THUẬT TOÁN PHƯƠNG PHÁP SỐ

TUẦN 1: ĐA THỨC NỘI SUY

Câu 1: Viết thuật toán tìm $n+1$ mốc nội suy tối ưu trong khoảng $[a, b]$ cho trước (viết ra 1 mảng có mốc sắp xếp theo thứ tự tăng dần).

Input: Bậc n của đa thức, 2 mốc a, b.

Output: Mảng $c[]$ chứa $n+1$ mốc nội suy theo thứ tự tăng dần.

1. Khởi tạo mảng $c[]$ gồm $n+1$ chữ số 0.
2. Cho biến chạy $i : 0 \to n$, thực hiện các bước sau:

   2.1: Gán $t_i = \cos\left(\dfrac{2i+1}{2n+2}\pi\right)$

   2.2: Gán $c_i = \dfrac{b-a}{2}t_i + \dfrac{b+a}{2}$
3. Trả về mảng $c[]$.

TUẦN 2: NỘI SUY LAGRANGE

Câu 4a: Viết thuật toán tính $P(c)$

Input: $a[a_0, a_1, \ldots, a_n]$ với a_i là hệ số của bậc $n-i$, n, c.
Mảng $a[a_0, a_1, \ldots, a_n]$ gồm $n+1$ hệ số của đa thức $P(x)$ bậc n theo bậc tương ứng, giá trị mốc nội suy c.

Output: Giá trị $P(c)$.

1. Kiểm tra $n = 0$? Nếu có trả về giá trị a_0. Ngược lại, chuyển tới bước 2.
2. Khởi tạo mảng $b[]$ gồm $n+1$ số 0.
3. Gán $b_n = a_n$.
4. Cho biến chạy $i : n-1 \to 0$, thực hiện gán $b_i = a_i + b_{i-1} \times c$
5. Trả về giá trị b_0.

Câu 4b: Viết thuật toán tính phép chia đa thức $P_n(x)$ cho đa thức bậc nhất dạng $(x-c)$ bằng sơ đồ Horner.

Input: Mảng $a = [a_n, a_{n-1}, \ldots, a_0$ chứa hệ số a_i của đa thức $P_n(x)$ tương ứng với bậc i và giá trị c của đa thức $x - c$.

Output: Mảng $b = [b_{n-1}, b_{n-2}, \ldots, b_0]$ là hệ số của đa thức thương và số dư R.

1. Khởi tạo mảng $b[]$.
2. Gán $b_{n-1} = a_n$.
3. Cho biến chạy $i : n-1 \to 0$, thực hiện gán $b_i = a_{i+1} + c \times b_{i+1}$
4. Tính số dư $R = a_0 + c \times b_0$
5. Trả về mảng hệ số thương b và số dư R.

Câu 4c: Viết thuật toán tính $P^{(k)}(c)$

Input: $a[a_0, a_1, \ldots, a_n]$, n, k, c
Mảng $a[a_0, a_1, \ldots, a_n]$ gồm $n+1$ hệ số của đa thức $P(x)$ bậc n theo bậc tương ứng, đạo hàm cấp k, giá trị mốc nội suy c.

Output: Giá trị $P^{(k)}(c)$.

1. Kiểm tra $k < 0$?
   - Nếu có thì báo lỗi rồi dừng thuật toán.
   - Ngược lại thì chuyển tới bước 2.
2. Kiểm tra $k = 0$?

- Nếu có thì tính $b = horner_{div}(a[], n, c)$ rồi trả về giá trị b_0.
- Ngược lại chuyển tới bước 3.

3. Kiểm tra $n < k$?
   - Nếu có thì trả về giá trị 0 rồi kết thúc thuật toán.
   - Ngược lại gán $b[\dots] = a[\dots]$ chuyển tới bước 4.
4. Cho biến chạy $i : 0 \to k :$, thực hiện tính: $b[] = horner_{div}(b[], i, c)$
5. Trả về giá trị $k! \times b_{n-k}$ rồi kết thúc thuật toán.

Trong đó: gói $horder_{div}(b[], m, c)$:

1. Kiểm tra $len(b) = 0$?
   - Có thì báo lỗi rồi kết thúc thuật toán.
   - Ngược lại, chuyển tới bước 2.
2. Gán $n = len(b) - 1$ *(bậc của đa thức $b[]$).*
3. *Kiểm tra $n = 0$? Nếu có trả về giá trị a_0. Ngược lại, chuyển tới bước 4.*
4. *Khởi tạo mảng $b[]$ gồm $n + 1$ số 0.*
5. *Gán $b_n = a_n$.*
6. *Cho biến chạy $i : n - 1 \to m$, thực hiện gán $b_i = a_i + b_{i+1} \times c$*
7. *Trả về mảng b_m.*

Câu 5a: Viết thuật toán tính phép nhân đa thức $P_n(x)$ với đa thức bậc nhất dạng $x - c$.

Input: Mảng $a = [a_n, a_{n-1}, \dots, a_0$ chứa hệ số a_i của đa thức $P_n(x)$ tương ứng với bậc i và giá trị c của đa thức $x - c$.

Output: Mảng $b = [b_{n+1}, b_n, \dots, b_0]$ chứa các hệ số của đa thức $P_n(x)(x - c)$.

1. Tạo mảng $b = []$.
2. Gán $b_{n+1} = a_n$
3. Cho biến chạy $i : 1 \to n$, thực hiện gán: $b_i = a_{i-1} - a_i \times c$
4. Gán $b_0 = -c \times a_0$
5. Trả về mảng $b[]$.

Câu 5b: Biểu diễn W_{n+1} trong cơ sở chính tắc.
Input: Các mốc nội suy $x[x_0, x_1, \ldots, x_n], n$
Output: Mảng $W_{n+1}(c)$ chứa các hệ số của đa thức $\prod_{i=0}^{n}(x - x_i)$.

1. Tạo mảng $b = [1]$.
2. Cho $c : x_0 \to x_n$, thực hiện:
   2.1 Gán $a[] = b[]$
   2.2 Đặt $m = len(a)$
   2.3 Tạo mảng mới d có độ dài là $m + 1$.
   2.4 Gán $d_0 = 1$
   2.5 Cho biến chạy $i : 1 \to m - 1$, thực hiện gán: $d_i = a_i - a_{i-1} \times c$
   2.6 Gán $d_m = -c \times a_{m-1}$
   2.7 Gán $b[] = d[]$
3. Trả về mảng $b[]$.

Câu 6: Viết thuật toán xây dựng đa thức nội suy Lagrange tương ứng với bộ điểm $S = \{(x_i, y_i), i = \overline{0, n}\}$ cho trước.
Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và mảng $y = [y_0, y_1, \ldots, y_m]$ chứa các giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy.
Output: Mảng $w = [w_0, w_1, \ldots, w_n]$ là hệ số của đa thức Lagrange

1. Khởi tạo ma trận M vuông cấp n toàn số 0
2. Cho biến chạy $i : 0 \to n - 1$, thực hiện:
   2.1 Cho biến chạy $j : 0 \to n - 1$, thực hiện:
   - Nếu $i = j$ thì gán $M[i][j] = 1$.
   - Nếu $i \neq j$ thì gán $M[i][j] = x_j - x_i$
3. Khởi tạo mảng $tich$ có n phần tử.
4. Cho biến chạy $i : 0 \to n - 1$, thực hiện:
   4.1 Cho biến chạy $j : 0 \to n - 1$, thực hiện:
   - Gán $tich[i] = 1$
   - $tich[i] = tich[i] \times M[j][i]$
5. Khởi tạo mảng hệ số d gồm n phần tử
6. Gán $D[i] = \dfrac{tich[i]}{i!}$

7. Tạo ma trận N vuông cấp $n+1$ chứa toàn số 0.

8. Gán $M_{0,n} = 1$

9. Cho biến chạy $i : 1 \to n$, thực hiện:

   (a) Cho biến chạy $j : 0 \to n$:

      9.1 Nếu $j = n$ thì gán $N_{i,j} = -N_{i-1,j} \times x_{i-1}$
      - Nếu không thì gán $N_{i,j} = N_{i-1,j+1} - N_{i-1,j} \times x_{i-1}$

10. Khởi tạo mảng hs_horner gồm $n+1$ phần tử

11. Cho biến chạy $i : 0 \to n$, gán $hs\_horner[i] = N[n][i]$

12. Tạo ma trận H vuông cấp n toàn số 0.

13. Cho biến chạy $i : 0 \to n$, gán $H[i][0] = 1$

14. cho biến chạy $i : 0 \to n-1$,

    14.1 Cho biến chạy $j : 1 \to n-1$, gán $H[i][j] = H[i][j-1] \times x_i + hs\_horner[j]$

15. Gán $w = D \times H$

16. Trả về mảng w.

TUẦN 3: NỘI SUY NEWTON

Câu 10: Tính hệ số đa thức nội suy Newton.

Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy.

Output: Mảng w chứa hệ số của đa thức Newton.

