn tại một lân cận không chứa các nghiệm thực khác của phương trình

5.1.2 Các bước giải

Việc tính gần đúng nghiệm thực của (5.1) được tiến hành theo hai bước:

1. Tách nghiệm hay tìm khoảng cách li nghiệm (a, b) chỉ chứa một nghiệm của (5.1)
2. Kiện toàn nghiệm: tính gần đúng nghiệm với độ chính xác cho trước

5.1.3 Tách nghiệm

➢ *Cơ sở của phương pháp tách nghiệm*

Hệ quả của định lí Lagrange : Nếu hàm f(x) xác định và liên tục trên [a, b], f(a)f(b) < 0 và có f '(x) giữ dấu không đổi trên (a, b) thì tồn tại duy nhất một nghiệm thực x* □ (a, b) của phương trình (5.1).

➢ *Các phương pháp tách nghiệm*

- Phương pháp giải tích: lập bảng xét dấu của đạo hàm cấp một f '(x) rồi tìm các khoảng (a, b) thỏa mãn hệ quả trên

  Ví dụ: Tìm các khoảng chứa các nghiệm cô lập của phương trình

  $f(x) = x^3 - 6x + 2 = 0$

  Giải: $f'(x) = 3(x^2 - 2)$ có bảng xét dấu sau

x	-∞	-3	-√2	-1	0	1	√2	3	+∞
f '(x)	+	+	-	-	-	-	+	+	
f(x)	-	-7	7,66	7	2	-3	-3,66	11	+

Vậy phương trình trên có ba nghiệm x_1□(-3, -√2), x_2□(0, 1) và x_3□(√2, 3)

- Phương pháp hình học

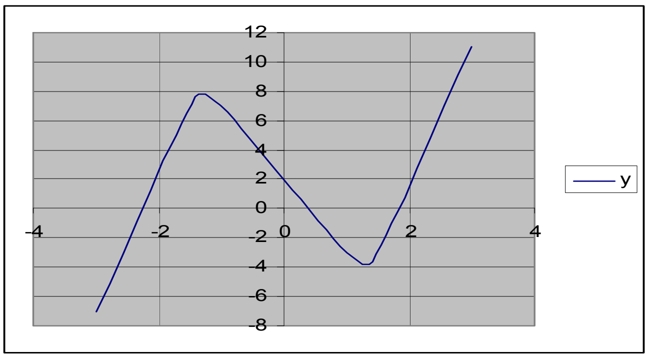
Vẽ đồ thị hàm y = f(x), từ đó tìm ra các khoảng cách li (a, b) chứa các giao điểm của đồ thị với trục hoành

Ví dụ:

Tìm các khoảng chứa các nghiệm cô lập của phương trình

$$f(x) = x^3 - 6x + 2 = 0$$

Vẽ đồ thị hàm $y = x^3 - 6x + 2$



Nhìn vào đồ thị ta thấy đường cong cắt trục hoành tai ba điểm phân biệt trên 3 khoảng rời nhau: (–4, –1), (0, 1) và (1, 3)

5.2 CÁC PHƯƠNG PHÁP KIỆN TOÀN NGHIỆM

5.2.1 Phương pháp chia đôi

➢ *Thuật toán*:

Giả sử tách được khoảng (a, b) chứa nghiệm x*. Chia đôi (a, b) rồi xét dấu của f((a+b)/2)

- Nếu f((a+b)/2) = 0 thì x* = (a+b)/2 là nghiệm đúng của phương trình f(x) = 0 (5.2)
- Nếu f((a+b)/2) ≠ 0 thì x_1 = (a+b)/2 là nghiệm gần đúng ở lần lặp thứ nhất. Chọn khoảng (a_1, b_1) chứa nghiệm có một trong hai đầu mút là (a+b)/2 còn đầu mút kia là a hay b

Lặp lại quá trình trên đến khi tìm được nghiệm đúng x* hoặc nghiệm gân đúng x_i đạt được độ chính xác mong muốn.

➢ *Sự hội tụ và sai số:*

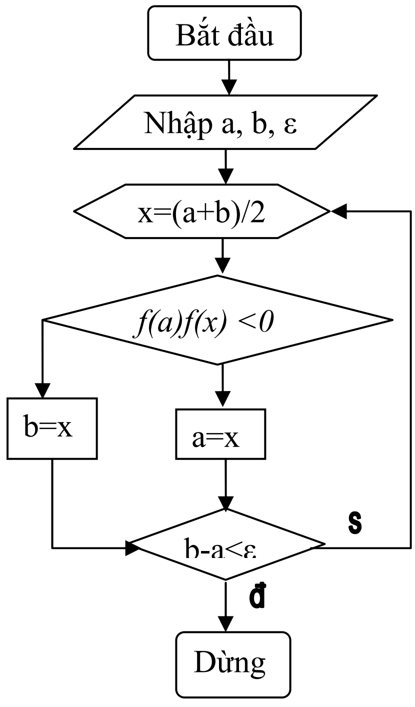
Sử dụng phương pháp chia đôi liên tiếp ta nhận được dãy khoảng lồng nhau {(a_i, b_i)} hữu hạn nếu x* là điểm giữa của khoảng thứ i hay vô hạn nhỏ dần:

$$a_i < x^* < b_i,\ f(a_i).f(b_i) < 0,\ b_i - a_i = (b - a)/2^i \tag{5.3}$$

Khi $i \to \infty$, do sự liên tục của f(x) nên lim b_i = lim a_i = x*

$$\varepsilon = \left|x^* - x_i\right| < \left|\frac{a_i + b_i}{2} - x_i\right| \le \frac{b_i - a_i}{2} = \frac{b-a}{2^{i+1}} \tag{5.4}$$

➢ *Sơ đồ khối*



5.2.2 Phương pháp lặp đơn

➢ *Thuật toán* Giả sử tách được khoảng (a, b) chứa nghiệm x*

Biến đổi tương đương phương trình $f(x) = 0 \Leftrightarrow x = \varphi(x)$ (5.5)

Chọn x_0 □ (a, b) bất kì rồi tính các xâp xỉ liên tiếp nhờ phép lặp $x_{i+1} = \varphi(x_i)$ (5.6)

Nếu dãy nghiệm gần đúng $\{x_i\}$ hội tụ về điểm ξ thì đó chính là nghiệm x* của (5.5)

➢ *Điều kiện hội tụ và sai số*

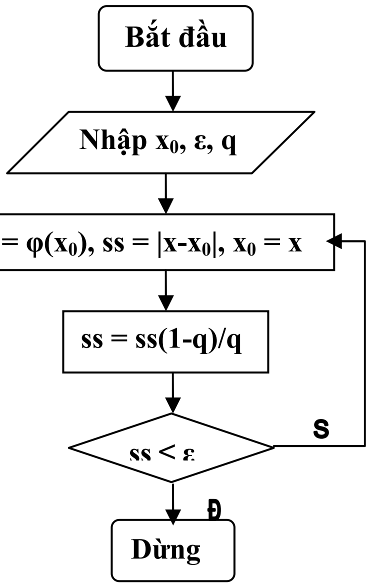
Định lí: Nếu φ(x) khả vi liên tục trên [a, b], φ(x) ∈ [a, b] và $|\omega'(x)| \le q < 1$ thì:

Phương trình (5.5) có nghiệm duy nhất x* □ (a, b)

Phép lặp (5.6) hội tụ về x*

$$|x_i - x^*| \le \frac{q}{1-q}|x_i - x_{i-1}| \text{ hay } |x \ - x^*| \le \frac{q^i}{1-q}|x_1 - x_0| \quad (5.7)$$

➢ *Sơ đồ khối*



Ví dụ: Giải phương trình $f(x) = x^3 - 6x + 2 = 0$ trên (0; 1) bằng phương pháp chia đôi và lặp đơn với độ chính xác ε = 0,01

i	Chia đôi	Lặp đơn
	x_i	$x_{i+1} = \varphi(x_i) = \frac{x_i^3}{6} + \frac{1}{3}$ $, \lvert\varphi'(x_i)\rvert = \left\lvert\frac{x^2}{2}\right\rvert \le \frac{1}{2} < 1, x_0 = 0{,}5$
0	0,5	0,354167
1	0,25	0,340737
2	0,375	0,339927
3	0,3125	0,339820
4	0,34375	0,339877
5	0,328125	0,339877
6	0,335938	

Nhận xét: Phương pháp chia đôi đơn giản nhưng hội tụ chậm hơn phương pháp lặp đơn

Dưới đây ta luôn luôn giả thiết rằng f(x) khả vi liên tục cấp hai trên [a, b] và f′(x), f″x) *giữ dấu không đổi* trên đó (không giảm tổng quát, ta coi f″x) > 0 nếu không ta xét phương trình g(x) = 0, với g(x) = –f(x)

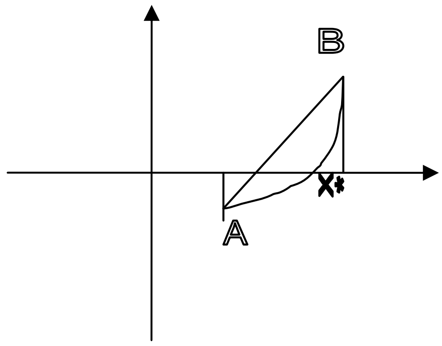
5.2.3 Phương pháp dây cung

➢ *Thuật toán:* Giả sử f''(x) > 0 trên [a, b]

Thay cung đường cong trên [a, b] bởi dây trương cung ấy và xem hoành độ giao điểm của dây cung với trục hoành là nghiệm gần đúng của phương trình (1).

- *Trường hợp 1: f'(x) > 0*

Khi đó f(a) < 0, f(b) > 0, đường cong y = f(x) là lõm trên [a, b]. Phương trình dây trương cung qua hai điểm A(a, f(a)) và B(b, f(b)) là



$$\frac{y-f(a)}{f(b)-f(a)}=\frac{x-a}{b-a}$$

Tại giao điểm với trục hoành (x, 0) ta có

$$\frac{0-f(a)}{f(b)-f(a)}=\frac{x-a}{b-a}\Rightarrow x=a-\frac{f(a).(b-a)}{f(b)-f(a)}\in(a,b)$$

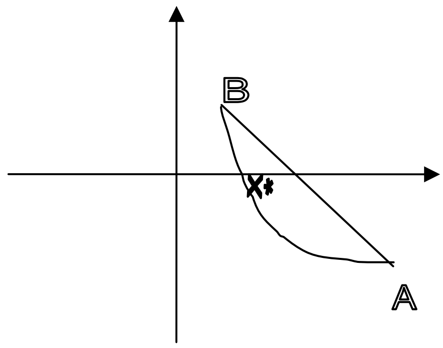
Thay a mới bởi x vừa tìm được. Quá trình lặp

lại cho đến khi x đạt độ chính xác cho trước

$$\boxed{x_{i+1}=x_i-\frac{f(x_i)(b-x_i)}{f(b)-f(x_i)}} \qquad (5.8)$$

- *Trường hợp 2: f'(x) < 0*

Khi đó f(b) < 0, f(a) > 0, đường cong y = f(x) là lồi trên [a, b]. Phương trình dây trương cung qua hai điểm A(a, f(a)) và B(b, f(b)) là



$$\frac{y-f(b)}{f(b)-f(a)}=\frac{x-b}{b-a}$$

Tại giao điểm với trục hoành (x, 0) ta có

$$\frac{0-f(b)}{f(b)-f(a)}=\frac{x-b}{b-a}\Rightarrow x=b-\frac{f(b).(b-a)}{f(b)-f(a)}\in(a,b)$$

