**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC**

**A picture containing text, sign

Description automatically generated**

Báo cáo: Kĩ thuật lập trình

Đề tài: Viết chương trình giải gần đúng phương trình f(x) = 0 [ f(x) là đa thức]

Giảng viên hướng dẫn: Cô Nguyễn Thị Thanh Huyền

Nhóm sinh viên thực hiện:

Nguyễn Công Hiếu - 20195016

Phạm Văn Thành - 20195918

Lớp: Toán – Tin 02 – K64

Mã nhóm báo cáo: 31

Hà Nội, Tháng 5 Năm 2021

MỤC LỤC

[Mở đầu 3](#_Toc74527809)

[BÀI TOÁN ĐẶT RA 5](#_Toc74527810)

[MỤC I: PHÂN TÍCH CHƯƠNG TRÌNH 6](#_Toc74527811)

[MỤC II: QUÁ TRÌNH THIẾT KẾ 8](#_Toc74527812)

[ MODULE 0 : Xác định đa thức 8](#_Toc74527813)

[ MODULE 1: Tìm miền chứa nghiệm 10](#_Toc74527814)

[ MODULE 2: Khoảng phân li nghiệm của phương pháp chia đôi thỏa mãn: 14](#_Toc74527815)

[|a - b| < 0.5. 14](#_Toc74527816)

[ **MODULE 3: Tìm nghiệm dựa trên số lần lặp N** 23](#_Toc74527817)

[ **MODULE 4: Tìm nghiệm với epsilon cho trước** 28](#_Toc74527818)

[ **MODULE 5: Tìm nghiệm thỏa mãn: ** 30](#_Toc74527819)

[**MỤC III: CÀI ĐẶT CHƯƠNG TRÌNH** 31](#_Toc74527820)

[**1.** **Khai báo thư viện và khai báo nguyên mẫu hàm (header.h)** 31](#_Toc74527821)

[**2.** **Chương trình chính (main.cpp)** 33](#_Toc74527822)

[**3.** **Ảnh source code** 53](#_Toc74527823)

[3.1. <Header.h> 53](#_Toc74527824)

[3.2. <Main.cpp> 56](#_Toc74527828)

[Mục IV: Kiểm thử chương trình 74](#_Toc74527829)

[**0.** **MODULE nhập vào đa thức** 74](#_Toc74527830)

[**1.** **Giao diện người dùng** 74](#_Toc74527831)

[**1.** **MODULE lựa chọn in ra miền nghiệm.** 75](#_Toc74527832)

[**2.** **MODULE khoảng phân ly nghiệm của phương trình sau khi chia đôi.** 75](#_Toc74527833)

[**3.** **MODULE tìm nghiệm với số lần lặp N** 76](#_Toc74527834)

[**4.** **MODULE tìm nghiệm với epsilon cho trước** 77](#_Toc74527835)

[**5.** **Lựa chọn tìm nghiệm thỏa mãn:**  77](#_Toc74527836)

[**6.** **Hiển thị ra file** 78](#_Toc74527837)

[Ví dụ 1: Da thuc f(x) = 8461x^5 + -973x^4 + 9341x^3 + 1253x^2 + -8753x^1 + -454 78](#_Toc74527838)

[Ví dụ 2: Da thuc f(x) = 214x^7 + 273x^6 + 87.213x^5 + 2452.45x^4 + 3154x^3 + 23.54x^2 + 125.364x^1 + 123.1 81](#_Toc74527839)

[Ví dụ 3: Da thuc f(x) = 1x^5 + 7x^4 + 14x^3 + 2x^2 + -15x^1 + -9 84](#_Toc74527840)

# Mở đầu

Hiện nay toán học có vai trò hết sức to lớn trong cuộc sống hiện đại. Toán học có ở khắp mọi nơi, thậm trí nó thấm nhuần vào trong suy nghĩ của mỗi chúng ta đến mức quên đi rằng dù chúng ta bước đi thì toán cũng chứa trong đó. Tuy nhiên đó là những thứ đơn giản và nếu để nghiên cứu sâu hơn về Toán thì chắc chắn chúng ta cần phải có những công cụ hỗ trợ.

Vì thế trong báo cáo này chúng em đã xây dựng một chương trình trong đó áp dụng các cơ sở toán học để thực hiện việc giải phương trình f(x) = 0, f(x) là đa thức. Có thể chương trình sẽ là một công cụ giúp ích một phần nào đấy trong việc tìm nghiệm của phương trình góp phần cho việc nghiên cứu chuyên sâu.

Do thời gian làm báo cáo chỉ trong một thời gian ngắn nên việc nâng cấp chương trình vẫn có hạn chế.

# BÀI TOÁN ĐẶT RA

Bài 1. Viết chương trình giải gần đúng phương trình f(x) = 0 (f(x) là đa thức) bằng phương pháp dây cung. Thực hiện các yêu cầu sau: 1) Tìm các miền chứa nghiệm của phương trình.

2) Tìm khoảng phân ly nghiệm (a, b) của phương trình thoả mãn |a − b| ≤ 0,5 bằng cách sử dụng phương pháp chia đôi để thu hẹp dần một khoảng phân ly nghiệm đã tìm được ở ý 1).

3) Tìm nghiệm gần đúng với số lần lặp n cho trước trong khoảng phân ly nghiệm (a,b) và đánh giá sai số theo cả hai công thức (n được nhập vào từ bàn phím, (a,b) có thể lấy từ kết quả của ý 2) hoặc được nhập vào từ bàn phím).

4) Tìm nghiệm gần đúng trong khoảng (a, b) với sai số e cho trước (e được nhập vào từ bàn phím, (a, b) có thể lấy từ kết quả của ý 2) hoặc được nhập vào từ bàn phím). Tính toán theo 2 cách áp dụng công thức sai số.

5) Tìm nghiệm gần đúng xn trong khoảng (a, b) thoả mãn điều kiện: |xn − xn−1| ≤ e (e được nhập vào từ bàn phím).

# MỤC I: PHÂN TÍCH CHƯƠNG TRÌNH

Quy trình phân tích chương trình

TOP - DOWN

(6)

Bảng giao diện người dùng điều khiển

(1)

Tìm miền chứa nghiệm

(2)

Các khoảng phân li nghiệm của phương pháp chia đôi thỏa mãn:

|a - b| < 0.5

(5)

Tìm nghiệm thỏa mãn:



(4)

Tìm nghiệm với  cho trước

(3)

Tìm nghiệm dựa trên số lần lặp N

Xây dựng chương trình tìm nghiệm gần đúng của f(x) đa thức

(1.1)

Cận trên của miền nghiệm:



(1.2)

Cận dưới của miền nghiệm:



(1.2.1)

Kiểm tra hệ số âm đầu tiên của đa thức là k

Và

B là giá trị Max của trị tuyệt đối các hệ số âm trong đa thức

(0)

Đầu vào đa thức f(x)

(3.2)

Tìm m1 và M1 là min, max của đạo hàm để đánh giá sai số

(2.1)

Trang bên

(3.1)

Công thức lặp

(3.3)

Tìm điểm fourier

Áp dụng 3.1, 3.2, 3,3 cho MODULE 4

Áp dụng 3.1, 3,3 cho MODULE 5

(2.2.1.1)

Tìm các điểm cực trị

(2.1)

Phương pháp chia đôi

|a – b| < 0.5

(2.2)

Các khoảng phân li nghiệm

(2.2.1)

Các điểm cực trị trái dấu

(2.2.1.2)

Kiểm tra hai điểm cực trị liên tiếp trái dấu

(2.2.1.1.1)

Thuật toán gradient descent để tìm cực trị

(2.2.1.1.1.2)

Tính giá trị f(x)

tại một điểm

(2.2.1.1.1.1)

Tính giá trị f’(x)

tại một điểm

# MỤC II: QUÁ TRÌNH THIẾT KẾ

**Quá trình thiết kế**

**BOTTOM \_ UP**

*Đi ngược từ dưới lên của các MODULE để hoàn thành chương trình*

* MODULE 0 : Xác định đa thức

1. Đầu vào: n là bậc của đa thức,  là hệ số thứ i của đa thức bậc n, i =
2. Đầu ra : Mảng chứa các hệ số  của đa thức.
3. Ý tưởng:

Ban đầu ta cần phải xác định đa thức đầu vào là gì để giải quyết các vấn đề tiếp theo.

Cho người dùng nhập vào bậc của đa thức.

Chỉ cho phép họ thao tác với các đa thức có bậc nhỏ hơn 50 và lớn hơn 0.

Tiếp theo là nhập vào các hệ số của đa thức.

Lưu các hệ số của đa thức lại để thao tác sau này.

1. Function khởi tạo đa thức

Input\_poly(int n, Double poly[] , Double der[], Double der2[])

Begin

//poly[]: mảng chứa các hệ số của đa thức

input 🡨 n

while n smaller than 0 or n bigger than 50:

{

input 🡨 n

}

end while;

For i 🡨 n to 0:

{

Add 

Poly[i] 🡨 

}

end for;

For i🡨n - 1 to 0:

{

Der[i] 🡨poly[i+1] \*(i+1)

}

End for;

For i🡨n - 2 to 0:

{

Der2[i] 🡨Der[i+1] \*(i+1)

}

End for;

END;

* MODULE 1: Tìm miền chứa nghiệm

Để hoàn thành MODULE 1 ta thực hiện từ dưới lên trên để hoàn thành các MODULE cây phân nhánh của nó.