1. Khởi tạo ma trận M vuông cấp n chứa toàn số 0.
2. Cho biến chạy $i : 0 \to n-1$, gán $M_{i,0} = y_i$.
3. Cho biến chạy $j : 1 \to n-1$, thực hiện:

   3.1 Cho biến chạy $i : j \to n-1$, gán $M_{i,j} = \dfrac{M_{i,j-1} - M_{i-1,j-1}}{x_i - x_{i-j}}$.
4. Khởi tạo mảng $w[]$.
5. Cho biến chạy $i : 0 \to n-1$, gán $w_i = \dfrac{M_{i,i}}{i!}$.
6. Tạo ma trận N vuông cấp n chứa toàn số 0.
7. Gán $M_{0,n-1} = 1$
8. Tạo list x_{new} chứa m phần tử đầu của x.
9. Cho biến chạy $i : 1 \to n-1$, thực hiện:

   9.1 Cho biến chạy $j : 0 \to n-1$:
   - Nếu $j = n-1$ thì gán $N_{i,j} = -N_{i-1,j} \times x_{i-1}$
   - Nếu không thì gán $N_{i,j} = N_{i-1,j+1} - N_{i-1,j} \times x_{i-1}$
10. Gán ma trận $A = w \times N$.
11. Trả về ma trận A là hệ số của đa thức Newton.

Câu 10a: Tính hệ số đa thức nội suy Newton tiến.

Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy.

Output: Mảng w chứa hệ số của đa thức Newton.

1. Tạo mảng $x' = [x'_0, x'_1, \ldots, x'_m]$ là mảng đã sắp xếp các mốc nội suy ban đầu theo thứ tự tăng dần, và mảng $y' = [y'_0, y'_1, \ldots, y'_m]$ là sắp xếp các giá trị nội suy tương ứng theo mốc nội suy x'.
2. Khởi tạo ma trận M vuông cấp n chứa toàn số 0.

3. Cho biến chạy $i : 0 \to n-1$, gán $M_{i,0} = y'_i$.

4. Cho biến chạy $j : 1 \to n-1$, thực hiện:

   3.1 Cho biến chạy $i : j \to n-1$, gán $M_{i,j} = M_{i,j-1} - M_{i-1,j-1}$.

5. Khởi tạo mảng $w[]$.

6. Cho biến chạy $i : 0 \to n-1$, gán $w_i = \dfrac{M_{i,i}}{i!}$.

7. Tạo ma trận N vuông cấp n chứa toàn số 0.

8. Gán $M_{0,n-1} = 1$

9. Cho biến chạy $i : 1 \to n-1$, thực hiện:

   9.1 Cho biến chạy $j : 0 \to n-1 :$
   - Nếu $j = n-1$ thì gán $N_{i,j} = -N_{i-1,j} \times (i-1)$
   - Nếu không thì gán $N_{i,j} = N_{i-1,j+1} - N_{i-1,j} \times (i-1)$

10. Gán ma trận $A = w \times N$.

11. Trả về ma trận A là hệ số của đa thức Newton tiến.

Câu 10b: Tính hệ số đa thức nội suy Newton lùi.
Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy.
Output: Mảng w chứa hệ số của đa thức Newton.

1. Kiểm tra giá trị đầu vào nếu có $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại các giá trị (x_i, y_i).

2. Tạo mảng $x' = [x'_0, x'_1, \ldots, x'_m]$ là mảng đã sắp xếp các mốc nội suy ban đầu theo thứ tự tăng dần, và mảng $y' = [y'_0, y'_1, \ldots, y'_m]$ là sắp xếp các giá trị nội suy tương ứng theo mốc nội suy x'.

3. Khởi tạo ma trận M vuông cấp n chứa toàn số 0.

4. Cho biến chạy $i : 0 \to n-1$, gán $M_{i,0} = y'_i$.

5. Cho biến chạy $j : 1 \to n-1$, thực hiện:

   3.1 Cho biến chạy $i : j \to n-1$, gán $M_{i,j} = M_{i,j-1} - M_{i-1,j-1}$.

6. Khởi tạo mảng $w[]$.

7. Cho biến chạy $i : 0 \to n-1$, gán $w_i = \dfrac{M_{n-1,i}}{i!}$.

8. Tạo ma trận N vuông cấp n chứa toàn số 0.

9. Gán $M_{0,n-1} = 1$

10. Cho biến chạy $i : 1 \to n-1$, thực hiện:

    9.1 Cho biến chạy $j : 0 \to n-1$:

    - Nếu $j = n-1$ thì gán $N_{i,j} = N_{i-1,j} \times (i-1)$
    - Nếu không thì gán $N_{i,j} = N_{i-1,j+1} + N_{i-1,j} \times (i-1)$

11. Gán ma trận $A = w \times N$.

12. Trả về ma trận A là hệ số của đa thức Newton lùi.

Câu 11: Viết thuật toán trích xuất k điểm nội suy từ 1 danh sách các mốc nội suy cách đều cho trước phù hợp cho bài toán tính gần đúng giá trị hàm số tại điểm $\overline{x}$.

Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m+1$ là số lượng mốc nội suy, k là số lượng mốc nội suy cần trích xuất, $\overline{x}$ là điểm cần tính giá trị.

Output: Mảng x' chứa k điểm nội suy quanh $\overline{x}$ và mảng y' chứa các giá trị nội suy tương ứng.

1. Kiểm tra $k > n$?

    - Nếu có thì in ra thông báo "Không trích xuất được k mốc nội suy vì chỉ có n mốc, do đó ta sẽ lấy n mốc để tính."
    - Không thì chuyển tới bước sau.

2. Kiểm tra $k = n$?

    - Nếu có thì in ra thông báo "Ta lấy toàn bộ n mốc ban đầu để tính" rồi trả về mốc $x' = x$ và $y' = y$ rồi kết thúc thuật toán.
    - Không thì chuyển tới bước sau.

3. Tính bước nhảy $h = x_1 - x_0$.

4. Gán hiệu $min = |\overline{x} - x_0|,\ thu\_tu = 0$.

5. Cho biến chạy $i : 1 \to n-1$, thực hiện:

    (a) Gán $current\_diff = |\overline{x} - x_i|$

    (b) Nếu $current\_diff < min$ thì gán $min = current\_diff$ và $thu\_tu = i$.

6. Gán $m = \left\lfloor \dfrac{k-1}{2} \right\rfloor$ và $p = k - 1 - m$.

7. Gán $start = thu\_tu - m$ và $end = thu\_tu + p$.

8. Kiểm tra $start < 0$?

- Nếu không thì chuyển tới bước sau.
- Nếu có thì gán $end = end - start$ và $start = 0$ rồi chuyển tới bước sau.

9. Kiểm tra $end > n - 1$?
   - Nếu không thì chuyển tới bước sau.
   - Nếu có thì gán $start = start - end + n - 1$ và $end = n - 1$ rồi chuyển tới bước sau.
10. Cho biến chạy $i : 0 \to end - start$, thực hiện:
    (a) Gán $x'_i = x_{start+i}$
    (b) Gán $y'_i = y_{start+i}$
11. Trả về mảng x', y'.

TUẦN 4: NỘI SUY TRUNG TÂM

Câu 15b: Viết thuật toán thiết lập đa thức nội suy Stirlin

Input: Mảng $x = [x_0, x_1, \dots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \dots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy.

Output: Mảng w chứa hệ số của đa thức nội suy Stirlin.

1. Cho biến chạy $i : 0 \to m, j : i \to m$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.
2. Kiểm tra $n\%2 == 0$?
   (a) Nếu đúng thì in ra thông báo "Có chẵn mốc nội suy, không sử dụng được Stirlin" và kết thúc thuật toán.
   (b) Nếu không thì tiếp tục bước 3.
3. Khởi tạo ma trận M vuông cấp n toàn số 0.
4. Cho biến chạy $i : 0 \to n$, gán $M_{i,0} = y_i$.
5. Cho biến chạy $j : 1 \to n$, thực hiện:
   (a) Cho biến chạy $i : j \to n$, gán $M_{i,j} = M_{i,j-1} - M_{i-1,j-1}$
6. Khởi tạo mảng $tsp[]$ gồm có n phần tử.
7. Cho biến chạy $i : 0 \to n$, thực hiện:
   (a) Nếu $i\%2 = 0$? thì gán $k = \dfrac{n+i-1}{2}$ rồi gán $tsp_i = \dfrac{M_{k,i}}{i!}$
   (b) Nếu $i\%2 \neq 0$ thì gán $k = \dfrac{n+i-2}{2}$ rồi gán $tsp_i = \dfrac{1}{2}\dfrac{M_{k,i} + M_{k+1,i}}{i!}$
8. Đặt $t = \dfrac{n+1}{2}$.
9. Khởi tạo mảng $tsp\_chan[]$ có t phần tử và tsp_le có $t - 1$ phần tử.
10. Cho biến chạy $i : 0 \to t$, gán $tsp\_chan_i = tsp_{2\times i}$.
11. Cho biến chạy $i : 0 \to t - 1$, gán $tsp\_le_i = tsp_{2\times i+1}$.
12. Khởi tạo ma trận N có cấp t gồm toàn số 0.
13. Gán $N_{0,t-1} = 1$
14. Khởi tạo mảng x' gồm có $t - 1$ phần tử.