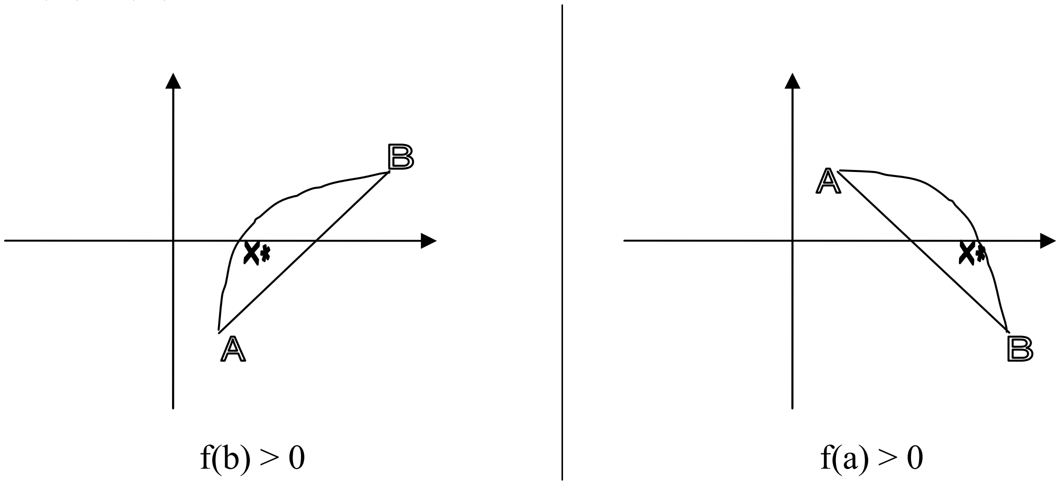
Thay b mới bởi x vừa tìm được. Quá trình lặp lại cho đến khi x đạt độ chính xác cho trước

$$\boxed{x_{i+1}=x_i-\frac{f(x_i)(x_i-a)}{f(x_i)-f(a)}} \qquad (5.9)$$

Nhận xét:

Công thức (5.8) và (5.9) còn đúng khi f″(x) < 0

Nghiệm xấp xỉ ban đầu x_0 là một trong hai điểm đầu mút a hoặc b được chọn theo qui tắc: $f(x_0).f''(x_0) < 0$



f(b) > 0 f(a) > 0

➢ *Sự hội tụ và sai số:*

- Dễ thấy dãy $\{x_i\}$ trong (5.8) hay (5.9) là đơn điệu và bị chặn bởi x* nên luôn có giới hạn.ξ Chuyển qua giới hạn hai công thức này ta thấy giá trị giới hạn ξ thoả mãn phương trình (5.1), mà trên [a, b] nó chỉ có duy nhất nghiệm, vậy ξ ≡ x*.
- Sử dụng các giả thiết đã cho về hàm f(x) và công thức số gia hữu hạn ta nhận được đánh giá độ chính xác ở bước lặp thứ i:

$$\boxed{\ell_i = |x_{i+1} - x^*| \le \frac{M - m}{m}|x_{i+1} - x_i|} \qquad (5.10)$$

trong đó $0 < m \le |f'(x)| \le M$ với mọi x ε [a, b]

Ví dụ: Tìm nghiệm dương của phương trình $x^3 - 0{,}2x^2 - 0{,}2x - 1{,}2 = 0$ trên (1,1; 1,4) với độ chính xác ε = 0,01
Giải: do f'(x) và f''(x) > 0 trên (1,1; 1,5) nên ta chọn xấp xỉ ban đầu x_0 =1,1 (= a) rồi lặp theo (5.8)
Với m = f'(1,1) =2,99 và M = f'(1,4) = 5,12 ta ước lượng được sai số sau mỗi bước lặp theo (5.10)

i	x_i	$f(x_i) = x_i^3 - 0{,}2x_i^2 - 0{,}2x_i - 1{,}2$	$\|x_i - x_{i-1}\|$	ε_i
0	1,1	–0,08254		
1	1,18254	–0,06252	0,08254	0,058802
2	1,19709	–0,01056	0,01455	0,010362
3	1,19952	–0,00175	0,00243	0,001730

Vậy x_2=1,19952 là nghiệm gần đúng phải tìm (nghiệm đúng là 1,2).

5.2.4 Phương pháp tiếp tuyến (Newton)

➢ *Thuật toán*: Với các giả thết về f(x) như ở phương pháp dây cung, ta chọn nghiệm xấp xỉ ban đầu x_0 là một trong hai điểm đầu mút a hoặc b theo qui tắc

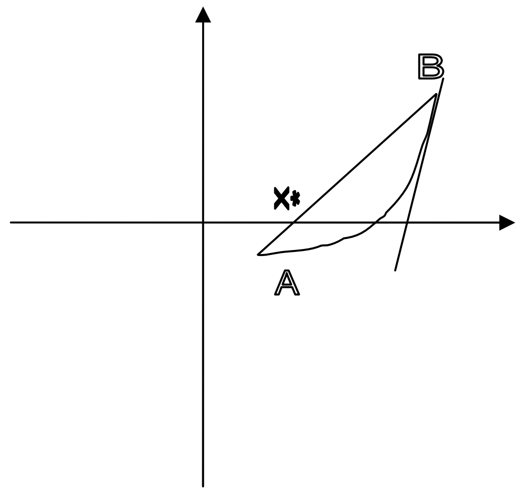
$$f(x_0).f''(x_0) > 0$$

rồi viết phương trình tiếp tuyến của đường cong y = f(x) tại điểm $M_0(x_0, f(x_0)$):

$$y = f'(x_0)(x - x_0) + f(x_0)$$

Hoành độ giao điểm của tiếp tuyến với trục hoành là

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$



$f''(x_0) > 0$, $f(b) > 0$

Tiến hành lặp theo công thức

$$\boxed{x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}} \qquad (5.11)$$

đến khi nghiệm gần đúng đạt được độ chính xác đã cho.

➢ *Sự hội tụ và sai số:*

- Sử dụng các giả thiết đã cho về hàm f(x) trên [a, b] và nhận xét rằng dãy $\{x_i\}$ trong (5.11) là dãy đơn điệu nên tồn tai giới hạn
$\lim_{i\to\infty} x_i = \xi$.
Rõ ràng ξ là nghiệm của (5.1), do tính duy nhất nghiệm trên (a, b) nên $\xi \equiv x^*$.
- Sử dụng các giả thiết đã cho về hàm f(x) và công thức khai triển Taylo ta nhận được đánh giá độ chính xác

$$\boxed{|x_{i+1} - x^*| \le \frac{M_2}{2m_1}|x_{i+1} - x_i|^2} \qquad (5.12)$$

trong đó $0 < m_1 \le |f'(x)|$ và $|f''(x)| \le M_2$ với mọi x ε [a, b]

Ví dụ: Tìm nghiệm gần đúng của phương trình $x^3 - 0,2x^2 - 0,2x - 1,2 = 0$ trên (1,1;1,4) bằng phương pháp tiếp tuyến với sai số ε = 0,01

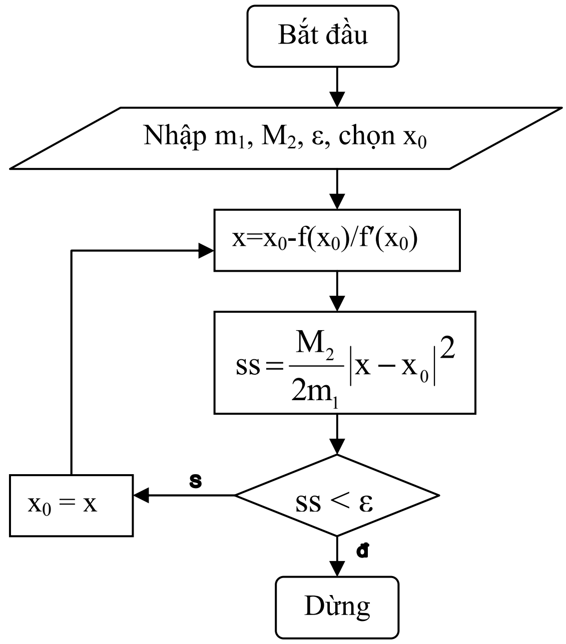
Giải: do f(1,4) cùng dấu với f″(x) nên ta chọn x_0 = 1,4 rồi tiến hành lặp theo (5.11) với m_1 = f(1,1) = 2,99 và $M_2 = |f''(1,4)| = 8$:

i	x_i	$\lvert x_i - x_{i-1}\rvert$	ε_i	$f(x_i)$	$f'(x_i)$	$f(x_i)/f'(x_i)$
0	1,4			0,872	5,12	0,17031
1	1,22969	0,17031	0,03880	0,11109	3,84454	0,02890
2	1,20079	0,02890	0,00112			

Dừng ở bước lặp thứ hai ta có $\varepsilon_2 = |x_2 - x^*| \le \frac{8}{5,98}|1,22969 - 1,20078|^2 \le 0,00112 < 0,01$

Vậy *phương pháp tiếp tuyến có tốc độ hội tụ nhanh hơn phương pháp dây cung.*

- *Sơ đồ khối và chương trình của phương pháp tiếp tuyến*



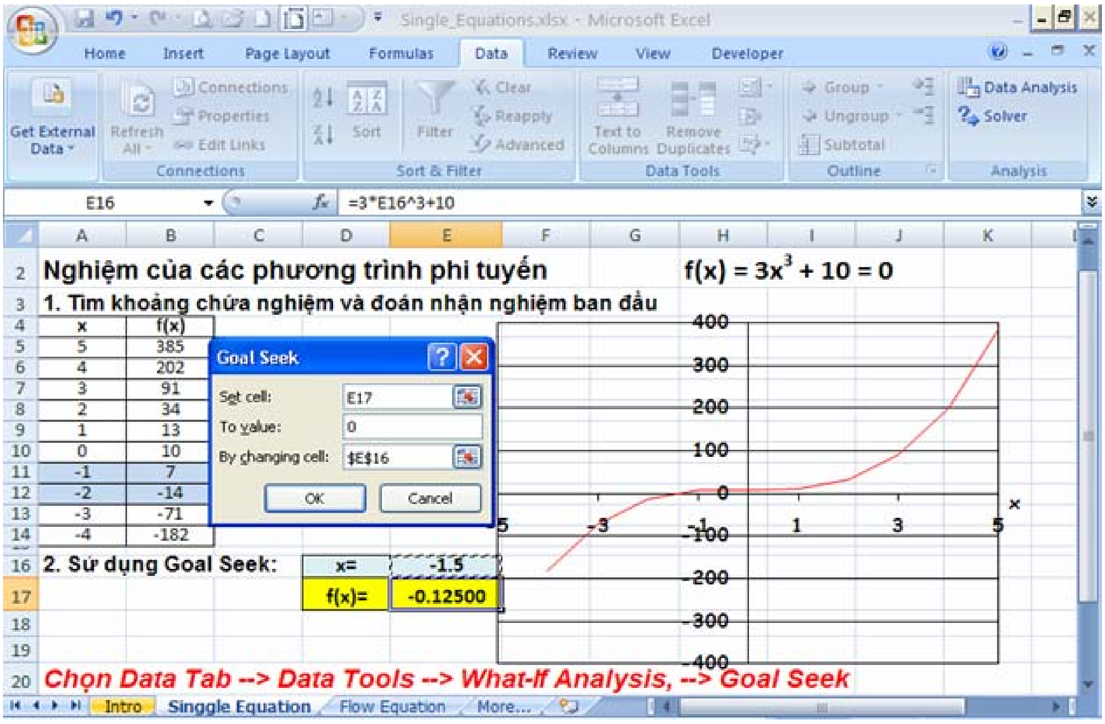

```
...
if (f(a) = = 0) printf("NGHIEM PHUONG TRINH x = %f\n" , a);
if (f(b) = = 0) printf("NGHIEM PHUONG TRINH x = %f\n" , b);
if (f(a)*f(b) > 0)  printf("f(a) va f(b) cung dau\n");
else
    {
     if(f(a)*D2f(a) > 0) x0 = a;
     else x0 = b;
     sbl = 1 ;
     printf("CAC KET QUA TRUNG GIAN\n");
     do
           {
             x = x0 - f(x0) / Df(x0);
             printf("x[%d] = %f\n" , sbl , x);
             ss = (M / (2 * m)) * (x - x0) * (x - x0);
             x0 = x ;
             sbl++ ;
           } while (ss > epsilon) ;
      printf("NGHIEM PHUONG TRINH x = %f\n " , x);
    }
...
```

Nhận xét:

- Độ phức tạp thời gian của thuật toán là O(n).
- Phương pháp tiếp tuyến là trường hợp riêng của phương pháp lặp đơn.