* MODULE1.2.1. B là giá trị MAX của trị tuyệt đối các hệ số âm của đa thức.

1. Đầu vào: Hệ số  của đa thức hay poly[], , i =, n là bậc của đa thức
2. Đầu ra : Là giá trị B Max của trị tuyệt đối các hệ số âm trong đa thức.
3. Ý tưởng:

Ta duyệt từ hệ số  đến hệ số  để tìm hệ số âm đầu tiên của đa thức

Nếu gặp hệ số âm thì lưu lưu lại hệ số đó là k.

Ta kiểm tra xem hệ số  có lớn hơn 0 hay không.

 là hệ số của 

Nếu  lớn hơn không ta sẽ duyệt từ  đến  để tìm các hệ số âm của đa thức.

Tạo ra một mảng trung gian để lưu các giá trị chứa các hệ số âm và vị trí của nó.

Mỗi một phần tử nhỏ hơn 0 sẽ dc thêm vào mảng.

Nếu  nhỏ hơn không ra sẽ đổi dấu hệ số của các đa thức có bậc lẻ

Rồi làm tương tự như đối với  lớn hơn không.

Sau khi duyệt hết và có các hệ số âm ta so sánh xem trị tuyệt đối của hệ số nào lớn nhất rồi trả về B.

1. Function tìm giá trị MAX của trị tuyệt đối các hệ số âm

Begin

integer k //k là chỉ số để xác định

For i 🡨 0 to n:

{

If poly[i] smaller than 0:

{

k 🡨 i

break;

}

End if;

End for;

Double temp[] //Mảng lưu các giá trị âm của đa thức

If  bigger than 0:

{

For I 🡨 1 to n: //n là bậc của đa thức

{

If  smaller than 0:

{

Temp[] 🡨 

}

End if;

}

End for;

}

End if;

Else if  smaller than 0:

{

For I 🡨 0 to n:

{

If i chia dư cho 2 dư 1:

{

 🡨 (-1) \*

}

}

End for;

For i 🡨 1 to n:

{

If  smaller than 0:

{

Temp[i] 🡨 

}

End if;

End for;

}

End else;

Double B 🡨poly[k]

For I 🡨 1 to n:

{

If |temp[i] bigger than |B|:

{

B 🡨 |temp[i]|

}

End if;

End for;

Return B;

End;

* MODULE 1.1 & 1.2: Cận dưới và cận trên

1. Đầu vào : k, B đã được tìm ở (1.2.1) ,  hệ số đầu tiên của đa thức.
2. Đầu ra: Cận dưới và cận trên chứa miền nghiệm của đa thức.
3. Ý tưởng: Gọi lại hàm Tìm B để tính lower\_bound và Upper\_bound

Lower\_bound =  //Cận dưới

Upper\_bound =  //Cận trên

\*Note: Như vậy ta có miền chứa nghiệm: (Lower\_bound, Upper\_bound) và hoàn thành MODULE 1.

* MODULE 2: Khoảng phân li nghiệm của phương pháp chia đôi thỏa mãn:

|a - b| < 0.5.

Để hoàn thành MODULE 2 ta cũng đi từ dưới lên và hoàn thành các cây phân cấp của nó.

* MODULE 2.2.1.1.1.1 Tính giá trị f’().

1. Đầu vào: tham số , hệ số của đa thức f(x), n là bậc của đa thức.
2. Đầu ra: giá trị của f’() tại .
3. Function Tình giá trị của f’().

f1(Double poly[], Double , integer n)

//poly[]: là hệ số của đa thức

// : cần tính đạo hàm tại 

Begin

For i 🡨n to 0:

{

f1 🡨  //pow(,n-1):  mũ n-1

}

//f1: là giá trị của của đạo hàm tại x0

End for;

Return f1;

End;

* MODULE 2.2.1.1.1.2. Tính giá trị f().

1. Đầu vào: tham số x, hệ số của đa thức f(x), n là bậc của đa thức.
2. Đầu ra: giá trị của f() tại .
3. Function Tình giá trị của f().

f(Double poly[], Double , integer n)

//poly[]: là hệ số của đa thức

Begin

For i 🡨n to 0:

{

f 🡨 //pow(,n):  mũ n

}

//f: là giá trị của hàm số tại 

End for;

Return f;

End;

* MODULE2.2.1.1.1: Thuật toán Gradient descent.

Để hoàn thành MODULE 2.2.1.1.1 ta cần sử dụng đến MODULE 2.2.1.1.1.1 và MODULE 2.2.1.1.1.2 ở trên.

1. Đầu vào: x là nơi bắt đầu để kiểm tra đối đa thức đã định nghĩa.
2. Đầu ra: Điểm cực trị
3. Ý tưởng:

Công thức của gradient descent như sau:

**** //sign: dấu của đạo hàm tại x

Ta bắt đầu duyệt từ một điểm được chọn và duyệt tới khi nào đạo hàm của f(x) nhỏ hơn 10e-15 thì dừng.

Eta là một số ta tự định nghĩa. Sao cho eta không quá lớn và không quá bé.

Nếu |f’(x)\*eta| lớn hơn 0.2 thì f’(x)\*eta = 0.2\*sign

Và để tránh lặp vô hạn khi không tìm thấy cực trị ta sẽ thêm vào một phần tử chốt chặn pivot để kết thúc lặp.

Khi tìm được thì trả về điểm cực trị đã tìm dc

1. Function thi triển thuật toán gradient descent

Gradient descent(Double upper\_bound, Double lower\_bound, Double der[], Double x, Double poly[], integer n, Double eta)

Begin

integer Sign //Biến kiểm tra dấu đạo hàm

Double  //Nơi bắt đầu kiểm tra

integer pivot //pivot: tránh lặp vô hạn

Double x\_new // lưu điểm cực trị

If f1(poly,, n) bigger than 0:

Sign = 1;

Else if f1(x) smaller than 0:

Sign = -1

Else

Sign = 0

while pivot smaller than 300000:

{

//eta là giá trị tự định nghĩa để giúp x tiến đến x-eta.f1(x)

//eta có thể thay đổi, eta không nên quá lớn hoặc quá nhỏ

If |eta\* f1(poly, , n)| bigger than 0.2:

{

x🡨x+Sign\*0.2;

}

End if;

X\_new 🡨x +sign\*eta\* f1(poly,,n)

If f1(poly, , n) smaller than 5e-15:

End while;

}

 🡨x\_new;

Pivot 🡨pivot +1

}

End while;

Return x

End;

* MODULE2.2.1.1: Tìm các điểm cực trị.

1. Đầu vào: Miền chứa nghiệm của đa thức đã tìm ở (1)
2. Đầu ra : Các điểm cực trị.
3. Ý tưởng:

Ta bắt đầu từ điểm cận dưới chạy thuật toán gradient descent đến khi gặp điểm cận trên thì dừng

Mỗi lần tìm được một cực trị thì ta sẽ cho điểm đó + với 1 hằng số để vượt quá điểm cực trị đã tìm dc. Khi đó ta có thể tìm các điểm cực trị phía sau.

Lưu các điểm cực trị lại.

1. Function tim các điểm cực trị của đa thức

all\_critical\_points(Double lower\_bound, Double upper\_bound, Double poly[], Double der[], integer n, Double eta)

Begin

Double Local[] //local: lưu các giá trị cực trị

Double weird[] //weird: lưu nghiệm kì dị

x 🡨 lower\_bound

Local add lower\_bound

while x smaller than upper\_bound:

{

x 🡨 gradient\_descent(x)

if x smaller than upper\_bound and |f(x,poly,n)| defferent 0:

{

Local add x

}

End if;

If |f(x,poly,n)| is 0:

{

Weird add x

}

End if;

x 🡨x+0.0001

}

End while;

Local add upper\_bound

Rerurn local

End;

**MODULE2.2.1.Tìm các điểm cực trị trái dấu.**

Dựa vào MODULE 2.2.1.1 và MODULE 2.2.1.2 ta xây dựng lên MODULE 2.2.1

1. Đầu vào: Các điểm cực trị.
2. Đầu ra : Bộ các điểm cực trị trái dấu.
3. Ý tưởng:

Kiểm tra lần lượt các giá trị của các điểm cực trị

Nếu 2 điểm cực trị gần nhau mà trái dấu nhau thì đó là khoảng phân li nghiệm.

Khi tìm hết các khoảng phân li thì trả về bộ các điểm cực trị trái dấu.

1. Function tìm các điểm cực trị trái dấu.

root\_interval(Double lower\_bound, Double upper\_bound, Double poly[], Double der[], integer n, Double eta)

//Local là biến có kiểu dữ liệu lưu các giá trị của các cực trị.

Begin

Double Interval

//Interval: là biến có kiểu dữ liệu lưu trữ được 2 điểm liên tiếp.