15. Cho biến chạy $i : 0 \to t-2$, gán $x'_i = i \times i$

16. Cho biến chạy $i : 1 \to t-1$, thực hiện:

    9.1 Cho biến chạy $j : 0 \to t-1$:
    - Nếu $j = t-1$ thì gán $N_{i,j} = -N_{i-1,j} \times x'_{i-1}$
    - Nếu không thì gán $N_{i,j} = N_{i-1,j+1} - N_{i-1,j} \times x'_{i-1}$

17. Khởi tạo ma trận N' có cấp $t-1$ gồm toàn số 0.

18. Cho biến chạy $i : 0 \to t-2$, thực hiện

    (a) Cho biến chạy $j : 0 \to t-2$, gán $N'_{i,j} = N_{i+1,j}$

19. Gán ma trận $hs\_chan = tsp\_chan \times N$

20. Gán ma trận $hs\_le = tsp\_le \times N'$

21. Khởi tạo ma trận $hs[]$ gồm n phần tử.

22. Cho biến chạy $i : 0 \to n-1$, thực hiện:

    (a) Nếu $i\%2 = 0$ thì gán $hs_i = hs\_chan_{i/2}$

    (b) Nếu $i\%2 \neq 0$ thì gán $hs_i = hs\_le_{(i-1)/2}$

23. Trả về mảng $hs[]$.

Câu 15c: Viết thuật toán thiết lập đa thức nội suy Bessel

Input: Mảng $x = [x_0, x_1, \dots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \dots, y_m]$ chứa giá trị nội suy, $n = m+1$ là số lượng mốc nội suy.

Output: Mảng w chứa hệ số của đa thức nội suy Bessel.

1. Cho biến chạy $i : 0 \to m, j : i \to m$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n-1$.

2. Kiểm tra $n\%2 == 0$?

    (a) Nếu không thì in ra thông báo "Có lẻ mốc nội suy, không sử dụng được Besel" và kết thúc thuật toán.

    (b) Nếu có thì tiếp tục bước 3.

3. Khởi tạo ma trận M vuông cấp n toàn số 0.

4. Cho biến chạy $i : 0 \to n$, gán $M_{i,0} = y_i$.

5. Cho biến chạy $j : 1 \to n$, thực hiện:

    (a) Cho biến chạy $i : j \to n$, gán $M_{i,j} = M_{i,j-1} - M_{i-1,j-1}$

6. Khởi tạo mảng $tsp[]$ gồm có n phần tử.
7. Cho biến chạy $i : 0 \to n$, thực hiện:
   (a) Nếu $i\%2 = 0$? thì gán $k = \dfrac{n+i}{2}$ rồi gán $tsp_i = \dfrac{M_{k,i} + M_{k-1,i}}{2 \times i!}$
   (b) Nếu $i\%2 \neq 0$ thì gán $k = \dfrac{n+i-1}{2}$ rồi gán $tsp_i = \dfrac{M_{k,i}}{i!}$
8. Đặt $t = \dfrac{n}{2}$.
9. Khởi tạo mảng $tsp\_chan[]$ có t phần tử và tsp_le có t phần tử.
10. Cho biến chạy $i : 0 \to t$, gán $tsp\_chan_i = tsp_{2\times i}$.
11. Cho biến chạy $i : 0 \to t-1$, gán $tsp\_le_i = tsp_{2\times i+1}$.
12. Khởi tạo ma trận N có cấp t gồm toàn số 0.
13. Gán $N_{0,t-1} = 1$
14. Khởi tạo mảng x' gồm có t phần tử.
15. Cho biến chạy $i : 0 \to t-1$, gán $x'_i = \dfrac{(2i+1)^2}{4}$
16. Cho biến chạy $i : 1 \to t-1$, thực hiện:
    9.1 Cho biến chạy $j : 0 \to t-1$:
    - Nếu $j = t-1$ thì gán $N_{i,j} = -N_{i-1,j} \times x'_{i-1}$
    - Nếu không thì gán $N_{i,j} = N_{i-1,j+1} - N_{i-1,j} \times x'_{i-1}$
17. Gán ma trận $hs\_chan = tsp\_chan \times N$
18. Gán ma trận $hs\_le = tsp\_le \times N$
19. Khởi tạo ma trận $hs[]$ gồm n phần tử.
20. Cho biến chạy $i : 0 \to n-1$, thực hiện:
    (a) Nếu $i\%2 = 0$ thì gán $hs_i = hs\_le_{i/2}$
    (b) Nếu $i\%2 \neq 0$ thì gán $hs_i = hs\_chan_{(i-1)/2}$
21. Trả về mảng $hs[]$.

TUẦN 6+7: NỘI SUY NGƯỢC + HÀM GHÉP TRƠN

Câu 18a: Viết thuật toán xác định tất cả các khoảng cách ly của phương trình $f(x) = \overline{y}$ từ bộ dữ liệu cho trước.

Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy, $\overline{y}$ là giá trị đã cho.

Output: Danh sách $khoang\_cach\_ly$ chứa các phần tử, mỗi phần tử có dạng (x_i, x_j) là các khoảng cách ly của phương trình.

1. Cho biến chạy $i : 0 \to m, j : i \to m$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.
2. Khởi tạo $khoang\_cach\_ly = []$.
3. Cho biến chạy $i : 0 \to n - 1$, thực hiện:
   (a) Nếu $y_i \leq \overline{y} < y_{i+1}$ thì thêm $khoang\_cach\_ly.append(x_i, x_{i+1})$
4. Đưa ra danh sách $khoang\_cach\_ly$.

Nếu phải xác định thêm cả các khoảng đơn điệu từ khoảng cách ly đã tìm.

1. Đưa ra danh sách $khoang\_cach\_ly$.
2. Xác định danh sách thu_tu chứa vị trí của các mốc cách ly đã tìm trong danh sách ban đầu.
3. Khởi tạo danh sách $don\_dieu = []$
4. Cho biến chạy $i : 0 \to m - 1$, thực hiện:
   (a) Gán $start = thu\_tu[i][0]$
   (b) Lặp với điều kiện $(y_{start} - y_{start-1}) \times (y_{thu\_tu[i][1]} - y_{thu\_tu[i][0]}) > 0$, thực hiện gán $start = start - 1$.
   (c) Gán $end = thu\_tu[i][1]$
   (d) Kiểm tra $end = n - 1$?
      - Nếu có thì in thông báo đã duyệt hết danh sách và đưa ra khoảng đơn điệu rồi kết thúc thuật toán.
      - Nếu không thì chuyển tới bước sau.
   (e) Lặp với điều kiện $(y_{end} - y_{end+1}) \times (y_{thu\_tu[i][1]} - y_{thu\_tu[i][0]}) < 0$, thực hiện gán $end = end + 1$.
   (f) Thêm $don\_dieu.append((x_{start}, x_{end}))$

5. Đưa ra các khoảng đơn điệu ứng với các khoảng cách ly.

Câu 18b: Viết thuật toán xác định tất cả các khoảng đơn điệu của hàm số từ bộ dữ liệu cho trước.

Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy.

Output: Danh sách *don_dieu* chứa các khoảng đơn điệu của bộ dữ liệu đã cho.

1. Kiểm tra $n < 2$?
   - Nếu có: Báo lỗi rồi **Trả về** mảng rỗng.
   - Ngược lại: Chuyển tới bước sau.
2. Khởi tạo mảng $D_{\text{don\_dieu}} = []$.
3. Gán $i_{\text{start}} = 0$
4. Gán $\text{current\_trend} = \text{sign}(y_1 - y_0)$
5. Cho biến chạy $i : 1 \to n - 2$, thực hiện:
   (a) Gán $\text{new\_trend} = \text{sign}(y_{i+1} - y_i)$
   (b) Kiểm tra $\text{current\_trend} \neq 0$, $\text{new\_trend} \neq 0$ và $\text{new\_trend} \neq \text{current\_trend}$)?
      - Nếu có thì chuyển tới bước 6.
      - Ngược lại chuyển tới bước.
   (c) Gán $D_{\text{don\_dieu}}.\text{append}((X_{i_{\text{start}}}, X_i))$
   (d) Gán $i_{start} = i$ và $\text{current\_trend} = \text{new\_trend}$.
   (e) Kiểm tra $\text{current\_trend} = 0$ và $\text{new\_trend} \neq 0$?
      - Nếu có thì gán $\text{current\_trend} = \text{new\_trend}$
6. Gán $i_{\text{end}} = n - 1$
7. Gán $D_{\text{don\_dieu}}.\text{append}((X_{i_{\text{start}}}, X_{i_{\text{end}}}))$
8. Trả về mảng $D_{\text{don\_dieu}}$

Câu 18c: Viết thuật toán nội suy cho phương pháp hàm ngược giải bài toán nội suy ngược.

Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy, giá trị cần tính $\overline{y}$.

Output: Danh sách các giá trị x' mà $f(x') = \overline{y}$.

1. Cho biến chạy $i : 0 \to m, j : i \to m$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.

2. Đưa ra các khoảng cách ly (x_k, x_{k+1}) thuộc danh sách $khoang\_cach\_ly[]$ từ bộ dữ liệu đã cho.

3. Với mỗi khoảng cách ly (x_k, x_{k+1}), tìm khoảng đơn điệu (x_p, x_q) tương ứng.

4. Trên mỗi khoảng đơn điệu (x_p, x_q), thực hiện:

    (a) Áp dụng gói NS_Newton để tìm ra nghiệm x'_i trong mỗi khoảng cách ly.