➢ *Sử dụng GOAL SEEK trong EXCEL tìm nghiệm của phương trình f(x) = 0*

Chọn ô chứa phương trình ràng buộc cho hộp **Set cell** và ô chứa nghiệm, cho hộp **By changing cell rồi** nhập giá trị cho vế phải của phương trình



5.3 GIẢI HỆ PHƯƠNG TRÌNH PHI TUYẾN

5.3.1 Lập công thức:

Cho hệ phi tuyến

$$\begin{cases} f_1(x_1,...,x_n)=0 \\ f_2(x_1,...,x_n)=0 \\ .... \\ f_n(x_1,...,x_n)=0 \end{cases} \tag{5.13}$$

trong đó các hàm $f_i(x_1, ... , x_n)$, i = 1, ... , n có các đạo hàm riêng cấp 2 liên tục và bị chặn.

Giả sử $X^{(0)} = (x_1^{(0)}, ... , x_n^{(0)})$ là nghiệm xấp xỉ ban đầu của hệ (5.13). Khai triển Taylor n hàm $f_i(x_1, ... , x_n)$, i = 1, ..., n tại điểm $X(x_1, x_2, ... , x_n)$ thuộc lân cận điểm $X^{(0)}$ và bỏ qua các số hạng bậc hai trở đi, ta có hệ:

$$\begin{cases} f_1(x_1,...,x_n) \approx f_1(x_1^{(0)},...,x_n^{(0)}) + \sum_{j=1}^{n} \dfrac{\partial f_1(x_1^{(0)},...,x_n^{(0)})}{\partial x_j}(x_j - x_j^{(0)}) = 0 \\ .... \\ f_n(x_1,...,x_n) \approx f_n(x_1^{(0)},...,x_n^{(0)}) + \sum_{j=1}^{n} \dfrac{\partial f_n(x_1^{(0)},...,x_n^{(0)})}{\partial x_j}(x_j - x_j^{(0)}) = 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} \displaystyle\sum_{j=1}^{n} \frac{\partial f_1(x_1^{(0)},...,x_n^{(0)})}{\partial x_j}(x_j - x_j^{(0)}) = -f_1(x_1^{(0)},...,x_n^{(0)}) \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ \displaystyle\sum_{j=1}^{n} \frac{\partial f_n(x_1^{(0)},...,x_n^{(0)})}{\partial x_j}(x_j - x_j^{(0)}) = -f_n(x_1^{(0)},...,x_n^{(0)}) \end{cases} \quad (5.14)$$

$$\Leftrightarrow \begin{bmatrix} x_1 - x_1^{(0)} \\ \dots\dots\dots \\ x_n - x_n^{(0)} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1(x_1^{(0)},...,x_n^{(0)})}{\partial x_1} & ... & \frac{\partial f_1(x_1^{(0)},...,x_n^{(0)})}{\partial x_n} \\ ... & ... & ... \\ \frac{\partial f_n(x_1^{(0)},...,x_n^{(0)})}{\partial x_1} & ... & \frac{\partial f_n(x_1^{(0)},...,x_n^{(0)})}{\partial x_n} \end{bmatrix}^{-1} \begin{bmatrix} -f_1(x_1^{(0)},...,x_n^{(0)}) \\ ... \\ -f_n(x_1^{(0)},...,x_n^{(0)}) \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} x_1 \\ .... \\ x_n \end{bmatrix} = \begin{bmatrix} x_1^{(0)} \\ .... \\ x_n^{(0)} \end{bmatrix} - \begin{bmatrix} \frac{\partial f_1(x_1^{(0)},...,x_n^{(0)})}{\partial x_1} & ... & \frac{\partial f_1(x_1^{(0)},...,x_n^{(0)})}{\partial x_n} \\ ... & ... & ... \\ \frac{\partial f_n(x_1^{(0)},...,x_n^{(0)})}{\partial x_1} & ... & \frac{\partial f_n(x_1^{(0)},...,x_n^{(0)})}{\partial x_n} \end{bmatrix}^{-1} \begin{bmatrix} f_1(x_1^{(0)},...,x_n^{(0)}) \\ ... \\ f_n(x_1^{(0)},...,x_n^{(0)}) \end{bmatrix} \quad (5.15)$$

Lặp theo (5.15) ta có

$$\begin{bmatrix} x_1^{(k)} \\ .... \\ x_n^{(k)} \end{bmatrix} = \begin{bmatrix} x_1^{(k-1)} \\ .... \\ x_n^{(k-1)} \end{bmatrix} - \begin{bmatrix} \frac{\partial f_1(x_1^{((k-1)},...,x_n^{(k-1)})}{\partial x_1} & ... & \frac{\partial f_1(x_1^{(k-1)},...,x_n^{(k-1)})}{\partial x_n} \\ ... & ... & ... \\ \frac{\partial f_n(x_1^{(k-1)},...,x_n^{(k-1)})}{\partial x_1} & ... & \frac{\partial f_n(x_1^{(k-1)},...,x_n^{(k-1)})}{\partial x_n} \end{bmatrix}^{-1} \begin{bmatrix} f_1(x_1^{(k-1)},...,x_n^{(k-1)}) \\ ... \\ f_n(x_1^{(k-1)},...,x_n^{(k-1)}) \end{bmatrix} \quad (5.16)$$

- Người ta đã chứng minh rằng nếu nghiệm gần đúng $X^{(0)}$ chọn đủ gần nghiệm đúng X^* của hệ (5.13) thì phép lặp (5.16) sẽ hội tụ về X^* khi $k \to \infty$

5.3.2 Các bước giải hệ phi tuyến bằng phương pháp lặp Newton-Raphson

Bước 1: Tính ma trận các đạo hàm riêng

$$J(X) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} \cdots \frac{\partial f_1}{\partial x_n} \\ \dots\dots\dots\dots \\ \frac{\partial f_n}{\partial x_1} \cdots \frac{\partial f_n}{\partial x_n} \end{bmatrix} \quad (5.17)$$

rồi thế $X = X^{(0)}$ để tính $J(X^{(0)})$:

Bước 2: Tính vectơ $F(X^{(0)}) = [f_1(x_1^{(0)}, ... , x_n^{(0)}), ... , f_n(x_1^{(0)}, ... , x_n^{(0)})]$

Bước 3: Thế $X = X^{(0)}$.vào (5.15), giải hệ (5.14)

$$J(X^{(0)})\ Y = -F(X^{(0)})$$

bằng một trong các phương pháp đã biết, tìm được véc tơ $Y^{(1)} = X^{(1)} - X^{(0)}$

Bước 4: Suy ra nghiệm xấp xỉ thứ nhất $X^{(1)} = (x_1^{(1)}, ... , x_n^{(1)}$ của hệ (5.13)

$$X^{(1)} = X^{(0)} + Y^{(1)}$$

Tiếp tục lặp từ bước 1 đến bước 4 theo công thức

$$J(X^{(k-1)})\,[[X^{(k)} - X^{(k-1)}] = -F(X^{(k-1)}) \qquad (5.17')$$

cho đến khi

$$\sum_{i=1}^{n}\left|x_i^{(k)} - x_i^{(k-1)}\right| < \varepsilon \text{ với } \varepsilon > 0 \text{ cho trước} \qquad (5.18)$$

Ví dụ: Giải hệ phương trình

$$\begin{cases} x_1^2 + x_2^2 = 1 \\ x_1^3 - x_2 = 0 \end{cases}$$

với xấp xỉ ban đầu $X^{(0)} = [0,9;\ 0,5]$ và $\varepsilon = 0,1$.

Bước 1: Tính ma trận $J(X^{(0)})$ theo công thức (5.17)

$$J(X) = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 2x_1 & 2x_2 \\ 3x_1^2 & -1 \end{bmatrix}$$

rồi thế $X = X^{(0)}$ để tính $J(X^{(0)})$:

$$\left(J\left(X^{(0)}\right)\right) = \left(\begin{bmatrix} 2.0,9 & 2.0,5 \\ 3.0,9^2 & -1 \end{bmatrix}\right) = \left(\begin{bmatrix} 1,8 & 1 \\ 2,43 & -1 \end{bmatrix}\right)$$

Bước 2: Tính vectơ

$$F\left(X^{(0)}\right) = \begin{bmatrix} f_1(0,9;0,5) \\ f_2(0,9;0,5) \end{bmatrix} = \begin{bmatrix} 0,9^2 + 0,5^2 - 1 \\ 0,9^3 - 0,5 \end{bmatrix} = \begin{bmatrix} 0,06 \\ 0,229 \end{bmatrix}$$

Bước 3: Giải Gauss

$$\left(J\left(X^{(0)}\right)\right)Y = \left[\begin{array}{cc|c} 1,8 & 1 & -0,06 \\ 2,43 & -1 & -o,229 \end{array}\right] \Leftrightarrow \left(\begin{array}{cc|c} 2,43 & -1 & -0,229 \\ 0 & 1,711 & 0,11 \end{array}\right) \Rightarrow Y = \begin{bmatrix} -0,0683 \\ 0,0630 \end{bmatrix}$$

Bước 4: Tìm xấp xỉ thứ nhất $X^{(1)} = (x_1^{(1)}\ \ x_2^{(1)})$ của nghiệm theo công thức (5.16)

$$X^{(1)} = X^{(0)} + Y = \begin{bmatrix} 0,9 \\ 0,5 \end{bmatrix} + \begin{bmatrix} -0,0683 \\ 0,0630 \end{bmatrix} = \begin{bmatrix} 0,831679 \\ 0,562979 \end{bmatrix}$$

Tính sai số theo công thức (5.18)

$$\sum_{i=1}^{2}|y_i| = |-0,0683| + |0,0630| = 0,1313 > 0,1$$

$\Rightarrow$ lặp lại từ bước 1 đến bước 4 với $X^{(1)}$.