For i 🡨0 to size local:

//local đã tìm được ở MODULE 2.2.1.1

{

If f( Local[i]) \* f(Local[i+1]) smaller than 0:

{

Interval add local[i]

Interval add local[i+1]

}

End for;

End;

**\*Note: Sau khi tìm được 2 điểm cực trị liên tiếp trái dấu thì khi đó đã hoàn thành MODULE 2.2: Các khoảng phân li nghiệm**

**MODULE2.1: Phương pháp chia đôi thỏa mãn |a-b| < 0.5**

1. Đầu vào : a, b là đầu mút của khoảng phân li nghiệm
2. Đầu ra: Rút ngắn được khoảng phân ly nghiệm a,b
3. Ý tưởng:

Với a và b là hai đầu mút của khoảng phân li nghiệm thì

f(a) \* f(b) < 0, f(x) đơn điệu trong khoảng [a,b] và hàm số có đạo hàm liên tục trong khoảng[a,b]

Dõ dàng với việc đã tìm được các điểm cực trị trái dấu cạnh nhau thì f(x) đơn điệu trên các khoảng phân ly nghiệm.

Ta thực hiện chia đôi.

, kiểm tra xem f(a)\*f(c)

* Nếu f(a)\*f(c) < 0 thì b = c
* Nếu f(a)\*f(c) > 0 thì a = c

Lặp đến khi nào thỏa mãn |a-b| < 0.5 thì dừng

Trả về khoảng phân li nghiệm mới

4.Function phương pháp chia đôi

bisection(a,b)

Begin

While |a-b| smaller than 0.5:

{

c🡨

If f(a) \* f(c) smailer than 0:

{

b 🡨 c

}

End if;

Else if f(a) \* f(c) bigger than 0:

{

a 🡨c

}

End elseif;

Else

{

//c là nghiệm bội

Break;

}

End else;

End;

**\*Note: Từ MODULE2.1 và MODULE2.2 kết hợp lại ta có được MODULE 2 là các khoảng phân li nghiệm của phương pháp chia đôi thỏa mãn |a-b| < 0.5**

# **MODULE 3: Tìm nghiệm dựa trên số lần lặp N**

* MODULE 3.1: Công thức lặp



1. Đầu vào: , d //điểm fourier
2. Đầu ra : x sau n lần lặp
3. Function regula

Regula(Double x0, Double poly[], Double der[], Double n)



* **MODULE 3.2: Tìm m1 và M1 là min, Max của đạo hàm để đánh giá sai số**

1. Đầu vào: với khoảng phân ly nghiệm [a,b] được chọn
2. Đầu ra : m1 và M1
3. Ý tưởng:

Xác định dấu của đạo hàm cấp 2 f2.

Nếu f2 lớn hơn 0 và f1 lớn hơn 0 thì m1 = đạo hàm cấp 1 tại a

Nếu f2 lớn hơn 0 và f1 nhỏ hơn 0 thì m1 = (-1)\* đạo hàm cấp 1 tại b

Nếu f2 nhỏ hơn 0 và đạo hàm cấp 1 tại a lớn hơn 0 thì m1 = đạo hàm cấp 1 tại b

Còn lại thì m1 = (-1)\* đạo hàm cấp 1 tại a

Kiểm tra xem nếu m1 là đạo hàm cấp 1 tại a thì M1 là trị tuyệt đối đạo hàm cấp 1 tại b

Và ngược lại

1. Hàm tìm m1 và M1 là min, Max của đạo hàm để đánh giá sai số

m1\_M1( Double der[], Double der2[], integer n, Double a, Double b)

Begin

//a ,b : là khoảng phân ly nghiệm

f2(a,der,n)

If f2 bigger than 0:

Sign 🡨1

Else if f2 smaller than 0:

Sign 🡨-1

If sign is 1 and f1(a,der,n) bigger than 0:

{

m1 🡨 f1(a,der,n)

}

End if;

Else if sign is 1 and f1(a,der,n) smaller than 0:

{

m1 🡨 (-1)\*f1(b,poly,n)

}

End else;

Else if sign is -1 and f1(a,der,n) bigger than 0:

{

m1 🡨f1(b,der,n)

}

End else;

Else

{

m1 🡨 f1(a,der,n)

}

End else;

If m1 is | f1(a,der,n)| :

{

M1 🡨f1(b,der,n)

}

Else

{

M1🡨 f1(a,der,n)

}

**MODULE 3.3: Tìm điểm fourier**

1. Đầu vào: a , b là điểm phân ly nghiệm
2. Đầu ra: , d là điểm fourier
3. Function tìm điểm fourier

Fourier\_point(Double poly[] , Double der2[], integer n, Double a, Double b)

Begin

Double 

Double d

If f(a,poly,n) \* f2(a,der,n) smaller than 0:

{

 🡨 a

d 🡨b

}

End if;

Else

{

 🡨 b

d 🡨a

}

End else;

* **MODULE 3: Tìm nghiệm theo số lần lặp**

1. Đầu vào: min, max của đạo hàm, điểm fourier
2. Đầu ra : Nghiệm sau N lần lặp, sai số mắc phải
3. Funtion tìm nghiệm

Interation\_one(Double m1, Double M1, Double poly[], Double der[], Double der2[], integer n]

Begin

integer max\_ite

double x\_old

double err

a 🡨 interval[][]

b 🡨 interval[][]

m1\_M1(m1,M1,poly,der,der2,n,a,b)

fourier\_point(, d, poly, der2,n,a,b)

max\_ite 🡨input //Nhập vào số lần lặp

for I 🡨 0 to max\_ite:

{

Regula(, d, poly, der, n)

If I is max\_ite – 2:

{

X\_old = x

}

}

Print 

If M1 smaller than 2\*m1:

{

err = | – x\_old|

print err

}

End if;

Else

{



Print err

}

End else;

# **MODULE 4: Tìm nghiệm với epsilon cho trước**

1. Đầu vào: epsilon, min, max của đạo hàm, điểm fourier
2. Đầu ra: Nghiệm của đa thức với sai số cho trước.
3. Function tìm nghiệp với epsilon cho trước

Epsilon\_one(Double m1, Double M1, Double poly[], Double der[], Double der2[], int n]

Begin

double epsilon

double x\_old

double c

a 🡨 interval[][]

b 🡨 interval[][]

c 🡨(M1-m1)/m1

m1\_M1(m1,M1,poly,der,der2,n,a,b)

fourier\_point(, d, poly, der2,n,a,b)

epsilon 🡨input //Nhập vào số lần lặp

do

{

x\_old 🡨x0

Regula(, d, poly, der, n)

}

while |f(x0,poly,n)|/m1 bigger than epsilon:

if c smaller than 1 :

{

do

{

x\_old 🡨x0

Regula(, d, poly, der, n)

}

while |f(x0,poly,n)|/m1 bigger than epsilon:

}

End if;

Else

do

{

x\_old 🡨 

Regula(, d, poly, der, n)

}

while c\* | -x\_old| bigger than epsilon:

end else;

Print 

# **MODULE 5: Tìm nghiệm thỏa mãn:**

1. Đầu vào: epsilon, điểm fourier
2. Đầu ra: Nghiệm của đa thức
3. Function Tìm nghiệm thỏa mãn: 

Last\_one(Double poly[], Double der[], Double der2[], int n)

Begin

double epsilon

double x\_old

a 🡨 interval[][]

b 🡨 interval[][]

fourier\_point(, d, poly, der2,n,a,b)

epsilon 🡨input

do

{

x\_old 🡨 

Regula(, d, poly, der, n)

}

while c| -x\_old| bigger than epsilon:

print 

# **MỤC III: CÀI ĐẶT CHƯƠNG TRÌNH**

1. **Khai báo thư viện và khai báo nguyên mẫu hàm (header.h)**

#include <iostream>

#include <Windows.h>

#include <conio.h>

#include <vector>

#include <math.h>

#include <iomanip>

#include <fstream>

#define eps 1.0e-7

#define step 1.0e-3

#define ESC 27

#define ENTER 13

#define UP 72

#define DOWN 80

#define RIGHT 77

#define LEFT 75

#define Somuc 5

using namespace std;

/\* khai bao nguyen mau ham\*/

// ham dau vao

void input(double poly[], double der[], double der2[], int& n);

void out\_poly(double poly[], int n, int xtt, int ytt);

// cac ham phu tro menu

void VeMenu(int mucchon, int xtt, int ytt, int xdp, int ydp, int maunen, int mauchu, int maumucchon, int nenmucchon, int count);

void vehcn(int xtt, int ytt, int xdp, int ydp, int chimuc, int count);

void veMenuchonkhoang(int count, int maumucchon, int maunen, int mucchon);

int sub\_menu(int count); // menu lua chon khoang nghiem

void color(int color); // mau

void gotoxy(int x, int y); // di den toa do (x, y) tren console

// cac ham tinh toan gia tri bieu thuc

double f(double x0, double poly[], int n); // tinh f(x0)

double f1(double x0, double der[], int n); // tinh f'(x0)

double f2(double x0, double der2[], int n); // tinh f''(x0)