    (b) Thêm giá trị $x'.append(x'_i)$

5. Đưa ra danh sách $x' = [x'_0, x'_1, \ldots, x'_k]$ là các giá trị mà $f(x') = \overline{y}$.

Trong đó, gói NS_Newton được mô tả:
Input: Mảng $y = [y_0, y_1, \ldots, y_m]$ chứa các mốc nội suy và $x = [x_0, x_1, \ldots, x_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy, $\overline{y}$ là giá trị cần tìm mốc nội suy.
Output: Giá trị $\overline{x}$ thỏa mãn $f(\overline{y}) = \overline{x}$.

1. Khởi tạo ma trận M vuông cấp n chứa toàn số 0.

2. Cho biến chạy $i : 0 \to n-1$, gán $M_{i,0} = x_i$.

3. Cho biến chạy $j : 1 \to n-1$, thực hiện:

    3.1 Cho biến chạy $i : j \to n-1$, gán $M_{i,j} = \dfrac{M_{i,j-1} - M_{i-1,j-1}}{y_i - y_{i-j}}$.

4. Khởi tạo mảng $w[]$.

5. Cho biến chạy $i : 0 \to n-1$, gán $w_i = \dfrac{M_{i,i}}{i!}$.

6. Tạo ma trận N vuông cấp n chứa toàn số 0.

7. Gán $M_{0,n-1} = 1$

8. Tạo list x_{new} chứa m phần tử đầu của x.

9. Cho biến chạy $i : 1 \to n-1$, thực hiện:

    9.1 Cho biến chạy $j : 0 \to n-1$:

    - Nếu $j = n-1$ thì gán $N_{i,j} = -N_{i-1,j} \times y_{i-1}$
    - Nếu không thì gán $N_{i,j} = N_{i-1,j+1} - N_{i-1,j} \times y_{i-1}$

10. Gán $A = w \times N$.

11. Gán $P = 0$.

12. Cho biến chạy $i : 0 \to n-1$, tính $P = P + A_i \times \overline{y}^{n-1-i}$

13. Trả về P là giá trị cần tìm.

Câu 18d: Viết thuật toán nội suy cho phương pháp lặp giải bài toán nội suy ngược.
Input: Mảng $x = [x_0, x_1, \ldots, x_m]$ chứa các mốc nội suy và $y = [y_0, y_1, \ldots, y_m]$ chứa giá trị nội suy, $n = m + 1$ là số lượng mốc nội suy, giá trị cần tính $\frac{}{y}$.
Output: Danh sách các giá trị x' mà $f(x') = \overline{y}$.

1. Cho biến chạy $i : 0 \to m, j : i \to m$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.
2. Kiểm tra các mốc có cách đều nhau?
   - Có thì tới bước sau.
   - Không thì in ra thông báo "Không áp dụng được phương pháp này mà qua phương pháp hàm lặp" rồi kết thúc thuật toán.
3. Đưa ra các khoảng cách ly (x_k, x_{k+1}) thuộc danh sách $khoang\_cach\_ly[]$ từ bộ dữ liệu đã cho.
4. Với mỗi khoảng cách ly (x_k, x_{k+1}), tìm khoảng đơn điệu (x_p, x_q) tương ứng.
5. Cho biến chạy $i : 0 \to \text{len}(khoang\_cach\_ly - 1)$, thực hiện:
   (a) Nếu $|q - k + 1| \geq |k + 2 - p|$ thì thực hiện gói lap_tien.
   (b) Ngược lại, thực hiện gói lap_lui.
6.

Trong đó: gói lap_tien :
Input: Khoảng cách ly (x_k, x_{k+1}) và khoảng đơn điệu (x_p, x_q)
Output: Nghiệm x' thỏa mãn $f(x') = \overline{y}$.

1. Gán $don\_dieu\_moi = (x_k, x_q)$ và $y' = (y_k, y_q)$
2. Khởi tạo ma trận M vuông cấp $t = q - k + 1$ chứa toàn số 0.
3. Cho biến chạy $i : 0 \to t - 1$, gán $M_{i,0} = y'_i$.
4. Cho biến chạy $j : 1 \to n - 1$, thực hiện:
   3.1 Cho biến chạy $i : j \to n - 1$, gán $M_{i,j} = M_{i,j-1} - M_{i-1,j-1}$.
5. Khởi tạo mảng $w[]$.
6. Cho biến chạy $i : 0 \to n - 1$, gán $w_i = M_{i,i}$.
7. Gán $t_0 = \dfrac{\overline{y} - y_k}{w_1}$

8. Gán $t = t_0$ và $t\_prev = t_0 + 10000$
9. Thực hiện vòng lặp sau với điều kiện $|t - t\_prev| > 10^{-6}$:
   (a) Gán $t\_prev = t$ và $sum = 0$
   (b) Cho biến chạy $i : 2 \to n - 1$, thực hiện:
      i. Gán $prod = 1$
      ii. Cho biến chạy $j : 0 \to i - 1$, gán $prod = prod \times (t\_prev - j)$
      iii. Gán $sum = sum + \dfrac{w_i \times prod}{i!}$
   (c) Gán $t = t_0 - \dfrac{sum}{w_1}$ và quay lại vòng lặp.
10. Gán $h = x_{k+1} - x_k$
11. Đưa ra giá trị $x' = h \times t + x_k$

Trong đó: gói lap_lui : (Khi đi thi ko kịp thì viết: gói lap_lui có các bước giống với gói lap_tien, chỉ khác ở bước 2 gán $t = k + 2 - p$; bước 6 gán $w_i = M_{t-1,i}$ và bước $9(b)(ii)$ gán $prod = prod(t\_prev + j)$.

Input: Khoảng cách ly (x_k, x_{k+1}) và khoảng đơn điệu (x_p, x_q)
Output: Nghiệm x' thỏa mãn $f(x') = \overline{y}$.

1. Gán $don\_dieu\_moi = (x_p, x_{k+1})$ và $y' = (y_p, y_{k+1})$
2. Khởi tạo ma trận M vuông cấp $t = k + 2 - p$ chứa toàn số 0.
3. Cho biến chạy $i : 0 \to t - 1$, gán $M_{i,0} = y'_i$.
4. Cho biến chạy $j : 1 \to n - 1$, thực hiện:
   3.1 Cho biến chạy $i : j \to n - 1$, gán $M_{i,j} = M_{i,j-1} - M_{i-1,j-1}$.
5. Khởi tạo mảng $w[]$.
6. Cho biến chạy $i : 0 \to n - 1$, gán $w_i = M_{t-1,i}$.
7. Gán $t_0 = \dfrac{\overline{y} - y_k}{w_1}$
8. Gán $t = t_0$ và $t\_prev = t_0 + 10000$
9. Thực hiện vòng lặp sau với điều kiện $|t - t\_prev| > 10^{-6}$:
   (a) Gán $t\_prev = t$ và $sum = 0$
   (b) Cho biến chạy $i : 2 \to n - 1$, thực hiện:
      i. Gán $prod = 1$

ii. Cho biến chạy $j : 0 \to i-1$, gán $prod = prod \times (t\_prev + j)$

iii. Gán $sum = sum + \dfrac{w_i \times prod}{i!}$

(c) Gán $t = t_0 - \dfrac{sum}{w_1}$ và quay lại vòng lặp.

10. Gán $h = x_{k+1} - x_k$

11. Đưa ra giá trị $x' = h \times t + x_k$

Câu 22: Viết thuật toán xác định hàm thực nghiệm bằng phương pháp bình phương tối thiểu biết tập cơ sở $\{\varphi_i(x)\}_{\overline{1,m}}$ và tập dữ liệu $\{(x_i, y_i)\}_{\overline{1,n}}$, và xác định sai số trung bình phương tìm được.

Input: Mảng $x = [x_1, x_2, \ldots, x_n]$ chứa các mốc nội suy và $y = [y_1, y_2, \ldots, y_n]$ chứa giá trị nội suy, Tập cơ sở $\{\varphi_i(x)\}_{\overline{1,m}}$

Output: Vecto a và sai số ϵ.

1. Cho biến chạy $i : 0 \to n, j : i \to n$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.

2. Kiểm tra các hàm $\{\varphi_i(x)\}_{\overline{1,m}}$ có độc lập tuyến tính?

   - Nếu có thì tiếp tục bước sau.
   - Nếu không thì loại bỏ các hàm phụ thuộc tuyến tính và đưa ra bộ $\{\varphi_i(x)\}_{\overline{1,k}}$ mới là các hàm độc lập tuyến tính và tiếp tục bước sau.

3. Tạo ma trận Φ có kích cỡ $n \times m$ toàn là số 0.

4. Cho biến chạy $i : 1 \to n$, thực hiện:

   (a) Cho biến chạy $j : 1 \to m$, gán $\Phi_{i,j} = \varphi_j(x_i)$

5. Tính ma trận $M = \Phi^T \Phi$

6. Tính vecto $b = \Phi^T \times y$

7. Giải hệ phương trình $Ma = b$ để tìm ra a.

8. Tính $\epsilon = \sqrt{\dfrac{1}{n}(\sum\limits_{i=1}^{n} (f(x_i) - g(x_i))^2}$ là sai số trung bình phương.

Câu 22b: Viết thuật toán xác định hàm thực nghiệm theo các dạng dưới đây bằng phương pháp bình phương tối thiểu biết tập cơ sở sở $\{\varphi_i(x)\}_{\overline{1,m}}$ và tập dữ liệu $\{(x_i, y_i)\}_{\overline{1,n}}$ với $y = ae^{b_1\varphi_1(x) + b_2\varphi_2(x)}$ trong đó các hàm cơ sở ko phải hàm đồng nhất.