Bước 1: Tính ma trận

$$\left(J\left(X^{(1)}\right)\right) = \begin{bmatrix} 1,663 & 1,126 \\ 2,075 & -1 \end{bmatrix}$$

Bước 2: Tính vectơ

$$F\left(X^{(1)}\right)=\begin{bmatrix} f_1(0,831679;\ 0,562979) \\ f_2(0,831679;\ 0,562979) \end{bmatrix}=\begin{bmatrix} 0,831679^2+0,562979^2-1 \\ 0,831679^3-0,562979 \end{bmatrix}=\begin{bmatrix} 0,008634 \\ 0,012284 \end{bmatrix}$$

Bước 3: Giải Gauss

$$\left(J\left(X^{(1)}\right)\right)Y=\left[\begin{array}{cc|c} 1,663 & 1,126 & -0,008634 \\ 2,075 & -1 & -0,012284 \end{array}\right]\Leftrightarrow\left[\begin{array}{cc|c} 2,075 & -1 & -0,012 \\ 0 & 1,928 & 0,001 \end{array}\right]\Rightarrow Y=\begin{bmatrix} -0,0056 \\ 0,0006 \end{bmatrix} B$$

ước 4: Tìm xấp xỉ thứ hai $X^{(2)} = (x_1^{(2)}\ \ x_2^{(2)})$ của nghiệm theo công thức (5.16)

$$X^{(2)}=X^{(1)}+Y=\begin{bmatrix} 0,831679 \\ 0,562979 \end{bmatrix}+\begin{bmatrix} 0,008634 \\ 0,012284 \end{bmatrix}=\begin{bmatrix} 0,826062 \\ 0,563608 \end{bmatrix}$$

Tính sai số theo công thức (5.18)

$\sum_{i=1}^{2}|y_i| = |-0056|+|0,0006|=0,0062 < 0,1 \Rightarrow$ dừng ở bước 2 với $X^{(2)}=\begin{bmatrix} 0,826062 \\ 0,563608 \end{bmatrix}$ là nghiệm gần đúng.

Nhận xét:

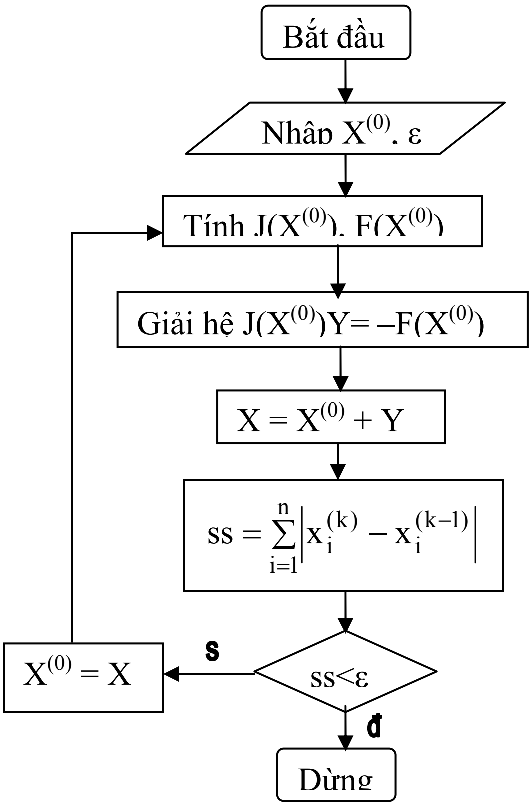
- Khi số phương trình lớn hơn 2, để việc giải lặp hệ (5.17’)

$$J(X^{(k-1)})\ [[X^{(k)} - X^{(k-1)}]\ = -F(X^{(k-1)})$$

bằng máy tính được thuận tiện, người ta viết riêng các chương trình con tính ma trận J(X), véc tơ F(X), và giải hệ phương trình tuyến tính (5.17’) bằng phương pháp khử Gauss – Jordan rồi gọi các chương trình này vào mỗi bước lặp.

- Sự hội tu của phép lặp về nghiệm đúng X* phụ thuộc vào việc chọn nghiệm $X^{(0)}$ ban đầu.

5.3.3 Sơ đồ khối và chương trình




```
#include <stdio.h)

void main ()
{
      float epsilon , ss;
      int i, m, sbl, L ; // so buoc lap L toi da
      mang1 F, x, x0, y;
      matran J;
      clrscr();
      printf("NHAP SO PHUONG TRINH CUA HE m = ");
      scanf("%d",&m);
      printf("NHAP VECTO NGHIEM BAN DAU XO\n");
      for (i = 1; i <= m; i++)
                 {
                 printf("x0[%d] = ", i);
                 scanf("%f",&x0[i]);
                 }
      printf("SO BUOC LAP = ");
      scanf("%d",&L);
      printf("SAI SO epsilon = ");
      scanf("%f",&epsilon);
      sbl = 0;
      do
            {
              Nhap_vecto_F(m, x0, F);
```

```
            Tao_ma_tran_J(m, x0, F, J);
            Gauss(m, J, y);
            printf("\n");
            for(i = 1; i <= m; i++)
              {
                x[i] = x0[i] + y[i];
                printf("x[%d] = %10.6f   ",i, x[i]);
              }
            ss = 0.;
            for (i = 1; i <= m; i++) ss = ss + fabs(y[i]);
            printf("ss = %7.6f",ss);
            sbl++;
            for (i = 1; i <= m; i++) x0[i] = x[i];
        }  while ((ss > epsilon) && (sbl < L));
    printf("\nNGHIEM PHUONG TRINH SAU %d LAN LAP VOI SAI SO
            %7.6f:\n",sbl, ss);
    for(i = 1; i <= m; i++) printf("x[%d] = %10.6f   ",i,x[i]);
}
```

5.4 PHƯƠNG PHÁP LẶP SEIDEL

Bước 1: Đưa hệ (5.13) về hệ tương đương

$$\begin{cases} x_1 = g_1(x_1,..,x_n) \\ ..... \\ x_n = g_n(x_1,..,x_n) \end{cases} \tag{5.19}$$

Bước 2: Kiểm tra điều kiện hội tụ

$$\sum_{j=1}^{n}\left|\frac{\partial g_i}{\partial x_j}\right| < 1, i = \overline{1,n} \tag{5.20}$$

Bước 3: Thay xấp xỉ ban đầu $X^{(0)} = (x_1^{(0)},...,x_n^{(0)})$ của nghiệm vào vế phải của (5.19) để tìm xấp xỉ thứ nhất $X^{(1)} = (x_1^{(1)},...,x_n^{(1)})$

$$\begin{cases} x_1^{(1)} = g\left(x_1^{(0)}, x_2^{(0)},..........., x_n^{(0)}\right) \\ x_2^{(1)} = g_2\left(x_1^{(1)}, x_2^{(0)},.........., x_n^{(0)}\right) \\ ...... \\ x_n^{(1)} = g_n\left(x_1^{(1)}, x_2^{(1)},..., x_{n-1}^{(1)}, x_n^{(0)}\right) \end{cases} \tag{5.21}$$

Như vậy các $x_1^{(1)},...,x_i^{(1)}$ đã tìm được ở các hàng trên được thay vào các hàng dưới để tính $x_{i+1}^{(1)},...,x_n^{(1)}$, i = 1,..., n-1.

Bước 4: Lặp lại bước 2 bằng cách thay $x_i^{(0)}$ bởi $x_i^{(1)}$,thay $x_i^{(1)}$ bởi $x_i^{(2)}$ i = 1,..., n tương ứng để tìm xấp xỉ thứ hai $X^{(2)}=(x_1^{(2)},...,x_n^{(2)})$ cho đến khi thoả mãn độ chính xác

$$\sum_{i=1}^{n}\left|x_i^{(l)} - x_i^{(l-1)}\right| < \varepsilon \text{ cho truoc } < \varepsilon \tag{5.22}$$

hoặc số bước lặp l = M cho trước.

Ví dụ Giải hệ

$$\begin{cases} x_1^3 + \ x_1^2 - 10x_1 - 2x_2^2 + 1 = 0 \\ 2x_1^2 + 3x_2^2 - 20x_2 \qquad\quad - 4 = 0 \end{cases}$$

với ε = 0,03 ; M = 10 và $X^{(0)}$ = (0 0).

Bước 1: Đưa hệ đã cho về hệ tương đương theo công thức (5.19)

$$\begin{cases} x_1 = 0{,}1x_1^3 + 0{,}1x_1^2 - 0{,}2x_2^2 + 0{,}1 \\ x_1 = 0{,}1x_1^2 + 0{,}15x_2^2 - 0{,}2 \end{cases}$$

Bước 2: Kiểm tra điều kiện (5.20) với $0 < x_1 < 1$, $-1 < x_2 < 1$

$$\left|\frac{\partial g_1}{\partial x_1}\right| + \left|\frac{\partial g_1}{\partial x_2}\right| = \left|0{,}3x_1^2 + 0{,}2x_1\right| + \left|-0{,}4x_2\right|$$

$$\le 0{,}5 + 0{,}4 = 0{,}9 < 1$$

$$\left|\frac{\partial g_2}{\partial x_1}\right| + \left|\frac{\partial g_2}{\partial x_2}\right| = \left|0{,}2x_1\right| + \left|0{,}45x_2^2\ \right|$$

$$\le 0{,}2 + 0{,}45 = 0{,}65 < 1$$

Bước 3: Thay xấp xỉ ban đầu $X^{(0)}$ của nghiệm để tìm xấp xỉ thứ nhất $X^{(1)} = (x_1^{(1)}, x_2^{(1)})$ theo công thức (5.21)

$$x_1^{(1)} = 0{,}1.0^3 + 0{,}1.0^2 - 0{,}2.0^2 + 0{,}1 = 0{,}1$$
$$x_2^{(1)} = 0{,}1.0{,}1^2 + 0{,}15.0^2 - 0{,}2 = -\ 0{,}199$$

Tính sai số theo công thức (5.22)

$$\sum_{i=1}^{2}\left|x_i^{(1)} - x_i^{(0)}\right| = \left|\,0{,}1 - 0\,\right| + \left|\,-0{,}199 - 0\,\right|$$

$$= 0{,}299 > 0{,}03$$

Vì vậy ta chuyển sang bước 4.