// thuat toan tim can tren, can duoi

double B(double poly[], int n); // tim B

double upper(double& upper\_bound, double poly[], int n); // tim can tren

double lower(double& lower\_bound, double poly[], int n); // tim can duoi

// tim cuc tri voi gradient descent

int check(double x0, double der[], int n);

double gradient\_descent(double lower\_bound, double upper\_bound, double poly[], double der[], int n, double eta); // tim cuc tri dia phuong

void all\_critical\_points(double lower\_bound, double upper\_bound, double poly[], double der[], int n, double eta); // dua cuc tri vao trong vector local

void root\_interval(double lower\_bound, double upper\_bound, double poly[], double der[], int n, double eta, int& count); // tim khoang phan ly nghiem va dua vao vector intervals

void show\_intervals(int count); // in cac khoang nghiem

// thu hep khoang cach ly

void bisection(double poly[], int n, int count); // thu nho tat ca khoang nghiem trong vector intervals

void regula(double& x0, double d, double poly[], double der[], int n); // cong thuc lap

void m1\_M1(double& m1, double& M1, double poly[], double der[], double der2[], int n, int number); // tim m1 va M1 tren khoang ma ta chon

void fourier\_point(double& x0, double& d, double poly[], double der2[], int n, double a, double b);

void iteration\_one(double m1, double M1, double poly[], double der[], double der2[], int n, int count); // tim nghiem voi so lan lap cho truoc

void epsilon\_one(double m1, double M1, double poly[], double der[], double der2[], int n, int count); // tim nghiem voi epsilon

void last\_one(double m1, double M1, double poly[], double der[], double der2[], int n, int count); // tim nghiem theo ct hau nghiem

## **Chương trình chính (main.cpp)**

#include "header.h"

/\* khai bao bien toan cuc\*/

vector<double> intervals[100]; // vector chua cac khoang phan ly nghiem

vector<double> varX; // vector chua gia tri cac lan lap o buoc gradient descent

vector<double> local; // vector chua cac diem cuc tri

vector<double> weird; // vector chua cac nghiem ky di (nghiem boi) chua lam duoc

string menu[] = { "1. In ra mien chua nghiem", "2. In ra cac khoang phan ly nghiem"

, "3. In ra nghiem voi so lan lap cho truoc"

, "4. In ra nghiem voi epsilon cho truoc"

, "5. In ra nghiem theo cong thuc hau nghiem" };

int main() {

int xtt = 0, ytt = 0, xdp = xtt + 60, ydp = ytt + 19;

int maunen = 12, mauchu = 12, maumucchon = 15, nenmucchon = 15;

int mucchon = 0;

int phim;

int ans = -1; // bien dung chuong trinh (nang cap sau nay)

int n = 0; // bac da thuc

int count = 0; // so cac vector interval

double m1 = 0, M1 = 0;

double eta = 0.0001; // learning rate

double lower\_bound, upper\_bound;

double poly[100] = { 0 };

double der[100] = { 0 };

double der2[100] = { 0 };

while (count == 0) {

input(poly, der, der2, n);

system("cls");

upper\_bound = upper(upper\_bound, poly, n);

lower\_bound = lower(lower\_bound, poly, n);

root\_interval(lower\_bound, upper\_bound, poly, der, n, eta, count);

bisection(poly, n, count);

if (count == 0) {

std::cout << "The polynomial has no roots! Try another one!" << endl;

}

}

while (ans != 0) {

out\_poly(poly, n, xtt, ytt);

VeMenu(mucchon, xtt, ytt, xdp, ydp, maunen, mauchu, maumucchon, nenmucchon, count);

phim = \_getch(); // trong Visual Studio 2019 ko su dung dc getch()

switch (phim) {

case UP:

if (mucchon == 0) mucchon = Somuc - 1;

else mucchon--;

VeMenu(mucchon, xtt, ytt, xdp, ydp, maunen, mauchu, maumucchon, nenmucchon, count);

break;

case DOWN:

if (mucchon == Somuc - 1) mucchon = 0;

else mucchon++;

VeMenu(mucchon, xtt, ytt, xdp, ydp, maunen, mauchu, maumucchon, nenmucchon, count);

break;

case ENTER:

system("cls");

switch (mucchon) {

case 0:

std::cout << " Roots are in range: " << endl;

std::cout << " Lower bound: " << lower\_bound << endl;

std::cout << " Upper bound: " << upper\_bound << endl;

break;

case 1:

show\_intervals(count);

break;

case 2:

iteration\_one(m1, M1, poly, der, der2, n, count);

break;

case 3:

epsilon\_one(m1, M1, poly, der, der2, n, count);

break;

case 4:

color(12);

last\_one(m1, M1, poly, der, der2, n, count);

break;

}

std::cout << endl;

system("pause");

system("cls");

break;

case ESC:

system("cls");

vehcn(0, 0, 50, 8, Somuc + 3, count);

gotoxy(0, 9);

return -1;

}

}

system("cls");

return 0;

}

void VeMenu(int mucchon, int xtt, int ytt, int xdp, int ydp, int maunen, int mauchu, int maumucchon, int nenmucchon, int count) {

color(maunen);

// dinh nghia cua so

gotoxy(xtt, ytt);

std::cout << char(218);

for (int i = ytt + 1; i < ydp; i++) {

gotoxy(xtt, i);

if (i == ytt + 3) {

std::cout << char(195);

}

else {

std::cout << char(179);

}

}

gotoxy(xtt, ydp);

std::cout << char(192);

for (int i = xtt + 1; i < xdp; i++) {

gotoxy(i, ytt);

std::cout << char(196);

}

for (int i = xtt + 1; i < xdp; i++) {

gotoxy(i, ydp);

if (i == 3 \* ((xdp + xtt) / 4) - 1) {

std::cout << char(193);

}

else {

std::cout << char(196);

}

}

gotoxy(xdp, ytt);

std::cout << char(191);

for (int i = ytt + 1; i < ydp; i++) {

gotoxy(xdp, i);

if (i == ytt + 3) {

std::cout << char(180);

}

else {

std::cout << char(179);

}

}

gotoxy(xdp, ydp);

std::cout << char(217);

for (int i = xtt + 1; i < xdp; i++) {

gotoxy(i, ytt + 3);

if (i == 3 \* ((xdp + xtt) / 4) - 1) {

std::cout << char(194);

}

else {

std::cout << char(196);

}

}

for (int i = ytt + 4; i < ydp; i++) {

gotoxy(3 \* ((xdp + xtt) / 4) - 1, i);

std::cout << char(179);

}

gotoxy(5 \* ((xtt + xdp) / 6) - 1, 1 \* ((ytt + ydp) / 2) + 1);

std::cout << "To Exit";

gotoxy(5 \* ((xtt + xdp) / 6) - 2, 1 \* ((ytt + ydp) / 2) + 2);

std::cout << "Press ESC";

// dat mau chu

for (int i = 0; i < Somuc; i++) {

if (i == mucchon) {

color(nenmucchon);

color(maumucchon);

}

else {

color(maunen);

color(mauchu);

}

// dinh nghia cua so

vehcn(xtt + 1, ytt + 3 \* i + 4, 43, ytt + 3 \* i + 6, i, count);

}

}

void vehcn(int xtt, int ytt, int xdp, int ydp, int chimuc, int count) {

static int j = count - 1;

gotoxy(xtt, ytt);

std::cout << char(218);

for (int i = ytt + 1; i < ydp; i++) {

gotoxy(xtt, i);

std::cout << char(179);

gotoxy(xdp, i);

std::cout << char(179);

}

gotoxy(xtt, ydp);

std::cout << char(192);

for (int i = xtt + 1; i < xdp; i++) {

gotoxy(i, ytt);

std::cout << char(196);

gotoxy(i, ydp);

std::cout << char(196);

}

gotoxy(xdp, ytt);

std::cout << char(191);

gotoxy(xdp, ydp);

std::cout << char(217);

// in text

gotoxy(xtt + 1, (ytt + ydp) / 2);

if (chimuc >= 0 && chimuc <= Somuc) {

std::cout << menu[chimuc];

}

if (chimuc == Somuc + 1) {

std::cout << setw(25) << "Choose your interval!";

}

if (chimuc == Somuc + 2) {

if (j < 0) {

j = count - 1;

}

std::cout << j + 1 << ". (" << intervals[j][0] << "; " << intervals[j][1] << ")";

j--;

}

if (chimuc == Somuc + 3) {

std::cout << setw(35) << "CHUONG TRINH DA KET THUC!";

}

}

void veMenuchonkhoang(int count, int maumucchon, int maunen, int mucchon) {

for (int i = 0; i < count; i++) {

if (i == mucchon) {

color(maumucchon);

}

else {

color(maunen);

}

vehcn(0, 2 \* (count - i), 30, 2 \* (count - i + 1), Somuc + 2, count);

}

color(maunen);

vehcn(0, 0, 30, 2, Somuc + 1, count);