Input: Mảng $x = [x_1, x_2, \ldots, x_n]$ chứa các mốc nội suy và $y = [y_1, y_2, \ldots, y_n]$ chứa giá trị nội suy, Tập cơ sở $\{\varphi_i(x)\}_{\overline{1,m}}$

Output: Bộ số a, b_1, b_2.

1. Cho biến chạy $i : 0 \to n, j : i \to n$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.
2. Gán $dem = 0, x'[]$
3. Cho biến chạy $i : 0 \to n$, kiểm tra $y_i < 0$?
   (a) Nếu có thì gán $dem = dem + 1$ và thêm $x'.append((x_i, y_i))$.
   (b) Nếu không thì chuyển tới bước sau.
4. Kiểm tra $dem > \dfrac{n}{2}$?
   (a) Nếu không thì tạo $x[]$ mới bằng cách bỏ các phần tử thuộc $x'[]$ ra khỏi $x[]$, và chuyển tới bước lớn
   (b) Gán $y' = \ln(-y)$ và chuyển tới bước
5. Kiểm tra các hàm $\{\varphi_i(x)\}_{\overline{1,m}}$ có độc lập tuyến tính?
   - Nếu có thì tiếp tục bước sau.
   - Nếu không thì loại bỏ các hàm phụ thuộc tuyến tính và đưa ra bộ $\{\varphi_i(x)\}_{\overline{1,k}}$ mới là các hàm độc lập tuyến tính và tiếp tục bước sau.
6. Gán $y' = \ln(y)$
7. Tạo ma trận Φ có kích cỡ $n \times m$ toàn là số 0.
8. Cho biến chạy $i : 1 \to n$, thực hiện:
   (a) Gán $\Phi_{i,1} = 1$
   (b) Gán $\Phi_{i,2} = \varphi_1(x_i)$
   (c) Gán $\Phi_{i,3} = \varphi_2(x_i)$
9. Tính ma trận $M = \Phi^T\Phi$
10. Tính vecto $b' = \Phi^T \times y'$
11. Giải hệ phương trình $Ma' = b'$ để tìm ra nghiệm $a' = [a'_0, a'_1, a'_2]$.
12. Gán $a = e^{a'_0}(hoc\ -e^{a'_0}),\ b_1 = a'_1, b_2 = a'_2$.
13. Trả về giá trị a, b_1, b_2.

Câu 22c: Viết thuật toán xác định hàm thực nghiệm theo các dạng dưới đây bằng phương pháp bình phương tối thiểu biết tập cơ sở sở $\{\varphi_i(x)\}_{\overline{1,m}}$ và tập dữ liệu $\{(x_i, y_i)\}_{\overline{1,n}}$ với $y = ax^b$.

Input: Mảng $x = [x_1, x_2, \ldots, x_n]$ chứa các mốc nội suy và $y = [y_1, y_2, \ldots, y_n]$ chứa giá trị nội suy, Tập cơ sở $\{\varphi_i(x)\}_{\overline{1,m}}$

Output: Bộ số a, b.

1. Cho biến chạy $i : 0 \to n, j : i \to n$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.
2. Cho biến chạy $i : 0 \to n$, kiểm tra $x_1 > 0 \&\& y_i > 0$?
   (a) Nếu có thì bỏ (x_i, y_i) khỏi bộ dữ liệu.
   (b) Nếu không thì chuyển tới bước sau.
3. Kiểm tra các hàm $\{\varphi_i(x)\}_{\overline{1,m}}$ có độc lập tuyến tính?
   - Nếu có thì tiếp tục bước sau.
   - Nếu không thì loại bỏ các hàm phụ thuộc tuyến tính và đưa ra bộ $\{\varphi_i(x)\}_{\overline{1,k}}$ mới là các hàm độc lập tuyến tính và tiếp tục bước sau.
4. Gán $y' = \ln(y); x' = \ln(x)$
5. Tạo ma trận Φ có kích cỡ $n \times m$ toàn là số 0.
6. Cho biến chạy $i : 1 \to n$, thực hiện:
   (a) Gán $\Phi_{i,1} = 1$
   (b) Gán $\Phi_{i,2} = \ln(x_i)$
7. Tính ma trận $M = \Phi^T \Phi$
8. Tính vecto $b' = \Phi^T \times y'$
9. Giải hệ phương trình $Ma' = b'$ để tìm ra nghiệm $a' = [a'_0, a'_1]$.
10. Gán $a = e^{a'_0},\ b_1 = a'_1, b_2 = a'_2$.
11. Trả về giá trị a, b.

TUẦN 8-10: ĐẠO HÀM - TÍCH PHÂN

Câu 25 +26: Viết thuật toán tính đạo hàm bằng công thức $p+1$ điểm tại các mốc nội suy trong một bảng dữ liệu cho trước.

Input:

- p: Số lượng mốc nội suy sẽ được sử dụng.
- $start$: Chỉ số ban đầu của mốc đầu tiên trong tập dữ liệu lớn hơn (B).
- $\overline{x}$: Giá trị cần tính đạo hàm.
- B: Ma trận/Mảng chứa dữ liệu gốc, với $B[0]$ là các mốc x_i và $B[1]$ là các giá trị $y_i = f(x_i)$.

Output:

- gt: Giá trị đạo hàm cấp một của hàm số tại điểm $\overline{x}$.

1. Cho biến chạy $i : 0 \to m, j : i \to m$, kiểm tra $x_i = x_j$ nhưng $y_i \neq y_j$ thì loại bỏ (x_i, y_i) khỏi bảng ban đầu và gán $n = n - 1$.
2. Khởi tạo mảng moc_noi_suy và gia_tri đều có p phần tử.
3. Cho biến chạy $i : 0 \to p - 1$, thực hiện:
    - Gán $moc\_noi\_suy_i = B_{0,start+i}$
    - Gán $gia\_tri_i = B_{1,start+i}$
4. Cho biến chạy $i : 1 \to p - 2$, kiểm tra nếu $moc\_noi\_suy_{i+1} - moc\_noi\_suy_i \neq moc\_noi\_suy_i - moc\_noi\_suy_{i-1}$ thì in thông báo lỗi mốc nội suy không cách đều và xử lí dữ liệu.
5. Gán $buoc\_nhay = moc\_noi\_suy_1 - moc\_noi\_suy_0$.
6. Tính $chi\_so = \frac{\overline{x} - moc\_noi\_suy_0}{buoc\_nhay}$
7. Kiểm tra $chi\_so > p$? Có thì in thông báo lỗi "Không thể thực hiện tính đạo hàm bằng p mốc" và **Dừng thuật toán**. Ngược lại tiếp tục thuật toán.
8. Khởi tạo mảng he_so có $p - 1$ phần tử.
9. Gán $f(t, p+2) = \prod_{i=0}^{p} (t - p)$
10. Cho biến chạy $i : 0 \to p - 2$, thực hiện:

- Gán $he\_so_i = \frac{d}{dt}\left[f(t, i+2)\right]$ tại $t = thu\_tu$.

11. Khởi tạo mảng hs có p phần tử.
12. Cho biến chạy $i : 0 \to p-1$, thực hiện:
    - Tính $S = \sum_{k=0}^{p-2}\left[\frac{(-1)^{k+1}\times C_{k+1}^{i}\times he\_so_k}{(k+1)!}\right]$
    - Nếu i là **số chẵn** ($i \pmod 2 = 0$): Gán $hs_i = \dfrac{S}{buoc\_nhay}$.
    - Nếu i là **số lẻ** ($i \pmod 2 \neq 0$): Gán $hs_i = \dfrac{-S}{buoc\_nhay}$.
13. Khởi tạo biến $gt = 0$.
14. Cho biến chạy $i : 0 \to p-1$, thực hiện:
$$gt = gt + gia\_tri_i \times MT\_hs[1][i]$$
15. Trả về giá trị gt.

Câu 29: Thuật toán tính gần đúng giá trị của các tích phân:

a) Cho $f(x), a, b, \epsilon$, tính $\int\limits_a^b f(x)dx$ bằng các công thức đã học.

a) **Phương pháp hình thang tổng hợp:**

Input: $f(x), a, b, \epsilon$

Output: $\int\limits_a^b f(x)dx$

1. Gán $n = 2$
2. Tính $h = \dfrac{b-a}{n}$
3. Gán $S_1 = f(a+h)$
4. Gán $S_{sau} = \dfrac{h}{2} \times (f(a) + f(b) + 2 \times S_1)$
5. Gán $\text{SaiSo} = 1$
6. Kiểm tra $\text{SaiSo} > \epsilon$? thì thực hiện:
    (a) Gán $S_{trc} = S_{sau}$
    (b) Gán $n = 2 \times n$.