Bước 4: Thay xấp xỉ thứ nhất $X^{(1)}$ của nghiệm để tìm xấp xỉ thứ hai

$$X^{(2)} = (x_1^{(2)}, x_2^{(2)})$$

theo công thức (5.21):

$$x_1^{(2)} = 0{,}1.0{,}1^3 + 0{,}1.0{,}1^2 - 0{,}2.(-0{,}199)^2 + 0{,}1 = 0{,}09318$$
$$x_2^{(1)} = 0{,}1.0{,}093^2 + 0{,}15.(-0{,}199)^2 - 0{,}2 = -0{,}1854109$$

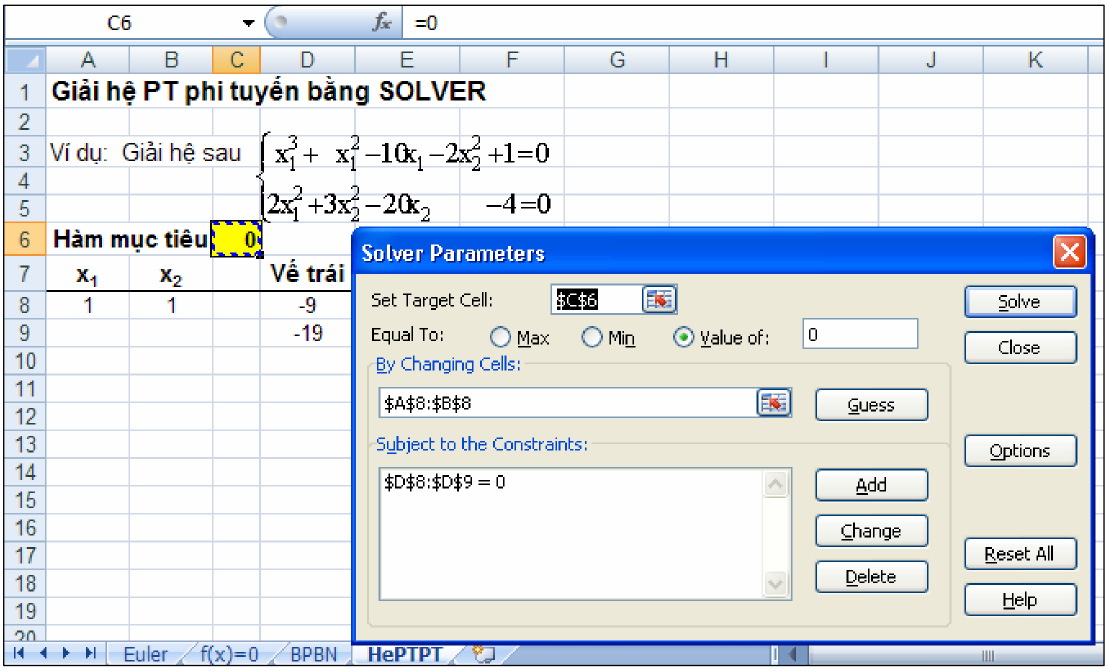
Tính sai số theo công thức (5.22)

$$\sum_{i=1}^{2}\left|x_i^{(2)} - x_i^{(1)}\right| = \left|\,0{,}093 - 0{,}1\,\right| + \left|\,-0{,}185 - (-0{,}199)\right|$$

$$= 0{,}021 < 0{,}03$$

Vậy ta dừng ở xấp xỉ thứ hai của nghiệm.

5.5 Sử dụng Solver trong EXCEL giải hệ phương trình phi tuyến



Nhập các giá trị đoán nhận ban đầu cho $x_1, x_2, \ldots, x_n$

1) Nhập các vế trái của các phương trình $f_1, f_2, \ldots, f_n$ vào các ô riêng được liên kết với giá trị của các ẩn
2) Chọn công cụ Solver (Data Tab → Solver)
   Đặt "Target Cell" cho ô có công thức = 0
   Chọn "Equal to Value of 0"
   Đối với hộp "By Changing Cells," trỏ tới các ô chứa giá trị ban đầu của $x_1, x_2, \ldots, x_n$
   Thêm các ràng buộc (các phương trình)
   Nhấn nút "Solve"

5.6 BÀI TẬP CHƯƠNG 5

1> Muốn dùng phương pháp chia đôi giải phương trình $xe^x - 2 = 0$ trên đoạn [0; 1] với độ chính xác $\varepsilon = 0,001$ thì cần phải lặp bao nhiêu lần?

2> Đa thức $f(x) = x^3 - 2x - 5$ có một nghiệm trong khoảng (2; 3).

(a) Áp dụng bốn bước của phương pháp chia đôi để thu nhỏ khoảng chứa nghiệm với độ rộng còn 1/16.

(b) Tính x_3 và x_4 bằng phương pháp dây cung, bắt đầu với $x_1 = 3$ và $x_2 = 2$.

Đánh giá sai số của các kết quả

(c) Tính x_2, x_3, và x_4 bằng phương pháp Newton với $x_1 = 2$.

Đánh giá sai số của các kết quả.

(d) Tính x_2, x_3, và x_4 bằng phương pháp lặp đơn với $x_1 = 2$.

Đánh giá sai số của các kết quả.

3> Giải hệ phương trình sau bằng phương pháp Newton

$$\begin{cases} f(x,y) = x^2 + xy^3 - 9 = 0 \\ g(x,y) = 3x^2y - y^3 - 4 = 0. \end{cases}$$

6 CHƯƠNG 6: CÁC PHƯƠNG PHÁP SỐ GIẢI PHƯƠNG TRINH VI PHÂN

6.1 ĐẶT VẤN ĐỀ

Nhiều bài toán của khoa học kĩ thuật dẫn đến việc tìm nghiệm của phương trình vi phân (PTVP) thoả mãn một số điều kiện nào đó. Những phương pháp giải đúng chỉ áp dụng được cho một lớp rất hẹp các PTVP. Phần lớn các PTVP mô tả các hệ thống cơ học, vật lí, thuỷ lực, sinh học, … đều không có lời giải đúng. Chương này sẽ nghiên cứu các phương pháp số giải gần đúng hai bài toán sau đây của PTVP.

6.1.1 Bài toán Cauchy (bài toán giá trị đầu)

- Cho PTVP cấp một sau đây:

$$y' = f(x,y), \ (x_0 \le x \le x^*, \text{ hàm f đã biết}) \tag{6.1a}$$

Tìm hàm y(x) thoả mãn các điều kiện

$$y(x_0) = y_0 \ \text{ (vế phải đã biết)} \tag{6.1b}$$

- Tương tự như trên, bài toán Cauchy đối với PTVP cấp n là:

Tìm hàm y(x) thoả mãn các điều kiện

$$y^{(n)} = f(x,y,y',\dots,y^{(n-1)}), \ (x_0 \le x \le x^*, \text{ hàm f đã biết}) \tag{6.2a}$$

$$y(x_0) = y_0,\ y'(x_0) = y'_0,\ \dots,\ y^{(n-1)}(x_0) = y_0^{(n-1)} \ \text{(vế phải đã biết)} \tag{6.2b}$$

- Đối với hệ PTVP cấp một, bài toán Cauchy là:

Tìm các hàm $y_1(x), y_2(x), \dots, y_n(x)$ thỏa mãn các điều kiện:

$$\begin{cases} \dfrac{dy_1}{dx} = f_1(x, y_1, y_2, \dots, y_n) \\ \dfrac{dy_2}{dx} = f_2(x, y_1, y_2, \dots, y_n) \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ \dfrac{dy_n}{dx} = f_n(x, y_1, y_2, \dots, y_n) \end{cases} \tag{6.3a}$$

$$y_1(x_0) = a_1,\ y_2(x_0) = a_2,\ \dots,\ y_n(x_0) = a_n, \tag{6.3b}$$

Nhận xét: Bằng cách đặt $y_1(x) = y'$, $y_2 = y''$, … , $y_{n-1} = y^{(n-1)}$ ta đưa được PTVP cấp n (6.2a) về hệ PTVP cấp một tương đương: Tìm các hàm $y(x), y_1(x), y_2(x), \dots, y_{n-1}(x)$

thoả mãn hệ PTVP cấp một:

$$\begin{cases} \dfrac{dy}{dx} = y_1(x,y) \\ \dfrac{dy_1}{dx} = y_2(x,y) \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad (x_0 \le x \le x^*) \\ \dfrac{dy_{n-2}}{dx} = y_{n-1}(x,y) \\ \dfrac{dy_{n-1}}{dx} = f(x, y, y_1, \dots, y_{n-1}) \end{cases} \tag{6.4a}$$

thoả mãn các điều kiện đầu: $y(x_0) = a_1,\ y_1(x_0) = a_2,\ \dots,\ y_{n-1}(x_0) = a_n$ (6.4b)

Vì vậy ta chỉ cần giải các bài toán Caushy (6.1a), (6.1b) và (6.3a), (6.3b) là đủ

6.1.2 Bài toán biên hai điểm tuyến tính đối với PTVP cấp hai:

Tìm hàim y(x) thoả mãn các điều kiện

$$y'' + p(x)y' + q(x)y = f(x), \quad (a \le x \le b) \tag{6.5a}$$

$$\alpha_0 y(a) + \alpha_1 y'(a) = A \tag{6.5b}$$

$$\beta_0 y(b) + \beta_1 y'(b) = B \tag{6.5c}$$

6.1.3 Các phương pháp số giải bài toán Cauchy

Trong *các phương pháp số giải bài toán Cauchy (6.1a), (6.1b)*, người ta tìm nghiệm tại các điểm $x_0 < x_1 < \dots < x_n \equiv x^*$. tại mỗi x_k ta tìm giá trị gần đúng y_k của nghiệm phương trình (6.1a) hay (6.2a) $y_k \approx y(x_k)$. Giá trị y_k được tính thông qua p giá trị đứng trước nó:

$$y_k = \Phi(y_{k-1}, \dots, y_{k-p}) \tag{6.6}$$

Phương pháp một bước chỉ tính y_k thông qua y_{k-1}. Sau đây ta sẽ xét các phương pháp một bước thông dụng như phương pháp Euler, phương pháp Euler cải tiến, phương pháp Runge – Kutta.

Tư tưởng chính của các phương pháp này là để tìm nghiệm bằng số y_1 của bài toán (6.1a) và (6.1b) tại điểm $x_1 = x_0 + h$ với bước $h > 0$, thay việc sử dụng khai triển Taylor hàm $y(x_0+h)$ bằng việc *tính giá trị của $f(x, y)$ tại một dãy điểm (x_i, y_i)*:

6.2 CÁC PHƯƠNG PHÁP SỐ GIẢI BÀI TOÁN CAUCHY

Tìm nghiệm bằng số của hàm y(x) thoã mãn các điều kiện

$$y' = f(x,y),\ x_0 \le x \le x^*, \text{ hàm f đã biết)} \tag{6.7a}$$

$$y(x_0) = y_0 \text{ (vế phải đã biết)} \tag{6.7b}$$

6.2.1 Phương pháp Euler

➢ *Lập công thức*

Giả sử cho các mốc $x_k = x_0 + kh$, $k = 0,1, 2, ..., n$; hằng số $h > 0$ được chọn sao cho $x_n = x^*$.