}

int sub\_menu(int count) {

int choose = 0;

int mucchon = count - 1;

int maumucchon = 15, maunen = 12;

int phim;

while (1) {

veMenuchonkhoang(count, maumucchon, maunen, mucchon);

phim = \_getch();

switch (phim) {

case DOWN:

if (mucchon == 0) mucchon = count - 1;

else mucchon--;

veMenuchonkhoang(count, maumucchon, maunen, mucchon);

break;

case UP:

if (mucchon == count - 1) mucchon = 0;

else mucchon++;

veMenuchonkhoang(count, maumucchon, maunen, mucchon);

break;

case ENTER:

system("cls");

return count - mucchon - 1;

}

}

}

void color(int color) {

SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE), color);

}

void gotoxy(int x, int y) {

COORD c;

c.X = x;

c.Y = y;

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), c);

}

void input(double poly[], double der[], double der2[], int& n) {

fstream file;

file.open("test.txt", ios::out);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

return;

}

std::cout << "Degree of the polynomial: ";

do {

std::cin >> n;

} while (n < 0 || n > 50); // chi cho phep bac da thuc nam trong khoang [0, 50]

for (int i = n; i >= 0; i--) {

std::cout << "Coefficient of x^" << i << ": ";

std::cin >> poly[i];

}

for (int i = n - 1; i >= 0; i--) {

der[i] = poly[i + 1] \* (i + 1.0); // he so f'(x)

}

for (int i = n - 2; i >= 0; i--) {

der2[i] = der[i + 1] \* (i + 1.0); // he so f''(x)

}

file << "Bac cua da thuc: " << n << endl;

file << "Da thuc f(x) = ";

for (int i = n; i >= 0; i--) {

if (i != 0) {

file << poly[i] << "x^" << i << " + ";

}

else {

file << poly[i];

}

}

file << endl << "Da thuc f'(x) = ";

for (int i = n - 1; i >= 0; i--) {

if (i != 0) {

file << der[i] << "x^" << i << " + ";

}

else {

file << der[i];

}

}

file << endl << "Da thuc f''(x) = ";

for (int i = n - 2; i >= 0; i--) {

if (i != 0) {

file << der2[i] << "x^" << i << " + ";

}

else {

file << der2[i];

}

}

file << endl << "--------------------------------------" << endl;

file.close();

}

void out\_poly(double poly[], int n, int xtt, int ytt) {

color(9);

gotoxy(xtt + 1, ytt + 1);

std::cout << setw(35);

std::cout << "THE POLYNOMIAL" << endl;

gotoxy(xtt + 1, ytt + 2);

std::cout << "f(x): ";

std::cout << poly[n] << "x^" << n;

for (int i = n - 1; i >= 0; i--) {

if (i != 0) {

if (poly[i] >= 0) {

std::cout << " + " << poly[i] << "x^" << i;

}

else {

std::cout << " - " << fabs(poly[i]) << "x^" << i;

}

}

else {

if (poly[i] >= 0) {

std::cout << " + " << poly[i];

}

else {

std::cout << " - " << fabs(poly[i]);

}

}

}

std::cout << endl;

}

double f(double x0, double poly[], int n) {

double res = poly[n];

for (int i = n - 1; i >= 0; i--) {

res = res \* x0 + poly[i];

}

return res;

}

double f1(double x0, double der[], int n) {

return f(x0, der, n - 1);

}

double f2(double x0, double der2[], int n) {

return f1(x0, der2, n - 1);

}

double B(double poly[], int n) {

static int j = 0;

int k = 0; // di tim chi so cua he so am dau tien

double B = 0;

bool flag = false;

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

exit(0);

}

if (poly[n] > 0) {

for (int i = 0; i <= n - 1; i++) {

if (poly[i] < 0) {

flag = true;

k = i;

}

}

if (flag == true) {

B = poly[k];

for (int i = n - 2; i >= 0; i--) {

if ((poly[i] < 0) && (fabs(B) < fabs(poly[i]))) {

B = poly[i];

}

}

}

}

else if (poly[n] < 0) {

for (int i = 0; i <= n; i++) {

poly[i] = (-1) \* poly[i];

}

for (int i = 0; i <= n - 1; i++) {

if (poly[i] < 0) {

flag = true;

k = i;

}

}

if (flag == true) {

B = poly[k];

for (int i = n - 2; i >= 0; i--) {

if ((poly[i] < 0) && (fabs(B) < fabs(poly[i]))) {

B = poly[i];

}

}

}

}

else {

std::cout << endl << "The coeficient of the polynomial must not equal to 0";

exit(0);

}

if (j == 0) {

file << "+ Doi voi can tren:" << endl;

file << "He so am dau tien: k = " << 5 - k << endl;

file << "Max cua cac tri tuyet doi cua cac he so am la: B = " << fabs(B) << endl;

j++;

}

else {

file << endl << "+ Doi voi can duoi: " << endl;;

file << "He so am dau tien: k = " << 5 - k << endl;

file << "Max cua cac tri tuyet doi cua cac he so am la: B = " << fabs(B);

}

file.close();

return fabs(B);

}

double upper(double& upper\_bound, double poly[], int n) {

double temp[100];

static int j = 0;

int k = 0; // he so am dau tien

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

exit(0);

}

if (poly[n] > 0) {

for (int i = 0; i <= n - 1; i++) {

if (poly[i] < 0) {

k = n - i;

}

}

upper\_bound = 1 + pow(B(poly, n) / poly[n], (double)1 / k);

}

else if (poly[n] < 0) {

for (int i = 0; i <= n; i++) {

temp[i] = (-1) \* poly[i];

}

for (int i = 0; i <= n - 1; i++) {

if (temp[i] < 0) {

k = n - i;

}

}

upper\_bound = 1 + pow(B(temp, n) / ((-1) \* poly[n]), (double)1 / k);

}

else {

std::cout << "Invalid input!";

exit(0);

}

if (j == 0) {

file << "Can tren cua mien chua nghiem: " << upper\_bound;

j++;

}

file.close();

return upper\_bound;

}

double lower(double& lower\_bound, double poly[], int n) {

double temp[100] = { 0 };

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

exit(0);

}

for (int i = 0; i <= n; i++) {

if (i % 2 == 0) {

temp[i] = poly[i];

}

else {

temp[i] = (-1) \* poly[i];

}

}

if (n <= 50 && n >= 0) {

if (temp[n] < 0) {

for (int i = 0; i <= n; i++) {

temp[i] \*= (-1);

}

lower\_bound = (-1) \* upper(lower\_bound, temp, n);

}

else {

lower\_bound = (-1) \* upper(lower\_bound, temp, n);

}

}

else {

exit(0);

}

file << endl << "Can duoi cua mien chua nghiem la: " << lower\_bound;

file << endl << "--------------------------------------" << endl;

file.close();

return lower\_bound;

}

int check(double x0, double der[], int n) {

if (f1(x0, der, n) == 0) {

return 0;

}

else if (f1(x0, der, n) > 0) {

return 1;

}

else {

return -1;

}

}

double gradient\_descent(double lower\_bound, double upper\_bound, double poly[], double der[], int n, double eta) {

double x\_new, x0 = lower\_bound;

double deltaX;

int sign = check(x0, der, n);

int it = 0;

varX.push\_back(x0);

while (it <= 300000) {

deltaX = eta \* f1(x0, der, n);

// ngan khong cho x chay qua nhanh

if (deltaX > 0.2) {

deltaX = 0.2;

}

if (deltaX < -0.2) {

deltaX = -0.2;

}

x\_new = x0 + sign \* deltaX;

if (fabs(f1(x\_new, der, n)) < 1.0e-20 || x\_new > upper\_bound) {

break;

}

if (f1(x\_new, der, n) \* f1(x0, der, n) > 0) {

eta \*= 3;

}

if (f1(x\_new, der, n) \* f1(x0, der, n) < 0) {

eta /= 5;

}

x0 = x\_new;

varX.push\_back(x\_new);

it++;

}

return x\_new;

}

void all\_critical\_points(double lower\_bound, double upper\_bound, double poly[], double der[], int n, double eta) {

double y = lower\_bound;

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

exit(0);

}

file << "Tim cac diem cuc tri" << endl;

local.push\_back(lower\_bound);

while (y < upper\_bound) {

y = gradient\_descent(y, upper\_bound, poly, der, n, eta); // neu f(y) = 0 thi sao??? (rat kho xay ra)

if (y < upper\_bound && f(y, poly, n) != 0) {

local.push\_back(y);

file << "Diem cuc tri: (" << y << "; " << f(y, poly, n) << ")" << endl;

}

if (f(y, poly, n) == 0) {

weird.push\_back(y); // diem cuc tri se la nghiem

file << "Nghiem ky di: (" << y << "; " << f(y, poly, n) << ")" << endl;

}

y += step; // step du lon de vuot qua y (hien dang la diem cuc tri) muc dich de tim diem cuc tri khac

}

local.push\_back(upper\_bound);

file << "--------------------------------------" << endl;

file.close();

}

void root\_interval(double lower\_bound, double upper\_bound, double poly[], double der[], int n, double eta, int& count) {

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

exit(0);