   (c) Gán $h = \dfrac{b-a}{n}$.
   (d) Cho biến chạy i: $0 \to \dfrac{n-1}{2}$, gán: $S_1 = S_1 + f(a + (2i+1) \times h)$
   (e) Gán $S_{sau} = \dfrac{h}{2} \times (f(a) + f(b) + 2 \times S_1)$
   (f) Gán $\text{SaiSo} = \frac{|S_{\text{sau}} - S_{\text{trước}}|}{3}$ và quay lại vòng lặp

7. Trả về $I = S_{sau}$

Phương pháp simson
Input:

- $f(x)$: Hàm số cần tính tích phân.
- a, b: Hai cận của tích phân.
- ε: Sai số tối đa cho phép.

Output:

- I: Giá trị tích phân gần đúng thỏa mãn sai số ε.

1. Gán $n = 4$
2. Tính $h = \dfrac{b-a}{n}$
3. Gán $S_1 = f(a+h) + f(a+3h)$
4. Gán $S_2 = f(a+2h)$
5. Gán $S_{sau} = \dfrac{h}{3} \times (f(a) + f(b) + 4 \times S_1 + 2 \times S_2)$
6. Gán $\text{SaiSo} = 1$
7. Kiểm tra $\text{SaiSo} > \epsilon$? thì thực hiện:
   (a) Gán $S_{trc} = S_{sau}$
   (b) Gán $n = 2 \times n$.
   (c) Gán $h = \dfrac{b-a}{n}$.
   (d) Gán $S_2 = S_1 + S_2$
   (e) Cho biến chạy i: $0 \to \dfrac{n-1}{2}$, gán: $S_1 = f(a + (2i+1) \times h)$
   (f) Gán $S_{sau} = \dfrac{h}{3} \times (f(a) + f(b) + 4 \times S_1 + 2 \times S_2)$
   (g) Gán $\text{SaiSo} = \frac{|S_{\text{sau}} - S_{\text{trước}}|}{15}$ và quay lại vòng lặp
8. Trả về $I = S_{sau}$

TUẦN 11: PHƯƠNG TRÌNH VI PHÂN

0.1 Euler hiện 1 phương trình

Câu 32c.1: Viết thuật toán tìm nghiệm số của bài toán 1 chiều giải phương trình vi phân bằng EUler hiện.

Input: Hàm $f(t,x) = x'$ và giá trị $x_0 = x(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.
Output: Bộ nghiệm số (t,x) với $t \in [t_0, T]$ và $x = x(t)$.

1. Tính số bước lặp $n = \dfrac{T - t_0}{h}$.
2. Khởi tạo mảng kết quả $x[]$, khoảng giá trị $t[]$, đạo hàm $f[]$.
3. Gán $x_0 = x_0$ và $t_0 = t_0, f_0 = f(t_0, x_0)$ *(# với $x[0]$ là x ở vị trí 0, còn x_0 là giá trị số x_0.)*
4. Cho biến chạy $i : 0 \to n-1$, thực hiện:
   (a) Tính $x_{i+1} = x_i + h \times f_i$
   (b) Gán $t_{i+1} = t_i + h$
   (c) Gán $f_{i+1} = f(t_{i+1}, x_{x+1})$
5. Trả về bộ biến $t[]$ và bộ giá trị $x[]$.

0.2 Euler hiện 2 phương trình

Câu 32c.2: Viết thuật toán tìm nghiệm số của bài toán 2 chiều giải phương trình vi phân bằng EUler hiện.

Input: Hàm $\begin{cases} f(t,x,y) = x' \\ g(t,x,y) = y' \end{cases}$ và giá trị $x_0 = x(t_0); y_0 = y(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.
Output: Bộ nghiệm số (t,x,y) với $t \in [t_0, T]$ và $x = x(t), y = y(t)$.

1. Tính $n = \dfrac{T - t_0}{h}$.
2. Khởi tạo mảng $x[]$, $y[]$, $t[]$, $f[], g[]$.
3. Gán $x_0 = x_0$; $y_0 = y_0, f_0 = f(t_0, x_0, y_0), g_0 = g(t_0, x_0, y_0)$ và $t_0 = t_0$ *(# với x_0 thứ nhất là x ở vị trí 0, còn x_0 là giá trị số x_0.)*

4. Cho biến chạy $i : 0 \to n-1$, thực hiện:
   (a) Tính $x_{i+1} = x_i + h \times f_i$
   (b) Tính $y_{i+1} = y_i + h \times g_i$
   (c) Gán $t_{i+1} = t_i + h$
   (d) Gán $f_{i+1} = f(t_{i+1}, x_{x+1}, y_{x+1})$
   (e) Gán $g_{i+1} = g(t_{i+1}, x_{x+1}, y_{x+1})$
5. Trả về bộ biến $t[]$ và bộ giá trị $x[], y[]$.

0.3 Euler ẩn lặp đơn

Câu 32c.3: Viết thuật toán tìm nghiệm số của bài toán 1 chiều giải phương trình vi phân bằng EUler ẩn dùng lặp đơn.

Input: Hàm $\begin{cases} f(t,x,y) = x' \\ g(t,x,y) = y' \end{cases}$ và giá trị $x_0 = x(t_0); y_0 = y(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.

Output: Bộ nghiệm số (t, x) với $t \in [t_0, T]$ và $x = x(t)$.

1. Tính $n = \dfrac{T - t_0}{h}$.
2. Khởi tạo mảng $x[]$, $t[]$, $f[]$.
3. Gán $x_0 = x_0; f_0 = f(t_0, x_0)$ và $t_0 = t_0$ *(# với x_0 thứ nhất là x ở vị trí 0, còn x_0 là giá trị số x_0.)*
4. Cho biến chạy $i : 0 \to n-1$, thực hiện:
   (a) Gán $t_{i+1} = t_i + h$
   (b) Tính $x_{predict} = x_i + h \times f_i$
   (c) Tính $x_{lap1} = x_i + h \times f(t_{i+1}, x_{predict})$
   (d) Tính $x_{lap2} = x_i + h \times f(t_{i+1}, x_{lap1})$
   (e) Gán $x_{i+1} = x_{lap2}$
   (f) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$
5. Trả về bộ biến $t[]$ và bộ giá trị $x[], y[]$.

0.4 Euler ẩn lặp Newton

Câu 32c.4: Viết thuật toán tìm nghiệm số của bài toán 1 chiều giải phương trình vi phân bằng EUler ẩn dùng lặp Newton.

Input: Hàm $f(t, x) = x'$ và giá trị $x_0 = x(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.
Output: Bộ nghiệm số (t, x) với $t \in [t_0, T]$ và $x = x(t)$.

1. Tính $n = \dfrac{T - t_0}{h}$.
2. Khởi tạo mảng $x[]$, $t[]$.
3. Gán $x_0 = x_0$; và $t_0 = t_0$ *(# với x_0 thứ nhất là x ở vị trí 0, còn x_0 là giá trị số x_0.)*
4. Gán $\epsilon = 10^{-8}$
5. Cho biến chạy $i : 0 \to n - 1$, thực hiện:
   (a) Gán $t_{i+1} = t_i + h$
   (b) Tính $x_{predict} = x_i + h \times f(t_i, x_i)$
   (c) Gán $x_{next} = x_{predict}$
   (d) Gán $saiso = 1$
   (e) Lặp các bước sau cho tới khi $saiso < 10^{-4}$:
      i. Gán $x_{prev} = x_{next}$
      ii. Tính $Gx = x_{prev} - x_i - h \times f(t_{i+1}, x_{prev})$
      iii. Tính $dfdx = \dfrac{f(t_{i+1}, x_{prev} + \epsilon) - f(t_{i+1}, x_{prev} - \epsilon)}{2 \times \epsilon}$
      iv. Gán $x_{next} = x_{prev} - \dfrac{Gx}{1 - h \times dfdx}$
      v. Gán $saiso = |x_{next} - x_{prev}|$
   (f) Gán $x_{i+1} = x_{next}$
6. Trả về bộ biến $t[]$ và bộ giá trị $x[]$.

0.5 Hình thang lặp Newton

Câu 32c.5: Viết thuật toán tìm nghiệm số của bài toán 1 chiều giải phương trình vi phân bằng hình thang.

Input: Hàm $f(t, x) = x'$ và giá trị $x_0 = x(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.
Output: Bộ nghiệm số (t, x) với $t \in [t_0, T]$ và $x = x(t)$.

1. Tính $n = \dfrac{T - t_0}{h}$.

2. Khởi tạo mảng $x[]$, $t[]$.
3. Gán $x_0 = x_0$; và $t_0 = t_0$ *(# với x_0 thứ nhất là x ở vị trí 0, còn x_0 là giá trị số x_0.)*
4. Gán $\epsilon = 10^{-8}$
5. Cho biến chạy $i : 0 \rightarrow n-1$, thực hiện:
   (a) Gán $t_{i+1} = t_i + h$
   (b) Tính $x_{predict} = x_i + h \times f(t_i, x_i)$
   (c) Gán $x_{next} = x_{predict}$
   (d) Gán $saiso = 1$
   (e) Lặp các bước sau cho tới khi $saiso < 10^{-4}$:
      i. Gán $x_{prev} = x_{next}$
      ii. Tính $Gx = x_{prev} - x_i - \frac{h}{2} \times (f(t_i, x_i) + f(t_{i+1}, x_{prev}))$
      iii. Tính $dfdx = \dfrac{f(t_{i+1}, x_{prev} + \epsilon) - f(t_{i+1}, x_{prev} - \epsilon)}{2 \times \epsilon}$
      iv. Gán $x_{next} = x_{prev} - \dfrac{Gx}{1 - \frac{h}{2} \times dfdx}$
      v. Gán $saiso = |x_{next} - x_{prev}|$
   (f) Gán $x_{i+1} = x_{next}$
6. Trả về bộ biến $t[]$ và bộ giá trị $x[]$.