Kí hiệu giá trị gần đúng của nghiệm y(x) tại các mốc x_k là $y_k \approx y(x_k)$

Khai triển Taylor hàm y(x) đến cấp một tai lân cận điểm x_k

$$y(x) = y(x_k) + (x - x_k)y'(x_k) + \frac{(x - x_k)^2}{2!}y''(c), c \in (x_k, x)$$

Thay $x = x_{k+1} = x_k + h$ và $y'(x_k) = f(x_k, y(x_k))$ vào đẳng thức trên ta có

$$y(x_{k+1}) = y(x_k) + hf(x_k, y(x_k)) + \frac{h^2}{2}y''(c) \tag{6.8}$$

Bỏ qua số hạng cuối cùng ở vế phải và thay $y(x_k) \approx y_k$ vào đẳng thức trên ta được

$$\boxed{y_{k+1} \approx y_k + hf(x_k, y_k)} \tag{6.9}$$

Như vậy xuất phát từ y_0 ta có thể lần lượt tìm gia trị gần đúng $y_1, y_2, ..., y_n$

➢ *Sai số*

Từ (6.8) người ta chứng minh được rằng sai số địa phương

$$\boxed{\varepsilon_k = |y(x_k) - y_k| \le \frac{M}{2}h^2 = O(h^2)} \tag{6.10}$$

trong đó $|y''(c)| \leq M$., c □ (x_k, x_{k+1}) và sai số toàn phần $\varepsilon = O(h)$

➢ *Ví dụ*

Dùng phương pháp Euler tìm giá trị gần đúng y(1) từ phương trình vi phân sau đây

$$y' = xy / 2$$

thoả mãn điều kiện đầu

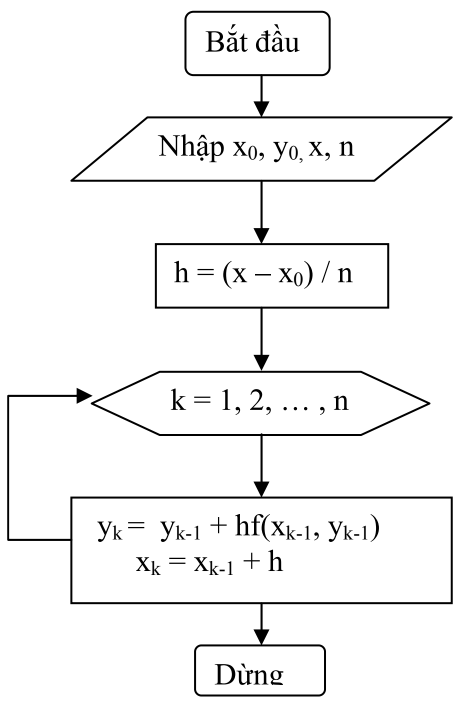
$$y(0) = 1$$

với sai số địa phương cỡ 10^{-2}

Giải: Để đạt được sai số địa phương cỡ 10^{-2} ta chọn bước h = 0,1 hay chia [0, 1] thành 10 khoảng bằng nhau

i	x_i	y_i	$f(x_i, y_i)$	$hf(x_i, y_i)$	**Nghiệm đúng**
0	0	**1**	0	0	1
1	0,1	**1**	0,05	0,00	**1,03**
2	0,2	**1,01**	0,10	0,01	**1,01**
3	0,3	**1,02**	0,15	0,02	**1,02**
4	0,4	**1,03**	0,21	0.02	**1,04**
5	0,5	**1,05**	0,26	0,03	**1,06**
6	0,6	**1,07**	0,32	0,03	**1,09**
7	0,7	**1,11**	0,39	0,04	**1,13**
8	0,8	**1,15**	0,46	0,05	**1,17**
9	0,9	**1,19**	0,58	0,05	**1,22**
10	1	**1,25**			**1,28**

➢ *Sơ đồ khối và chương trình*



```
…
printf("\nSo diem chia n = "); scanf("%d", &n);
printf("Gia tri diem dau x0 = "); scanf("%lf", &x[0]);
printf("Gia tri diem cuoi x* = "); scanf("%lf", &x[n]);
printf("Gia tri ban dau y0 = "); scanf("%lf", &y[0]);
h = (x[n]-x[0]) / n;
for (i=1; i<=n; i++)
 {
  y[i] = y[i-1] + h*f(x[i-1], y[i-1]);
  x[i] = x[i-1] + h;
 }
…
```

Nhận xét:

- Phương pháp Euler đơn giản, dễ lập trình nhưng có độ chính xác thấp.
- Có thể áp dụng phương pháp Euler để giải bài toán Cauchy cho hệ phương trình vi phân cấp một, chẳng hạn với hệ hai phương trình

$$\begin{cases} y' = f(x,y,z) \\ z' = g(x,y,z) \end{cases}$$

và điều kiện đầu $y(x_0) = y_0$; $z(x_0) = z_0$ công thức (6.3a) có dạng sau

$$\begin{cases} y_{k+1} = y_k + hf(x_k, y_k, z_k) \\ z_{k+1} = z_k + hg(x_k, y_k, z_k) \end{cases} \quad k = 1, 2, \ldots, n \qquad (6.11)$$

➢ *Ví dụ:* Giải hệ phương trình sau trên [0; 0,6]

$$\begin{cases} y' = (z-y)x \\ z' = (z+y)x \end{cases}$$

với điều kiện đầu y(0) = z(0) = 1,0000.và h = 0,1

Giải: Áp dụng công thức (6.11) ta tính được bảng sau:

i	**x_i**	**y_i**	**z_i**	**$f(x_i, y_i, z_i)$**	**$hf(x_i, y_i, z_i)$**	**$g(x_i, y_i, z_i)$**	**$hg(x_i, y_i, z_i)$**
0	0	1	1	0	0	0,0	0
1	0,1	1	1	0	0	0,2000	0,0200
2	0,2	1	1,0200	0,0040	0,0004	0,4040	0,0404
3	0,3	1,0004	1,0604	0,0180	0,0018	0,6182	0,0618
4	0,4	1,0022	1,1222	0,0480	0,0048	0,8498	0,0850
5	0,5	1,0070	1,2072	0,1001	0,0100	1,1071	0,1107
6	0,6	**1,0170**	**1,3179**				

6.2.2 Phương pháp Euler cải tiến

➢ *Lập công thức*

Áp dụng công thức Newton – Lebnitz ta có

$$y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} y'(x)dx$$

rồi tính gần đúng tích phân xác định ở vế phải theo công thức hình thang ta nhận được

$$y(x_{k+1}) - y(x_k) = \frac{h}{2}[y'(x_k) + y'(x_{k+1})] - \frac{h^3}{12}y'''(c)$$

Bỏ qua số hạng cuối ở vế phải và thay $y(x_k) \approx y_k$, $y(x_{k+1}) \approx y_{k+1}$, $y'(x_k) \approx f(x_k, y_k)$, $y'(x_{k+1}) \approx f(x_{k+1}, y_{k+1})$ ta nhận được

$$y_{k+1} - y_k = \frac{h}{2}[f(x_k, y_k) + f(x_{k+1}, y_{k+1})] \Leftrightarrow$$

$$y_{k+1} = y_k + \frac{h}{2}[f(x_k, y_k) + f(x_{k+1}, y_{k+1})], i = \overline{0, n} \tag{6.12}$$

➢ *Sai số*

Người ta chứng minh được rằng phương pháp Euler cải tiến có sai số toàn phần $O(h^3)$

Nhận xét: Nhược điểm cơ bản của phương pháp Euler cải tiến là muốn tính y_{k+1} ta phải *giải phương trình (6.12) đối với y_{k+1}*. Để khắc phục điều này, người ta sử dụng phối hợp hai phương pháp trên như sau:

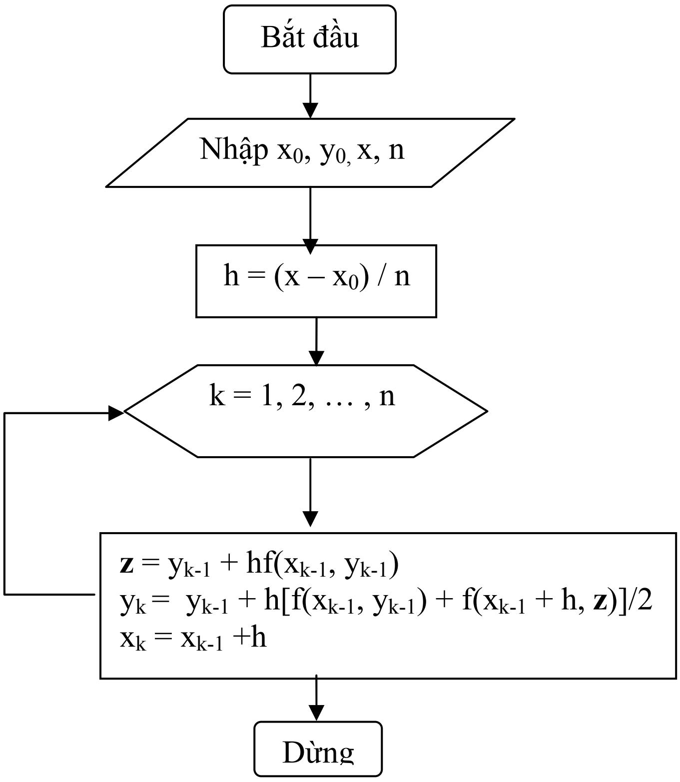
$$\boxed{y_{k+1} = y_k + \frac{h}{2}[f(x_k, y_k) + f(x_k + h, y_k + hf(x_k, y_k))]} \tag{6.13}$$

➢ *Ví dụ:*

Dùng phương pháp Euler cải tiến giải lại bài toán Cauchy trong ví dụ trước ta theo công thức (6.12) ta nhận được bảng sau

i	**x_i**	**y_i**	**$f(x_i, y_i)$**	**$hf(x_i, y_i)$**	**$z_i=y_i+hf(x_i, y_i)$**	**$hf(x_{i+1}, z_i)$**	**Nghiệm đúng**
0	0	**1**	0	0			**1**
1	0,1	**1,0025**	0,0511	0,0051	1	0,0050	**1,0025**
2	0,2	**1,0100**	0,1010	0,0101	1,0075	0,0101	**1,0101**
3	0,3	**1,0227**	0,1534	0,0153	1,0201	0,0153	**1,0228**
4	0,4	**1,0408**	0,2082	0.0208	1,0381	0,0208	**1,0408**
5	0,5	**1,0646**	0,2661	0,0266	1,0616	0,0265	**1,0645**
6	0,6	**1,0943**	0,3283	0,0328	1,0911	0,0327	**1,0942**
7	0,7	**1,1305**	0,3957	0,0396	1,1270	0,0394	**1,1337**
8	0,8	**1,1738**	0,4695	0,0470	1,1698	0,0468	**1,1735**
9	0,9	**1,2248**	0,5512	0,0551	1,2204	0,0549	**1,2245**
10	1	**1,2812**					**1,2840**

➢ *Sơ đồ khối và chương trình tính*




```
…
for (i=1; i<=n; i++)
    {
     z = y[i-1] + h*f(x[i-1], y[i-1]);
     x[i] = x[i-1] + h;
     y[i] = y[i-1] + h*(f(x[i-1], y[i-1]) + f(x[i], z)) / 2;
    }
…
```

6.2.3 Phương pháp Runge-Kutta

➢ *Lập công thức:*

Để giải bài toán Cauchy (6.1a), (6.1b) hai nhà toán học Runge và Kutta đặt

$y(x_0+h) = y(x_0) + r_1k_1(h) + r_2k_2(h) + \ldots + r_sk_s(h) + \varphi_s(h)$

$y(x_0+h) = y_0 + \Delta y_0 + \varphi_s(h)$ (6.14)

trong đó

$\Delta y_0 = r_1k_1(h) + r_2k_2(h) + \ldots + r_sk_s(h)$

$k_i(h) = hf(\xi_i, \square_i)$; $\xi_i = x_0 + a_i h$, $a_1 = 0$

$\square_i = y_0 + b_{i1}k_1(h) + \ldots + b_{i,\,i-1}k_{i-1}(h)$, $i = 1, 2, \ldots, s$

$\varphi_s(h) = y(x_0+h) - y_0 - \Delta y_0$ là sai số (địa phương) của phương pháp. (6.15)

Các hệ số a_i, b_{ij}, r_i được chọn sao cho các đạo hàm cấp i của $\varphi_s(h)$ thoả mãn điều kiện

$\varphi_s^{(i)}(0) = 0\ (i = \overline{0, s})$; $\varphi_s^{(i+1)}(0) \neq 0$ $i = 1, 2, \ldots, s$ với s càng lớn càng tốt