}

all\_critical\_points(lower\_bound, upper\_bound, poly, der, n, eta);

file << "Cac khoang phan ly nghiem (truoc khi dung chia doi):" << endl;

for (size\_t i = 0; i + 1 < local.size(); i++) {

if (f(local[i], poly, n) \* f(local[i + 1], poly, n) < 0) {

intervals[count].push\_back(local[i] + step);

intervals[count].push\_back(local[i + 1] - step);

file << "(" << intervals[count][0] << "; " << intervals[count][1] << ")" << endl;

count++;

}

}

file << "--------------------------------------" << endl;

file.close();

}

void show\_intervals(int count) {

string str[20] = { "The First ", "The Second ", "The Third ", "The Forth ", "The Fifth ", "The Sixth " };

std::cout << endl;

for (int i = 0; i < count; i++) {

std::cout << str[i] << "interval: (" << intervals[i][0] << "; " << intervals[i][1] << ")";

std::cout << endl;

}

}

void bisection(double poly[], int n, int count) {

double c;

double a, b;

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

exit(0);

}

file << "Cac khoang phan ly nghiem sau khi dung chia doi: " << endl;

for (int i = 0; i < count; i++) {

file << "khoang so " << i + 1 << ": " << endl;

if (!intervals[i].empty()) {

a = intervals[i][0];

b = intervals[i][1];

do {

c = (a + b) / 2;

file << "a = " << setw(8) << a << " | b = " << b << " | c = " << c << " | f(a) \* f(c) = " << f(a, poly, n) \* f(c, poly, n) << endl;

if (f(c, poly, n) \* f(a, poly, n) < 0) {

b = c;

}

else if (f(c, poly, n) \* f(a, poly, n) > 0) {

a = c;

}

else {

weird.push\_back(c); // neu f(c) = 0 thi c se la nghiem boi

break;

}

} while (fabs(a - b) > 0.5);

if (f(c, poly, n) != 0) {

intervals[i][0] = a;

intervals[i][1] = b;

file << "(" << a << "; " << b << ")" << endl;

}

}

else {

continue;

}

}

file << "--------------------------------------" << endl;

file.close();

}

void regula(double& x0, double d, double poly[], double der[], int n) {

x0 = x0 - f(x0, poly, n) \* ((d - x0) / (f(d, poly, n) - f(x0, poly, n)));

}

void m1\_M1(double& m1, double& M1, double poly[], double der[], double der2[], int n, int number) {

/\* ta biet rang trong khoang ta xet thi f'(x), f''(x)

se khong doi dau nen min(|f'(x)|) la mot trong 2 dau mut

\*/

double a = intervals[number][0];

double b = intervals[number][1];

int sign = (f2(a, der2, n) > 0) ? 1 : (-1);

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

cout << "Failed to open this file!" << endl;

exit(0);

}

if (sign == 1 && f1(a, der, n) > 0) {

m1 = f1(a, der, n);

}

else if (sign == 1 && f1(a, der, n) < 0) {

m1 = (-1) \* f1(b, der, n);

}

else if (sign == -1 && f1(a, der, n) > 0) {

m1 = f1(b, der, n);

}

else {

m1 = (-1) \* f1(a, der, n);

}

if (m1 == fabs(f1(a, der, n))) {

M1 = fabs(f1(b, der, n));

}

else {

M1 = fabs(f1(a, der, n));

}

file << "Gia tri nho nhat va lon nhat cua f'(x) tren (" << a << "; " << b << ") la : m1 = " << m1 << " va M1 = " << M1 << endl;

file.close();

}

void fourier\_point(double& x0, double& d, double poly[], double der2[], int n, double a, double b) {

if (f(a, poly, n) \* f2(a, der2, n) < 0) {

x0 = a;

d = b;

}

else {

x0 = b;

d = a;

}

}

void iteration\_one(double m1, double M1, double poly[], double der[], double der2[], int n, int count) {

int max\_ite;

double x\_old = 0;

double x0, d;

double a, b;

double err;

int number = sub\_menu(count);

int dec;

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

std::cout << "Failed to open this file!" << endl;

exit(0);

}

file << "Ban dang dung chuc nang so 3" << endl;

a = intervals[number][0];

b = intervals[number][1];

m1\_M1(m1, M1, poly, der, der2, n, number);

fourier\_point(x0, d, poly, der2, n, a, b);

std::cout << "Number of iteration: ";

std::cin >> max\_ite;

file << "Diem fourier: x0 = " << x0 << " va d = " << d << endl;

file << "So lan lap: " << max\_ite << endl;

file << "Qua trinh lap: " << endl;

for (int i = 0; i < max\_ite; i++) {

regula(x0, d, poly, der, n);

if (i == max\_ite - 2) {

x\_old = x0;

}

file << setw(8) << "x" << i + 1 << " = " << x0 << endl;

}

std::cout << "Number of decimal displayed: ";

std::cin >> dec;

std::cout << "After " << max\_ite << " iteration, the root is: " << setprecision(dec) << x0 << setprecision(6) << endl;

std::cout << "Estimate error (formula 1): ";

std::cout << fabs(f(x0, poly, n)) / m1;

if (M1 <= 2 \* m1) {

err = fabs(x0 - x\_old);

std::cout << endl << "Estimate error (formula 2): " << err;

}

else {

err = ((M1 - m1) / m1) \* fabs(x0 - x\_old);

std::cout << endl << "Estimate error (formula 2): " << err;

}

file << "=> Nghiem tim duoc la: " << x0 << endl;

file << "Voi sai so theo cong thuc 1 la: " << fabs(f(x0, poly, n)) / m1 << endl;

file << "Voi sai so theo cong thuc 2 la: " << err << endl;

file << "--------------------------------------" << endl;

file.close();

}

void epsilon\_one(double m1, double M1, double poly[], double der[], double der2[], int n, int count) {

int dec;

int i = 0, j = 0;

double a, b, c;

double x0, d, temp, x\_old;

double epsilon;

int number = sub\_menu(count);

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

cout << "Failed to open this file!" << endl;

exit(0);

}

file << "Ban dang dung chuc nang so 4" << endl;

m1\_M1(m1, M1, poly, der, der2, n, number);

a = intervals[number][0];

b = intervals[number][1];

c = (M1 - m1) / m1;

std::cout << "Enter the epsilon: ";

std::cin >> epsilon;

std::cout << "Number of decimal displayed: ";

std::cin >> dec;

file << "Epsilon ma ban da nhap: " << epsilon << endl;

fourier\_point(x0, d, poly, der2, n, a, b);

file << "Diem fourier: x0 = " << x0 << " va d = " << d << endl;

file << "Qua trinh lap theo cong thuc tien nghiem:" << endl;

temp = x0; // giu gia tri ban dau cua x0

do {

file << setw(8) << "x" << i++ << " = " << x0 << endl;

regula(x0, d, poly, der, n);

} while (fabs(f(x0, poly, n)) / m1 >= epsilon);

std::cout << "Nghiem voi sai so " << epsilon << " la (theo cong thuc 1): " << setprecision(dec) << x0 << setprecision(6) << endl;

file << "=> Nghiem theo cong thuc sai so tien nghiem: " << x0 << "\n\n";

x0 = temp; // gan lai

file << "Qua trinh lap theo cong thuc hau nghiem: " << endl;

if (c <= 1) {

do {

x\_old = x0;

regula(x0, d, poly, der, n);

file << setw(8) << "x" << j++ << " = " << x\_old << " va x" << j << " = " << x0 << endl;

} while (fabs(x0 - x\_old) > epsilon);

}

else {

do {

x\_old = x0;

regula(x0, d, poly, der, n);

file << setw(8) << "x" << j++ << " = " << x\_old << " va x" << j << " = " << x0 << endl;

} while (c \* fabs(x0 - x\_old) > epsilon);

}

std::cout << "Nghiem voi sai so " << epsilon << " la (theo cong thuc 2): " << setprecision(dec) << x0 << setprecision(6) << endl;

file << "=> Nghiem theo cong thuc sai so hau nghiem: " << x0 << endl;

file << "--------------------------------------" << endl;

file.close();

}

void last\_one(double m1, double M1, double poly[], double der[], double der2[], int n, int count) {

int i = 0;

int dec;

double a, b;

double x0, d, x\_old;

double epsilon;

int number = sub\_menu(count);

fstream file;

file.open("test.txt", ios::app);

if (file.fail()) {

cout << "Failed to open this file!" << endl;

exit(0);

}

file << "Ban dang dung chuc nang so 5" << endl;

a = intervals[number][0];

b = intervals[number][1];

std::cout << "Enter the epsilon: ";

std::cin >> epsilon;

std::cout << "Number of decimal displayed: ";

std::cin >> dec;

file << "Ban chon khoang nghiem: (" << a << "; " << b << ")" << endl;

file << "Epsilon ma ban da nhap: " << epsilon << endl;

if (f(a, poly, n) \* f2(a, der2, n) < 0) {

x0 = a;

d = b;

}

else {

x0 = b;

d = a;

}

file << "Diem fourier: x0 = " << x0 << " va d = " << d << endl;

file << "Qua trinh lap: " << endl;

do {

x\_old = x0;

regula(x0, d, poly, der, n);

file << setw(8) << "x" << i++ << " = " << x\_old << " va x" << i << " = " << x0 << endl;