0.6 RK 1 phương trình

Câu 37a: Viết thuật toán tìm nghiệm số của bài toán 1 chiều giải phương trình vi phân bằng RK2,3,4 thông dụng đã xây dựng.

Định nghĩa: $f_i = f(t_i, x_i)$

Input: Hàm $f(t,x) = x'$ và giá trị $x_0 = x(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.

Output: Bộ nghiệm số (t,x) với $t \in [t_0, T]$ và $x = x(t)$.

1. Tính số bước lặp $n = \dfrac{T - t_0}{h}$.
2. Khởi tạo mảng kết quả $x[]$, khoảng giá trị $t[]$, đạo hàm $f[]$.
3. Gán $x_0 = x_0$ và $t_0 = t_0, f_0 = f(t_0, x_0)$ *(# với $x[0]$ là x ở vị trí 0, còn x_0 là giá trị số x_0.)*
4. Cho biến chạy $i : 0 \to n-2$, thực hiện các bước sau:
   - **Nếu dùng RK2:**
     (a) Gán $k_1 = h \times f(t_i, x_i)$
     (b) Gán $k_2 = h \times f(t_i + h, x_i + k_1)$
     (c) Gán $t_{i+1} = t_i + h$
     (d) Gán $x_{i+1} = x_i + \dfrac{1}{2}(k_1 + k_2)$
     (e) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$
   - **Nếu dùng RK3:**
     (a) Gán $k_1 = h \times f(t_i, x_i)$
     (b) Gán $k_2 = h \times f(t_i + \dfrac{h}{2}, x_i + \dfrac{k_1}{2})$
     (c) Gán $k_3 = h \times f(t_i + h, x_i + (2k_2 - k_1))$
     (d) Gán $t_{i+1} = t_i + h$
     (e) Gán $x_{i+1} = x_i + \dfrac{1}{6}(k_1 + 4 \times k_2 + k_3)$
     (f) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$
   - **Nếu dùng RK4:**
     (a) Gán $k_1 = h \times f(t_i, x_i)$
     (b) Gán $k_2 = h \times f(t_i + \dfrac{h}{2}, x_i + \dfrac{k_1}{2})$
     (c) Gán $k_3 = h \times f(t_i + \dfrac{h}{2}, x_i + \dfrac{k_2}{2})$
     (d) Gán $k_4 = h \times f(t_i + h, x_i + k_3)$
     (e) Gán $t_{i+1} = t_i + h$

(f) Gán $x_{i+1} = x_i + \frac{1}{6}(k_1 + 2 \times k_2 + 2 \times k_3 + k_4)$

(g) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$

5. Trả về bộ biến $t[]$ và bộ giá trị $x[]$.

0.7 RK 2 phương trình

Câu 37b: Viết thuật toán tìm nghiệm số của bài toán 2 chiều giải phương trình vi phân bằng RK2,3,4 thông dụng đã xây dựng.

Input: Hàm $\begin{cases} f(t,x,y) = x' \\ g(t,x,y) = y' \end{cases}$ và giá trị $x_0 = x(t_0); y_0 = y(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.

Output: Bộ nghiệm số (t, x, y) với $t \in [t_0, T]$ và $x = x(t), y = y(t)$.

1. Tính $n = \frac{T - t_0}{h}$.
2. Khởi tạo mảng $x[]$, $y[]$, $t[]$, $f[], g[]$.
3. Gán $x_0 = x_0$; $y_0 = y_0, f_0 = f(t_0, x_0, y_0), g_0 = g(t_0, x_0, y_0)$ và $t_0 = t_0$ *(# với x_0 thứ nhất là x ở vị trí 0, còn x_0 là giá trị số x_0.)*
4. Cho biến chạy $i : 0 \to n - 2$, thực hiện các bước sau:

- **Nếu dùng RK2:**

(a) Gán $k_1 = h \times f(t_i, x_i, y_i)$

(b) Gán $l_1 = h \times g(t_i, x_i, y_i)$

(c) Gán $k_2 = h \times f(t_i + h, x_i + k_1, y_i + l_1)$

(d) Gán $l_2 = h \times f(t_i + h, x_i + k_1, y_i + l_1)$

(e) Gán $t_{i+1} = t_i + h$

(f) Gán $x_{i+1} = x_i + \frac{1}{2}(k_1 + k_2)$, $\quad y_{i+1} = y_i + \frac{1}{2}(l_1 + l_2)$

(g) Gán $f_{i+1} = f(t_{i+1}, x_{i+1}, y_{i+1})$, $g_{i+1} = g(t_{i+1}, x_{i+1}, y_{i+1})$

- **Nếu dùng RK3:**

(a) Gán $k_1 = h \times f(t_i, x_i, y_i)$

(b) Gán $l_1 = h \times f(t_i, x_i, y_i)$

(c) Gán $k_2 = h \times f(t_i + \frac{h}{2}, x_i + \frac{k_1}{2}, y_i + \frac{l_1}{2})$

(d) Gán $l_2 = h \times g(t_i + \frac{h}{2}, x_i + \frac{k_1}{2}, y_i + \frac{l_1}{2})$

(e) Gán $k_3 = h \times f(t_i + h, x_i + (2k_2 - k_1), y_i + (2l_2 - l_1))$

(f) Gán $l_3 = h \times g(t_i + h, x_i + (2k_2 - k_1), y_i + (2l_2 - l_1))$
(g) Gán $t_{i+1} = t_i + h$
(h) Gán $x_{i+1} = x_i + \frac{1}{6}(k_1 + 4 \times k_2 + k_3)$
(i) Gán $y_{i+1} = y_i + \frac{1}{6}(l_1 + 4 \times l_2 + l_3)$
(j) Gán $f_{i+1} = f(t_{i+1}, x_{i+1}, y_{i+1}),\ g_{i+1} = g(t_{i+1}, x_{i+1}, y_{i+1})$

- **Nếu dùng RK4:**
  (a) Gán $k_1 = h \times f(t_i, x_i, y_i)$
  (b) Gán $l_1 = h \times g(t_i, x_i, y_i)$
  (c) Gán $k_2 = h \times f(t_i + \frac{h}{2}, x_i + \frac{k_1}{2}, y_i + \frac{l_1}{2})$
  (d) Gán $l_2 = h \times g(t_i + \frac{h}{2}, x_i + \frac{k_1}{2}, y_i + \frac{l_1}{2})$
  (e) Gán $k_3 = h \times f(t_i + \frac{h}{2}, x_i + \frac{k_2}{2}, y_i + \frac{l_2}{2})$
  (f) Gán $l_3 = h \times g(t_i + \frac{h}{2}, x_i + \frac{k_2}{2}, y_i + \frac{l_2}{2})$
  (g) Gán $k_4 = h \times f(t_i + h, x_i + k_3, y_i + l_3)$
  (h) Gán $l_4 = h \times g(t_i + h, x_i + k_3, y_i + l_3)$
  (i) Gán $t_{i+1} = t_i + h$
  (j) Gán $x_{i+1} = x_i + \frac{1}{6}(k_1 + 2 \times k_2 + 2 \times k_3 + k_4)$
  (k) Gán $y_{i+1} = y_i + \frac{1}{6}(l_1 + 2 \times l_2 + 2 \times l_3 + l_4)$
  (l) Gán $f_{i+1} = f(t_{i+1}, x_{i+1}, y_{i+1}),\ g_{i+1} = g(t_{i+1}, x_{i+1}, y_{i+1})$

5. Trả về bộ biến $t[]$ và bộ giá trị $x[], y[]$.

0.8 ABs-AMs 1 phương trình

Câu 38a: Viết thuật toán tìm nghiệm số của bài toán 1 chiều giải phương trình vi phân bằng công thức $ABs - AMs$.
Định nghĩa: $f_i = f(t_i, x_i)$
Input: Hàm $f(t, x) = x'$ và giá trị $x_0 = x(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.
Output: Bộ nghiệm số (t, x) với $t \in [t_0, T]$ và $x = x(t)$.