Vậy ta có hệ phương trình phi tuyến xác định các hệ số a_i, b_{ij}, r_i

$r_1k_1^{(i)}(0) + r_2k_2^{(i)}(0) + \ldots + r_sk_s^{(i)}(0) = y_0^{(i)}$, $i = 0, 1, 2, \ldots, s$ (6.16)

Trong tính toán, người ta thường dùng s = 4 và từ hệ các điều kiện (6.16) ta nhận được hệ 11 phương trình, 13 ẩn $a_2, a_3, a_4, b_{21}, b_{31}, b_{32}, b_{41}, b_{42}, b_{43}, r_1, r_2, r_3, r_4$.
Công thức Runge – Kutta bậc 4 thông dụng nhất có dạng:

$$\boxed{y_{i+1} \approx y_i + (k_1 + 2k_2 + 2k_3 + k_4) / 6} \qquad (6.17)$$

trong đó $k_1 = hf(x_i, y_k)$

$k_2 = hf(x_i + h/2, y_i + k_1/2)$

$k_3 = hf(x_i + h/2, y_i + k_2/2)$

$k_4 = hf(x_{i+1}, y_i + k_3)$

với sai số là $0(h^4)$.
Người ta chứng minh được rằng sai số địa phương của công thức là $O(h^4)$

- *Nhận xét*

- Phương pháp Runge – Kutta cho độ chính xác cao hơn phương pháp Euler và phương pháp Euler cải tiến.
- Phương pháp Euler và phương pháp Euler cải tiến.là hai trường hợp riêng của phương pháp Runge – Kutta bậc một và bậc hai tương ứng

- *Ví dụ*

Tính gần đúng các giá trị của y(x) trên [0; 0,4] từ phương trình vi phân y' = x + y thoả mãn điều kiện y(0) = 1 bằng công thức Runge-Kutta bậc bốn với bước h = 0,1.
Giải: Ta có $x_0 = 0$, $x_i = 0{,}1x\ i$, $y(0) = y_0 = 1$. Khi i = 0

$k_1 = hf(x_0, y_0) = 0{,}1(0 + 1) = 0{,}1$

$k_2 = hf(x_0 + h/2, y_0 + k_1/2) = 0{,}1\ (0 + 0{,}05) + (1 + 0{,}05) = 0{,}11$

$k_3 = hf(x_0 + h/2, y_0 + k_2/2) = 0{,}1\ (0 + 0{,}05) + (1 + 0{,}055) = 0{,}1105$

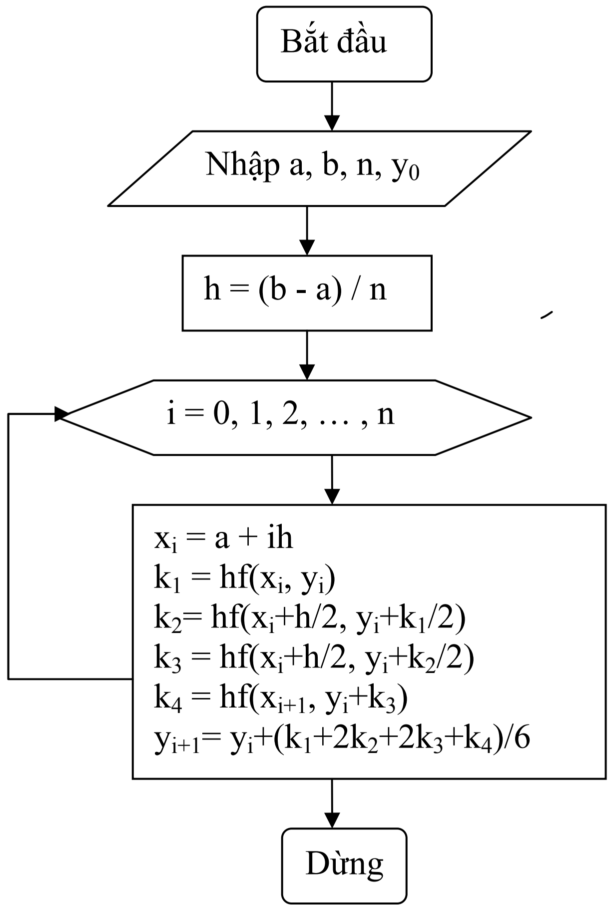
$k_4 = hf(x_1, y_0 + k_3) = 0{,}1\ (0 + 0{,}1) + (1 + 0{,}1105) = 0{,}12105$

Từ đó $\Delta y_0 = (0{,}1 + 2\ 0{,}11 + 2\ 0{,}1105 + 0{,}12105) / 6 = 0{,}1103$
và $y_1 = y_0 + \Delta y_0 = 1 + 0{,}1103 = 1{,}1103$
Khi i = 1, 2, 3 ta tính được y_2, y_3, y_4 hoàn toàn tương tự. Các kết quả ghi ở bảng sau:

i	x	y	k=hf(x,y)= 0,1(x+y)	Δy
0	0	**1**	0,1	0,1
	0,05	1,05	0,11	0,22
	0,05	1,055	0,1105	0,221
	0,1	1,1105	0,1210	0,121
			Nghiệm đúng y(0,1)=1,1103	0,662/6=0,1103
1	0,1	**1,1103**	0,1210	0,1210
	0,15	1,1708	0,1321	0,2642
	0,15	1,1763	0,1326	0,2652
	0,2	1,2429	0,1443	0,1443
			Nghiệm đúng y(0,2)=1,2428	0,7947/6=0,1324

2	0,2	**1,2427**	0,1443	0,1443
	0,25	1,3149	0,1565	0,3130
	0,25	1,3209	0,1571	0,3142
	0,3	1,3998	0,1700	0,1700
			Nghiệm đúng y(0,3)=1,3997	0,9415/6=0,1569
3	0,3	**1,3996**	0,1700	0,1700
	0,35	1,4846	0,1835	0,3670
	0,35	1,4904	0,1840	0,3680
	0,4	1,5836	0,1984	0,1984
			Nghiệm đúng y(0,4)=1,5837	1,1034/6=0,1840
4	0,4	**1,5836**		

➢ *Sơ đồ khối và chương trình trình tính*



…
```
printf("\n i        x              y              delta");
for (i=0; I <= n; i++)
  {
    x[i]    = a + h * i;
    k1   = h * f(x[i], y[i]);
    k2   = h * f(x[i]+h/2, y[i]+k1/2);
    k3   = h * f(x[i]+h/2, y[i]+k2/2);
    k4   = h * f(x[i]+h, y[i]+k3);
    d[i]= (k1 + 2*k2 + 2*k3 + k4) / 6;
    y[i+1]  =  y[i] + d[i];
    printf("\n%2d  %10.4lf    %10.4lf    %10.4lf",i,x[i],y[i],d[i]);
  }
```
…

Nhận xét: Thuật toán có độ phức tạp thời gian tuyến tính

6.2.4 Giải bài toán Cauchy của hệ PTVP cấp một

Các phương pháp trên có thể áp dụng được cho hệ phương trình vi phân cấp một. Chẳng hạn hãy giải bài toán Cauchy sau đây cho hệ hai PTVP

$$\begin{cases} y' = f(x, y, z) \\ z' = g(x, y, z) \end{cases}$$

với điều kiện đầu $y(x_0) = y_0$ và $z(x_o) = z_0$, bằng phương pháp Runge - Kutta

Khi đó công thức (6.11) có dạng

$$\begin{cases} y_{i+1} \approx y_i + (k_1 + 2k_2 + 2k_3 + k_4) / 6 \\ z_{i+1} \approx z_i + (l_1 + 2l_2 + 2l_3 + l_4) / 6 \end{cases} \qquad (6.18)$$

trong đó các hệ số k_i và l_i được tính như sau

$k_1 = hf(x_i, y_i, z_i)$	$l_1 = hg(x_i, y_i, z_i)$
$k_2 = hf(x_i+h/2, y_i+k_1/2, z_i+l_1/2)$	$l_2 = hg(x_i+h/2, y_i+k_1/2, z_k i l_1/2)$
$k_3 = hf(x_i+h/2, y_i+k_2/2, z_i+l_2/2)$	$l_3 = hg(x_i+h/2, y_i+k_2/2, z_i+l_2/2)$
$k_4 = hf(x_{i+1}, y_i+k_3, z_i+l_3)$	$l_4 = hg(x_{i+1}, y_i+k_3, z_i+l_3)$

với sai số địa phương là $0(h^4)$.

Ví dụ: Giải bài toán Cauchy của hệ PTVP cấp một sau đây trên [0; 0,4] với h = 0,1

$$\begin{cases} y' = z \\ z' = -xz - y \end{cases} \quad \text{với điều kiện đầu} \quad \begin{cases} y(0) = 0 \\ z(0) = 1 \end{cases}$$

Kết quả tính được ở bảng sau:

i	x	y	z	k=hf	l = hg	Δy	Δz
0	0	0	1	0,1	0,0	0,1	0
	0,05	0,05	1	0,1	-0,01	0,2	-0,02
	0,05	0,05	0,995	0,0995	-0,00998	0,1990	-0,02
	0,1	0,0995	0,990	0,0990	-0,0149	0,0990	-0,0149
$y_1 = y_0 + \Delta y_0 / 6 = 0,1$ $z_1 = z_0 + \Delta z_0 / 6 = 0,9908$						Δy_0 /6= 0,1	$\Delta z_0 / 6 =$ -0,0092
1	0,1	0,1	0,9908	0,0991	-0,0199	0,0991	-0,0199
	0,15	0,1496	0,9809	0,0981	-0,0248	0,1962	-0,0496
	0,15	0,1491	0,9784	0,0978	-0,0247	0,1956	-0,0494
	0,20	0,1978	0,9661	0,0968	-0,0294	0,0966	-0,0294
$y_2 = y_1 + \Delta y_1 / 6 = 0,1979$ $z_2 = z_1 + \Delta z_1 / 6 = 0,9661$						$\Delta y_1 / 6 =$ 0,0979	$\Delta z_1 / 6 =$ -0,0247
2	0,2	0,1979	0,9661	0,0966	-0,0295	0,0966	-0,0295
	0.25	0,2462	0,9514	0,0951	-0,0341	0,1902	-0,0682
	0.25	0,2455	0,9344	0,0934	-0,0339	0,1868	-0,0678
	0,3	0,2913	0,9322	0,0932	-0,0385	0,0932	-0,0385
$y_3 = y_2 + \Delta y_2 / 6 = 0,2923$ $z_3 = z_2 + \Delta z_2 / 6 = 0,9321$						$\Delta y_2 / 6 =$ 0,0944	$\Delta z_2 / 6 =$ -0,0340
3	0,30	0,2923	0,9321	0,0932	-0,0572	0,0932	-0,0572
	0,35	0,3389	0,9035	0,0904	-0,0655	0,1808	-0,1310
	0,35	0,3841	0,8707	0,0871	-0,0689	0,1742	-0,1378
	0,4	0,4772	0,8018	0,0802	-0,0798	0,0802	-0,0798
$y_4 = y_3 + \Delta y_3 / 6 = 0,3804$ $z_4 = z_3 + \Delta z_3 / 6= 0,8570$						$\Delta y_3 / 6 =$ 0,0881	$\Delta z_3 / 6 =$ -0,0751

6.3 PHƯƠNG PHÁP SAI PHÂN GIẢI BÀI TOÁN BIÊN TUYẾN TÍNH

6.3.1 Xét bài toán biên hai điểm tuyến tính đối với PTVP cấp hai:

Bài toán: Tìm hàm y(x) thoả mãn các điều kiện

$$y'' + p(x)y' + q(x)y = f(x), \quad (a \le x \le b)$$
$$\alpha_0 y(a) + \alpha_1 y'(a) = A$$
$$\beta_0 y(b) + \beta_1 y'(b) = B$$

Giải: Chia đoạn [a, b] thành n phần bằng nhau với bước $h = (b – a) / n$.