} while (fabs(x0 - x\_old) >= epsilon);

std::cout << "The root is: " << setprecision(dec) << x0 << setprecision(6);

file << "=> Nghiem thu duoc la: " << x0 << endl;

file << "--------------------------------------" << endl;

file.close();

}

1. **Ảnh source code** 
   1. <Header.h>

A picture containing text, screenshot, computer

Description automatically generated

Text

Description automatically generated

# A picture containing graphical user interface Description automatically generated

### <Main.cppText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedA picture containing text, screenshot, monitor, electronics Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedA computer screen capture Description automatically generated with low confidenceText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedText Description automatically generatedA picture containing text, screenshot, monitor, computer Description automatically generatedText Description automatically generatedText Description automatically generated

# Mục IV: Kiểm thử chương trình

1. **MODULE nhập vào đa thức**Graphical user interface, text

   Description automatically generated
2. Graphical user interface

   Description automatically generated with medium confidence**Giao diện người dùng**
3. **MODULE lựa chọn in ra miền nghiệm.**

Text

Description automatically generated

## **MODULE khoảng phân ly nghiệm của phương trình sau khi chia đôi.**

Text

Description automatically generated

## **MODULE tìm nghiệm với số lần lặp N**

Graphical user interface, text

Description automatically generated

Chọn khoảng nghiệm để thực hiện quá trình lặp

Text

Description automatically generated

Kết quả chạy với MODULE 3 sau 50 lần lặp với 12 chữ số lấy sau dấu phẩy

1. **MODULE tìm nghiệm với epsilon cho trước**

Text

Description automatically generated

1. **Lựa chọn tìm nghiệm thỏa mãn:** 

Text

Description automatically generated

1. **Hiển thị ra file**

Ví dụ 1: Da thuc f(x) = 8461x^5 + -973x^4 + 9341x^3 + 1253x^2 + -8753x^1 + -454

Bac cua da thuc: 5

Da thuc f(x) = 8461x^5 + -973x^4 + 9341x^3 + 1253x^2 + -8753x^1 + -454

Da thuc f'(x) = 42305x^4 + -3892x^3 + 28023x^2 + 2506x^1 + -8753

Da thuc f''(x) = 169220x^3 + -11676x^2 + 56046x^1 + 2506

--------------------------------------

+ Doi voi can tren:

He so am dau tien: k = 1

Max cua cac tri tuyet doi cua cac he so am la: B = 8753

Can tren cua mien chua nghiem: 2.03451

+ Doi voi can duoi:

He so am dau tien: k = 3

Max cua cac tri tuyet doi cua cac he so am la: B = 8753

Can duoi cua mien chua nghiem la: -2.01137

--------------------------------------

Tim cac diem cuc tri

Diem cuc tri: (-0.497358; 2743.08)

Diem cuc tri: (0.463646; -3175.61)

--------------------------------------

Cac khoang phan ly nghiem (truoc khi dung chia doi):

(-2.01037; -0.498358)

(-0.496358; 0.462646)

(0.464646; 2.03351)

--------------------------------------

Cac khoang phan ly nghiem sau khi dung chia doi:

khoang so 1:

a = -2.01037 | b = -0.498358 | c = -1.25437 | f(a) \* f(c) = 1.20291e+10

a = -1.25437 | b = -0.498358 | c = -0.876362 | f(a) \* f(c) = 1.05788e+08

(-0.876362; -0.498358)

khoang so 2:

a = -0.496358 | b = 0.462646 | c = -0.016856 | f(a) \* f(c) = -839780

(-0.496358; -0.016856)

khoang so 3:

a = 0.464646 | b = 2.03351 | c = 1.24908 | f(a) \* f(c) = -1.02029e+08

a = 0.464646 | b = 1.24908 | c = 0.856863 | f(a) \* f(c) = -7.06936e+06

(0.464646; 0.856863)

--------------------------------------

Ban dang dung chuc nang so 3

Cac nghiem ki di (nghiem boi) (neu co):

Gia tri nho nhat va lon nhat cua f'(x) tren (-0.876362; -0.498358) la : m1 = 49.1729 va M1 = 38145.6

Diem fourier: x0 = -0.498358 va d = -0.876362

So lan lap: 50

Qua trinh lap:

x1 = -0.677179

x2 = -0.749392

x3 = -0.769395

x4 = -0.774261

x5 = -0.775406

x6 = -0.775672

x7 = -0.775734

x8 = -0.775749

x9 = -0.775752

x10 = -0.775753

x11 = -0.775753

x12 = -0.775753

x13 = -0.775753

x14 = -0.775753

x15 = -0.775753

x16 = -0.775753

x17 = -0.775753

x18 = -0.775753

x19 = -0.775753

x20 = -0.775753

x21 = -0.775753

x22 = -0.775753

x23 = -0.775753

x24 = -0.775753

x25 = -0.775753

x26 = -0.775753

x27 = -0.775753

x28 = -0.775753

x29 = -0.775753

x30 = -0.775753

x31 = -0.775753

x32 = -0.775753

x33 = -0.775753

x34 = -0.775753

x35 = -0.775753

x36 = -0.775753

x37 = -0.775753

x38 = -0.775753

x39 = -0.775753

x40 = -0.775753

x41 = -0.775753

x42 = -0.775753

x43 = -0.775753

x44 = -0.775753

x45 = -0.775753

x46 = -0.775753

x47 = -0.775753

x48 = -0.775753

x49 = -0.775753

x50 = -0.775753

=> Nghiem tim duoc la: -0.775753

Voi sai so theo cong thuc 1 la: 2.77438e-14

Voi sai so theo cong thuc 2 la: 0

--------------------------------------

Ban dang dung chuc nang so 4

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Gia tri nho nhat va lon nhat cua f'(x) tren (-0.496358; -0.016856) la : m1 = 48.9797 va M1 = 8787.26

Epsilon ma ban da nhap: 5e-05

Diem fourier: x0 = -0.496358 va d = -0.016856

Qua trinh lap theo cong thuc tien nghiem:

x0 = -0.496358

x1 = -0.0649994

x2 = -0.0516351

=> Nghiem theo cong thuc sai so tien nghiem: -0.0516343

Qua trinh lap theo cong thuc hau nghiem:

x0 = -0.496358 va x1 = -0.0649994

x1 = -0.0649994 va x2 = -0.0516351

x2 = -0.0516351 va x3 = -0.0516343

x3 = -0.0516343 va x4 = -0.0516343

=> Nghiem theo cong thuc sai so hau nghiem: -0.0516343

--------------------------------------

Ban dang dung chuc nang so 5

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Ban chon khoang nghiem: (0.464646; 0.856863)

Epsilon ma ban da nhap: 3e-05

Diem fourier: x0 = 0.464646 va d = 0.856863

Qua trinh lap:

x0 = 0.464646 va x1 = 0.695223

x1 = 0.695223 va x2 = 0.763981

x2 = 0.763981 va x3 = 0.777339

x3 = 0.777339 va x4 = 0.779668

x4 = 0.779668 va x5 = 0.780066

x5 = 0.780066 va x6 = 0.780133

x6 = 0.780133 va x7 = 0.780145

=> Nghiem thu duoc la: 0.780145

--------------------------------------

Ví dụ 2: Da thuc f(x) = 214x^7 + 273x^6 + 87.213x^5 + 2452.45x^4 + 3154x^3 + 23.54x^2 + 125.364x^1 + 123.1

Bac cua da thuc: 7

Da thuc f(x) = 214x^7 + 273x^6 + 87.213x^5 + 2452.45x^4 + 3154x^3 + 23.54x^2 + 125.364x^1 + 123.1

Da thuc f'(x) = 1498x^6 + 1638x^5 + 436.065x^4 + 9809.8x^3 + 9462x^2 + 47.08x^1 + 125.364

Da thuc f''(x) = 8988x^5 + 8190x^4 + 1744.26x^3 + 29429.4x^2 + 18924x^1 + 47.08

--------------------------------------

+ Doi voi can tren:

He so am dau tien: k = 5

Max cua cac tri tuyet doi cua cac he so am la: B = 0

Can tren cua mien chua nghiem: 1

+ Doi voi can duoi:

He so am dau tien: k = -1

Max cua cac tri tuyet doi cua cac he so am la: B = 2452.45

Can duoi cua mien chua nghiem la: -12.46

--------------------------------------

Tim cac diem cuc tri

Diem cuc tri: (-1.85053; 1911.69)

Diem cuc tri: (-1.0033; -708.531)

--------------------------------------

Cac khoang phan ly nghiem (truoc khi dung chia doi):

(-12.459; -1.85153)

(-1.84953; -1.0043)

(-1.0023; 0.999)