1. Tính số bước lặp $n = \frac{T - t_0}{h}$.
2. Khởi tạo mảng $x[]$, $t[]$, $f[]$.
3. Gán $x_0 = x_0, f_0 = f(t_0, x_0)$ và $t_0 = t_0$

4. Cho $k : 0 \to s-1$, thực hiện tìm bộ $s-1$ nghiệm ban đầu x_k bằng hàm $nghiem\_RKs$.

5. Tạo mảng $\overline{x} = x[].copy$

6. Cho biến chạy $i : s-1 \to n-1$, thực hiện:

   - **Nếu $s = 2$:**
     (a) Gán $t_{i+1} = t_i + h$
     (b) Sử dụng ABs tìm nghiệm dự đoán $\overline{x_{i+1}} = x_i + \dfrac{h}{2} \times (3 \times f_i - f_{i-1})$
     (c) Sử dụng AMs tìm nghiệm hiệu chỉnh $x_{i+1} = x_i + \dfrac{h}{2} \times f(t_{i+1}, \overline{x_{i+1}}) + \dfrac{h}{2} \times f_i$
     (d) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$
   - **Nếu dùng $s = 3$:**
     (a) Gán $t_{i+1} = t_i + h$
     (b) Sử dụng ABs tìm nghiệm dự đoán $\overline{x_{i+1}} = x_i + \dfrac{h}{12} \times (23 \times f_i - 16 \times f_{i-1} + 5 \times f_{i-2})$
     (c) Sử dụng AMs tìm nghiệm hiệu chỉnh $x_{i+1} = x_i + \dfrac{h}{12} \times (5 \times f(t_{i+1}, \overline{x_{i+1}}) + 8 \times f_n - f_{n-1})$
     (d) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$
   - **Nếu dùng $s = 4$:**
     (a) Gán $t_{i+1} = t_i + h$
     (b) Sử dụng ABs tìm nghiệm dự đoán $\overline{x_{i+1}} = x_i + \dfrac{h}{24} \times (55 \times f_i - 59 \times f_{i-1} + 37 \times f_{i-2} - 9 \times f_{i-3})$
     (c) Sử dụng AMs tìm nghiệm hiệu chỉnh $x_{i+1} = x_i + \dfrac{h}{24} \times (9 \times f_{i+1} + 19 \times f_i - 5 \times f_{i-1} + f_{i-2})$
     (d) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$
   - **Nếu dùng $s = 5$:**
     (a) Gán $t_{i+1} = t_i + h$
     (b) Sử dụng ABs tìm nghiệm dự đoán $\overline{x_{i+1}} = x_i + \dfrac{h}{720} \times (1901 \times f_i - 2774 \times f_{i-1} + 2616 \times f_{i-2} - 1274 \times f_{i-3} + 251 \times f_{i-4})$
     (c) Sử dụng AMs tìm nghiệm hiệu chỉnh $x_{i+1} = x_i + \dfrac{h}{720} \times (251 \times f(t_{i+1}, \overline{x_{i+1}}) + 646 \times f_i - 264 \times f_{i-1} + 106 \times f_{i-2} - 19 \times f_{i-3})$
     (d) Gán $f_{i+1} = f(t_{i+1}, x_{i+1})$

7. Trả về bộ biến $t[]$ và bộ giá trị $x[]$.

0.9 ABs-AMs 2 phương trình

Câu 38b: Viết thuật toán tìm nghiệm số của bài toán 2 chiều giải phương trình vi phân bằng công thức $ABs - AMs$.

Input: Hàm $\begin{cases} f(t,x,y) = x' \\ g(t,x,y) = y' \end{cases}$ và giá trị $x_0 = x(t_0); y_0 = y(t_0)$, khoảng giá trị $[t_0, T]$, bước nhảy h.

Định nghĩa: $f_i = f(t_i, x_i, y_i); g_i = g(t_i, x_i, y_i)$

Output: Bộ nghiệm số (t, x, y) với $t \in [t_0, T]$ và $x = x(t), y = y(t)$.

1. Tính số bước lặp $n = \dfrac{T - t_0}{h}$.
2. Khởi tạo mảng $x[],\ y[], f[], g[],\ t[]$.
3. Gán $x_0 = x_0,\ y_0 = y_0, f_0 = f(t_0, x_0, y_0), g_0 = g(t_0, x_0, y_0)$ và $t_0 = t_0$
4. Cho $k : 0 \to s - 1$, thực hiện tìm bộ $s - 1$ nghiệm ban đầu (x_k, y_k) bằng hàm $nghiem\_RKs$.
5. Tạo mảng $\overline{x} = x[].copy,\quad \overline{y} = y[].copy$
6. Cho biến chạy $i : s - 1 \to n - 1$, thực hiện:

   - **Nếu $s = 2$:**
     (a) Gán $t_{i+1} = t_i + h$
     (b) Sử dụng ABs tìm nghiệm dự đoán
     $$\overline{x_{i+1}} = x_i + \frac{h}{2} \times (3 \times f_i - f_{i-1})$$
     $$\overline{y_{i+1}} = y_i + \frac{h}{2} \times (3 \times g_i - g_{i-1})$$
     (c) Sử dụng AMs tìm nghiệm hiệu chỉnh
     $$x_{i+1} = x_i + \frac{h}{2} \times f(t_{i+1}, \overline{x_{i+1}},\ \overline{y_{i+1}}) + \frac{h}{2} \times f_i$$
     $$y_{i+1} = y_i + \frac{h}{2} \times g(t_{i+1},\ \overline{x_{i+1}},\ \overline{y_{i+1}}) + \frac{h}{2} \times g_i$$
     (d) Gán
     $$f_{i+1} = f(t_{i+1}, x_{i+1}, y_{i+1})$$
     $$g_{i+1} = g(t_{i+1}, x_{i+1}, y_{i+1})$$
   - **Nếu dùng $s = 3$:**
     (a) Gán $t_{i+1} = t_i + h$

(b) Sử dụng ABs tìm nghiệm dự đoán

$$\overline{x_{i+1}} = x_i + \frac{h}{12} \times (23 \times f_i - 16 \times f_{i-1} + 5 \times f_{i-2})$$

$$\overline{y_{i+1}} = y_i + \frac{h}{12} \times (23 \times g_i - 16 \times g_{i-1} + 5 \times g_{i-2})$$

(c) Sử dụng AMs tìm nghiệm hiệu chỉnh

$$x_{i+1} = x_i + \frac{h}{12} \times (5 \times f(t_{i+1},\ \overline{x_{i+1}},\ \overline{y_{i+1}}) + 8 \times f_n - f_{n-1})$$

$$y_{i+1} = y_i + \frac{h}{12} \times (5 \times g(t_{i+1},\ \overline{x_{i+1}},\ \overline{y_{i+1}}) + 8 \times g_n - g_{n-1})$$

(d) Gán

$$f_{i+1} = f(t_{i+1}, x_{i+1}, y_{i+1})$$
$$g_{i+1} = g(t_{i+1}, x_{i+1}, y_{i+1})$$

- **Nếu dùng $s = 4$:**

(a) Gán $t_{i+1} = t_i + h$

(b) Sử dụng ABs tìm nghiệm dự đoán

$$\overline{x_{i+1}} = x_i + \frac{h}{24} \times (55 \times f_i - 59 \times f_{i-1} + 37 \times f_{i-2} - 9 \times f_{i-3})$$

$$\overline{y_{i+1}} = y_i + \frac{h}{24} \times (55 \times g_i - 59 \times g_{i-1} + 37 \times g_{i-2} - 9 \times g_{i-3})$$

(c) Sử dụng AMs tìm nghiệm hiệu chỉnh

$$x_{i+1} = x_i + \frac{h}{24} \times (9 \times f(t_{i+1},\ \overline{x_{i+1}},\ \overline{y_{i+1}}) + 19 \times f_i - 5 \times f_{i-1} + f_{i-2})$$

$$y_{i+1} = y_i + \frac{h}{24} \times (9 \times g(t_{i+1},\ \overline{x_{i+1}},\ \overline{y_{i+1}}) + 19 \times g_i - 5 \times g_{i-1} + g_{i-2})$$

(d) Gán

$$f_{i+1} = f(t_{i+1}, x_{i+1}, y_{i+1})$$
$$g_{i+1} = g(t_{i+1}, x_{i+1}, y_{i+1})$$

- **Nếu dùng $s = 5$:**

(a) Gán $t_{i+1} = t_i + h$

(b) Sử dụng ABs tìm nghiệm dự đoán

$$\overline{x_{i+1}} = x_i + \frac{h}{720} \times (1901 \times f_i - 2774 \times f_{i-1} + 2616 \times f_{i-2} - 1274 \times f_{i-3} + 251 \times f_{i-4})$$

$$\overline{y_{i+1}} = y_i + \frac{h}{720} \times (1901 \times g_i - 2774 \times g_{i-1} + 2616 \times g_{i-2} - 1274 \times g_{i-3} + 251 \times g_{i-4})$$

(c) Sử dụng AMs tìm nghiệm hiệu chỉnh

$$x_{i+1} = x_i + \frac{h}{720} \times (251 \times f(t_{i+1}, \overline{x_{i+1}}, \overline{y_{i+1}}) + 646 \times f_i - 264 \times f_{i-1} + 106 \times f_{i-2} - 19 \times f_{i-3})$$

$$y_{i+1} = y_i + \frac{h}{720} \times (251 \times g(t_{i+1}, \overline{x_{i+1}}, \overline{y_{i+1}}) + 646 \times g_i - 264 \times g_{i-1} + 106 \times g_{i-2} - 19 \times g_{i-3})$$

(d) Gán

$$f_{i+1} = f(t_{i+1}, x_{i+1}, y_{i+1})$$
$$g_{i+1} = g(t_{i+1}, x_{i+1}, y_{i+1})$$

7. Trả về bộ biến $t[]$ và bộ giá trị $x[]$.