Gọi $x_i = a + ih$ là các nút lưới, $i = 0, 1, \dots , n$ và $y_i = y(x_i)$ là giá trị của hàm y(x) tại các nút lưới, rồi khai triển $y(x_i \pm h)$ theo công thức Taylor ta có:

$$y_{i\pm1} = y_i \pm \frac{h}{1!}y_i' + \frac{h^2}{2!}y_i'' \pm \frac{h^3}{3!}y_i^{(3)} + \frac{h^4}{4!}y_i^{(4)} + O(h^4)$$

$$\Longrightarrow \quad y_i'' = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + O(h^2); \; y_i' = \frac{y_{i+1} - y_{i-1}}{2h} + O(h^2);$$

$$y_0' = \frac{y_1 - y_0}{h} + O(h); \; y_n' = \frac{y_{n-1} - y_n}{-h} + O(h)$$

Thay đạo hàm bằng các tỉ sai phân tại các nút lưới, ta được hệ ba đường chéo:

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p_i \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i = f_i \quad (i = \overline{1, n-1}) \tag{6.19a}$$

$$\alpha_0 y_0 + \alpha_1 \frac{y_1 - y_0}{h} = A \tag{6.19b}$$

$$\beta_0 y_n + \beta_1 \frac{y_{n-1} - y_n}{-h} = B \tag{6.19c}$$

Biến đổi tương đương hệ (6.19a) ta được

$$\boxed{d_i y_{i-1} + c_i y_i + y_{i+1} = h^2 e_i} \quad \text{với } i = 1, \ldots, n-1 \tag{6.19a'}$$

trong đó

$$c_i = \frac{2(q_i h^2 - 2)}{2 + p_i h}; \; d_i = \frac{2 - p_i h}{2 + p_i h}; \; e_i = \frac{2f_i}{2 + p_i h}$$

Hệ (6.19a'), (6.19b) và (6.19c) có đủ n+1 phương trình, n+1 ẩn hoàn toàn xác định nên giải được bằng phương pháp khử Gauss-Joordan đã biết ở chương 2.

6.3.2 Ví dụ: Tìm hàm y(x) trên [0; 1] với bước h = 0,1 là nghiệm của PTVP

$$y'' + 2y' + y = 2x$$

thỏa mãn các điều kiện

$$y(0) = 1; \; y(1) = 0$$

Từ đầu bài ta có p(x) = 2, q(x) =1, f(x) = 2x, $\alpha_0 = 1$, $\alpha_1 = 0$, A = 1, $\beta_0 = 1$, $\beta_1 = 0$, B = 0

Vậy $y_0 = 1$, $y_{10} = 0$, $p_i = 2$, $q_i = 1$. Từ đó tính được $c_i = -1{,}80909$, $d_i = 0{,}81818$, $e_i = 0{,}18182i$; $h^2 e_i = 0{,}00182i$, i=1, 2, …, 9

Ta có hệ 9 phương trình, 9 ẩn sau

$$\mathbf{0{,}81818y_{i-1} - 1{,}80909y_i + y_{i+1} = 0{,}00182\,i}, \; i = 1, 2, \ldots, 9$$

hay viết lại dưới dạng hệ 3 đường chéo:

$$\left[\begin{array}{cccccccc|c}
c_1 & 1 & 0 & 0 & \ldots & 0 & 0 & 0 & h^2e_1 - d_1 y_0 \\
d_2 & c_2 & 1 & 0 & \ldots & 0 & 0 & 0 & h^2e_2 \\
0 & d_3 & c_3 & 1 & \ldots & 0 & 0 & 0 & h^2e_3 \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
0 & 0 & 0 & 0 & \ldots & c_7 & 1 & 0 & h^2e_7 \\
0 & 0 & 0 & 0 & \ldots & d_8 & c_8 & 1 & h^2e_8 \\
0 & 0 & 0 & 0 & \ldots & 0 & d_9 & c_9 & h^2e_9
\end{array}\right]$$

ứng với các vectơ

$$\vec{d} = \begin{pmatrix} 0{,}81818 \\ 0{,}81818 \\ 0{,}81818 \\ 0{,}81818 \\ 0{,}81818 \\ 0{,}81818 \\ 0{,}81818 \\ 0{,}81818 \\ 0{,}81818 \end{pmatrix} \quad \vec{c} = \begin{pmatrix} -1{,}80909 \\ -1{,}80909 \\ -1{,}80909 \\ -1{,}80909 \\ -1{,}80909 \\ -1{,}80909 \\ -1{,}80909 \\ -1{,}80909 \\ -1{,}80909 \end{pmatrix} \quad \overrightarrow{\text{vế phải}} = \begin{pmatrix} -0{,}81636 \\ 0{,}00182 \\ 0{,}00364 \\ 0{,}00546 \\ 0{,}00728 \\ 0{,}00910 \\ 0{,}01092 \\ 0{,}01274 \\ 0{,}01456 \end{pmatrix} \quad \vec{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \end{pmatrix} \text{ là vectơ nghiệm}$$

Giải hệ này ta nhận được nghiệm bằng số

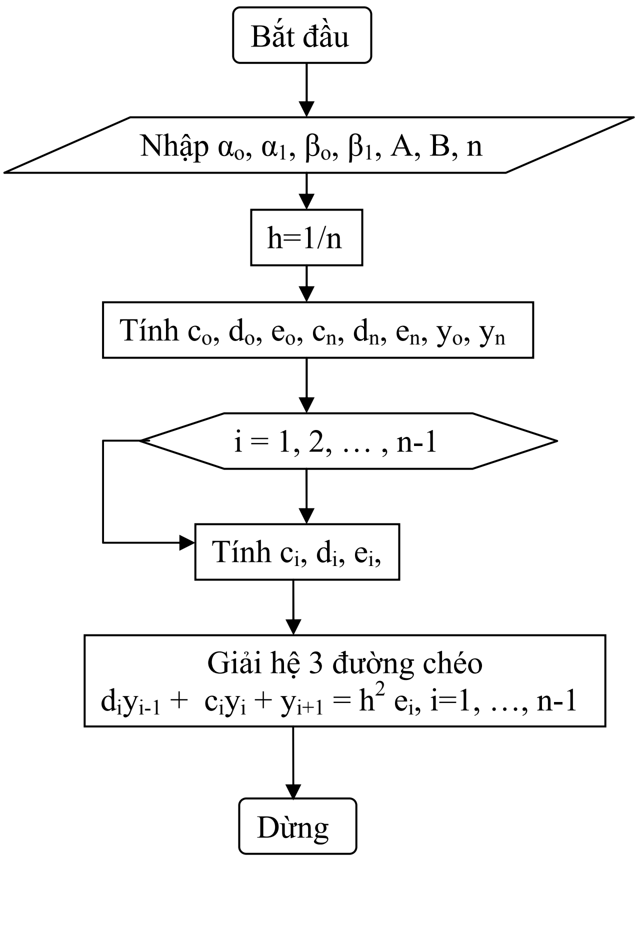
y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9
0,7632	0,5644	0,4002	0,2677	0,1641	0,0870	0,0340	0,0031	-0,0076

So sánh với nghiệm đúng của bài toán biên là $y = 5e^{-x} + 0{,}43656xe^{-x} + 2x - 4$

x	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
y	0,7637	0,5651	0,4011	0,2687	0,1650	0,0878	0,0347	0,0036	-0,0074

thì nghiệm gần đúng chỉ sai khác nghiệm đúng cỡ $O(h^3)$

6.3.3 Sơ đồ khối



6.4 BÀI TẬP CHƯƠNG 6

1> Hãy sử dụng phương pháp Euler giải các bài toán Cauchy sau đây trên [0; 1], sử dụng các bước $h = 0,2$ và $h = 0,1$.

Trong mỗi trường hợp, tính sai số tại $x = 1,0$.

(a) $y' = -y / (x + 1)$ với điều kiện đầu $y(0) = 1$ (nghiệm đúng $y(x) = \frac{1}{1+x}$).

(b) $y' = -y^3 / 2$ với điều kiện đầu $y(0) = 1$ (nghiệm đúng $y(x) = \frac{1}{\sqrt{1+x}}$).

2> Hãy sử dụng phương pháp Euler cải tiến giải các bài toán Cauchy trong bài tập 6.1 trên [0; 1], sử dụng bước $h = 0,2$.

Trong mỗi trường hợp, tính sai số tại $x = 1,0$.

3> Hãy sử dụng phương pháp Runge-Kutta giải các bài toán Cauchy trong bài tập 6.1 trên [0; 1], sử dụng bước $h = 0,5$.

Tính sai số tại $x = 0,5$ và $x = 1,0$.

Các sai số sẽ giảm đi bao nhiêu lần khi h bị giảm đi một nửa?

4> Hãy sử dụng phương pháp Runge-Kutta giải bài toán Cauchy của phương trình vi phân cấp hai

$$y'' = e^y - 1, \; y(0) = 0$$

trên đoạn [0; 1] với bước $h = 0,1$.

5> Giải bài toán biên 2 điểm của phương trình vi phân tuyến tính cấp hai

$$x^2y'' + xy' = 1, \; y(1) = 0, \; y(1,4) = 0,0566$$

tại $x = 1,1; 1,2; 1,3 ; 1,4$.

THÔNG TIN TÁC GIẢ

NGUYỄN THỊ VINH

THÔNG TIN CÁ NHÂN

Ngày tháng năm sinh: 8-3-1955
Giới tính: Nữ
Học vị: Thạc sỹ Khoa học Máy Tính và Toán học, Đại học Victoria, Úc
Chỗ ở: 323/39 Trần Khánh Dư , Hà Nội, Việt Nam
Email: :vinhnt@wru.vn

ĐỊA CHỈ CƠ QUAN

BỘ MÔN KHOA HỌC MÁY TÍNH & CÔNG NGHỆ PHẦN MỀM
KHOA CÔNG NGHỆ THÔNG TIN, ĐẠI HỌC THỦY LỢI
TẦNG 2, NHÀ C5, 175 TÂY SƠN, ĐỐNG ĐA, HÀ NỘI, VIỆT NAM
Phone (+84 4) 3563 7020
Website http://www.wru.edu.vn/

PHẠM VI VÀ ĐỐI TƯỢNG SỬ DỤNG GIÁO TRÌNH GIẢI TÍCH SỐ

Trường Đại học	• Khối Khoa học Kỹ thuật , Công nghệ
Ngành học	• Toán học, Khoa học Máy tính , Kỹ thuật, Công nghệ

KIẾN THỨC CHUẨN BỊ

- **Toán cao cấp**
- **Đại số tuyến tính**
- **Tin Đại Cương**

TỪ KHÓA TRA CỨU

- Sai số
- Hệ phương trình tuyến tính
- Hệ ba đường chéo
- Hệ chéo trội
- Ma trận và định thức
- Phương trình phi tuyến
- Đa thức nội suy
- Làm khớp dữ liệu
- Công thức hình thang tổng hợp
- Công thức Simpson tổng hợp
- Bài toán Cauchy của phương trình vi phân