--------------------------------------

Cac khoang phan ly nghiem sau khi dung chia doi:

khoang so 1:

a = -12.459 | b = -1.85153 | c = -7.15529 | f(a) \* f(c) = 1.47454e+18

a = -7.15529 | b = -1.85153 | c = -4.50341 | f(a) \* f(c) = 8.5959e+14

a = -4.50341 | b = -1.85153 | c = -3.17747 | f(a) \* f(c) = 1.55211e+12

a = -3.17747 | b = -1.85153 | c = -2.5145 | f(a) \* f(c) = 8.3297e+09

a = -2.5145 | b = -1.85153 | c = -2.18302 | f(a) \* f(c) = 6.95788e+07

(-2.18302; -1.85153)

khoang so 2:

a = -1.84953 | b = -1.0043 | c = -1.42692 | f(a) \* f(c) = 395153

(-1.42692; -1.0043)

khoang so 3:

a = -1.0023 | b = 0.999 | c = -0.00164956 | f(a) \* f(c) = -87073.2

a = -1.0023 | b = -0.00164956 | c = -0.501974 | f(a) \* f(c) = 125589

a = -0.501974 | b = -0.00164956 | c = -0.251812 | f(a) \* f(c) = -9304.43

(-0.501974; -0.251812)

--------------------------------------

Ban dang dung chuc nang so 3

Cac nghiem ki di (nghiem boi) (neu co):

Gia tri nho nhat va lon nhat cua f'(x) tren (-2.18302; -1.85153) la : m1 = 44.376 va M1 = 33882.1

Diem fourier: x0 = -1.85153 va d = -2.18302

So lan lap: 50

Qua trinh lap:

x1 = -1.99543

x2 = -2.05997

x3 = -2.0804

x4 = -2.08603

x5 = -2.08752

x6 = -2.08792

x7 = -2.08802

x8 = -2.08804

x9 = -2.08805

x10 = -2.08805

x11 = -2.08805

x12 = -2.08805

x13 = -2.08805

x14 = -2.08805

x15 = -2.08805

x16 = -2.08805

x17 = -2.08805

x18 = -2.08805

x19 = -2.08805

x20 = -2.08805

x21 = -2.08805

x22 = -2.08805

x23 = -2.08805

x24 = -2.08805

x25 = -2.08805

x26 = -2.08805

x27 = -2.08805

x28 = -2.08805

x29 = -2.08805

x30 = -2.08805

x31 = -2.08805

x32 = -2.08805

x33 = -2.08805

x34 = -2.08805

x35 = -2.08805

x36 = -2.08805

x37 = -2.08805

x38 = -2.08805

x39 = -2.08805

x40 = -2.08805

x41 = -2.08805

x42 = -2.08805

x43 = -2.08805

x44 = -2.08805

x45 = -2.08805

x46 = -2.08805

x47 = -2.08805

x48 = -2.08805

x49 = -2.08805

x50 = -2.08805

=> Nghiem tim duoc la: -2.08805

Voi sai so theo cong thuc 1 la: 3.20237e-15

Voi sai so theo cong thuc 2 la: 0

--------------------------------------

Ban dang dung chuc nang so 4

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Gia tri nho nhat va lon nhat cua f'(x) tren (-1.42692; -1.0043) la : m1 = 8.09554 va M1 = 4414.34

Epsilon ma ban da nhap: 0.0006

Diem fourier: x0 = -1.0043 va d = -1.42692

Qua trinh lap theo cong thuc tien nghiem:

x0 = -1.0043

x1 = -1.33147

x2 = -1.37464

x3 = -1.37737

x4 = -1.37753

=> Nghiem theo cong thuc sai so tien nghiem: -1.37754

Qua trinh lap theo cong thuc hau nghiem:

x0 = -1.0043 va x1 = -1.33147

x1 = -1.33147 va x2 = -1.37464

x2 = -1.37464 va x3 = -1.37737

x3 = -1.37737 va x4 = -1.37753

x4 = -1.37753 va x5 = -1.37754

x5 = -1.37754 va x6 = -1.37754

=> Nghiem theo cong thuc sai so hau nghiem: -1.37754

--------------------------------------

Ban dang dung chuc nang so 5

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Cac nghiem ki di (nghiem boi) (neu co):

Ban chon khoang nghiem: (-0.501974; -0.251812)

Epsilon ma ban da nhap: 0.00015

Diem fourier: x0 = -0.251812 va d = -0.501974

Qua trinh lap:

x0 = -0.251812 va x1 = -0.308969

x1 = -0.308969 va x2 = -0.32483

x2 = -0.32483 va x3 = -0.328728

x3 = -0.328728 va x4 = -0.329654

x4 = -0.329654 va x5 = -0.329872

x5 = -0.329872 va x6 = -0.329924

=> Nghiem thu duoc la: -0.329924

Ví dụ 3: Da thuc f(x) = 1x^5 + 7x^4 + 14x^3 + 2x^2 + -15x^1 + -9

Bac cua da thuc: 5

Da thuc f(x) = 1x^5 + 7x^4 + 14x^3 + 2x^2 + -15x^1 + -9

Da thuc f'(x) = 5x^4 + 28x^3 + 42x^2 + 4x^1 + -15

Da thuc f''(x) = 20x^3 + 84x^2 + 84x^1 + 4

--------------------------------------

+ Doi voi can tren:

He so am dau tien: k = 4

Max cua cac tri tuyet doi cua cac he so am la: B = 15

Can tren cua mien chua nghiem: 2.96799

+ Doi voi can duoi:

He so am dau tien: k = 1

Max cua cac tri tuyet doi cua cac he so am la: B = 15

Can duoi cua mien chua nghiem la: -16

--------------------------------------

Tim cac diem cuc tri

Nghiem ky di: (-3; 0)

Diem cuc tri: (-2.08062; -3.0407)

Nghiem ky di: (-1; 0)

Diem cuc tri: (0.480625; -13.7939)

--------------------------------------

Cac khoang phan ly nghiem (truoc khi dung chia doi):

(0.481625; 2.96699)

--------------------------------------

Cac khoang phan ly nghiem sau khi dung chia doi:

khoang so 1:

a = 0.481625 | b = 2.96699 | c = 1.72431 | f(a) \* f(c) = -1654.99

a = 0.481625 | b = 1.72431 | c = 1.10297 | f(a) \* f(c) = -105.74

a = 0.481625 | b = 1.10297 | c = 0.792295 | f(a) \* f(c) = 132.359

(0.792295; 1.10297)

--------------------------------------

Ban dang dung chuc nang so 3

Cac nghiem ki di (nghiem boi) (neu co): -3 -1

Gia tri nho nhat va lon nhat cua f'(x) tren (0.792295; 1.10297) la : m1 = 30.4299 va M1 = 85.4764

Diem fourier: x0 = 0.792295 va d = 1.10297

So lan lap: 50

Qua trinh lap:

x1 = 0.964997

x2 = 0.99494

x3 = 0.999287

x4 = 0.9999

x5 = 0.999986

x6 = 0.999998

x7 = 1

x8 = 1

x9 = 1

x10 = 1

x11 = 1

x12 = 1

x13 = 1

x14 = 1

x15 = 1

x16 = 1

x17 = 1

x18 = 1

x19 = 1

x20 = 1

x21 = 1

x22 = 1

x23 = 1

x24 = 1

x25 = 1

x26 = 1

x27 = 1

x28 = 1

x29 = 1

x30 = 1

x31 = 1

x32 = 1

x33 = 1

x34 = 1

x35 = 1

x36 = 1

x37 = 1

x38 = 1

x39 = 1

x40 = 1

x41 = 1

x42 = 1

x43 = 1

x44 = 1

x45 = 1

x46 = 1

x47 = 1

x48 = 1

x49 = 1

x50 = 1

=> Nghiem tim duoc la: 1

Voi sai so theo cong thuc 1 la: 0

Voi sai so theo cong thuc 2 la: 0

--------------------------------------

Ban dang dung chuc nang so 4

Cac nghiem ki di (nghiem boi) (neu co): -3 -1

Gia tri nho nhat va lon nhat cua f'(x) tren (0.792295; 1.10297) la : m1 = 30.4299 va M1 = 85.4764

Epsilon ma ban da nhap: 0.0003

Diem fourier: x0 = 0.792295 va d = 1.10297

Qua trinh lap theo cong thuc tien nghiem:

x0 = 0.792295

x1 = 0.964997

x2 = 0.99494

x3 = 0.999287

=> Nghiem theo cong thuc sai so tien nghiem: 0.9999

Qua trinh lap theo cong thuc hau nghiem:

x0 = 0.792295 va x1 = 0.964997

x1 = 0.964997 va x2 = 0.99494

x2 = 0.99494 va x3 = 0.999287

x3 = 0.999287 va x4 = 0.9999

x4 = 0.9999 va x5 = 0.999986

=> Nghiem theo cong thuc sai so hau nghiem: 0.999986

--------------------------------------

Ban dang dung chuc nang so 5

Cac nghiem ki di (nghiem boi) (neu co): -3 -1

Ban chon khoang nghiem: (0.792295; 1.10297)

Epsilon ma ban da nhap: 3e-05

Diem fourier: x0 = 0.792295 va d = 1.10297

Qua trinh lap:

x0 = 0.792295 va x1 = 0.964997

x1 = 0.964997 va x2 = 0.99494

x2 = 0.99494 va x3 = 0.999287

x3 = 0.999287 va x4 = 0.9999

x4 = 0.9999 va x5 = 0.999986

x5 = 0.999986 va x6 = 0.999998

=> Nghiem thu duoc la: 0.999998