Trường Đại Học Khoa Học Tự Nhiên, ĐHQG-HCM

**KHOA ĐIỆN TỬ VIỄN THÔNG**

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

****

**BÁO CÁO ĐỒ ÁN PHƯƠNG PHÁP TÍNH VÀ MATLAB**

Họ và Tên sinh viên: Nguyễn Đoàn Nguyên Khang

MSSV: 21200106

Giảng viên hướng dẫn: TS Trần Xuân Vinh

Thành phố Hồ Chí Minh, 12/2023

**MỤC LỤC**

[**I.** **Đề tài:** 1](#_Toc170590938)

[**II.** **Phần Code thuật toán:** 1](#_Toc170590939)

[**1.** **Hàm tính gần đúng nghiệm phương trình bằng phương pháp lặp và tiếp tuyến:** 1](#_Toc170590940)

[**a.** **Phương pháp lặp:** 1](#_Toc170590941)

[**b.** **Phươnng pháp Newton (tiếp tuyến):** 2](#_Toc170590942)

[**2.** **Hàm tính gần đúng nghiệm phương trình bằng phương pháp Cholesky:** 3](#_Toc170590943)

[**3.** **Hàm tính gần đúng nghiệm phương trình bằng đa thức nội suy Newton:** 4](#_Toc170590944)

[**4.** **Hàm tính gần đúng nghiệm phương trình bằng đa thức nội suy Larange:** 4](#_Toc170590945)

[**III.** **Phần Code giao diện:** 5](#_Toc170590946)

[**1.** **Lưu đồ giải thuật** 5](#_Toc170590947)

[**2.** **Button pushed function: TinhvaVeButton** 7](#_Toc170590948)

[**IV.** **Giao diện và chạy mô phỏng:** 9](#_Toc170590949)

[**1.** **Giao diện:** 9](#_Toc170590950)

[**2.** **Chạy mô phỏng:** 10](#_Toc170590951)

[**V.** **Tổng kết:** 13](#_Toc170590952)

1. **Đề tài:**

**A screenshot of a computer

Description automatically generated**

1. **Phần Code thuật toán:**
2. **Hàm tính gần đúng nghiệm phương trình bằng phương pháp lặp và tiếp tuyến:**

A white paper with black text

Description automatically generated

1. **Phương pháp lặp:**

function [nghiem] = lap(f,flap,a,b,solanlap)

syms x;

fd1 = str2func(['@(x)' char(diff(f(x)))]);

fd2 = str2func(['@(x)' char(diff(fd1(x)))]);

nostop = 1;

temp1 = double(solve(diff(f(x))));

temp2 = double(solve(diff(fd1(x))));

%Kiem tra f'(x) co doi dau trong khoang phan li nghiem hay khong

if ~isempty(temp1)

for i=1:length(temp1)

if (temp1(i)<=b) && (temp1(i)>=a)

disp('Ham f'' doi dau trong khoang phan li nghiem');

nostop=0;

nghiem = 'khong xac dinh';

% solanlap = 'khong xac dinh';

break;

end

end

end

%Kiem tra f"(x) co doi dau trong khoang phan li nghiem hay ko

if ~isempty(temp2)

for i=1:length(temp2)

if (temp2(i)<=b) && (temp2(i)>=a)

disp('Ham f" doi dau trong khoang phan li nghiem');

nostop=0;

nghiem = 'khong xac dinh';

% solanlap = 'khong xac dinh';

break;

end

end

end

if (nostop)

x1=b;

x0=a;

for i=1:1:solanlap

x1=flap(x0);

x0=x1;

nghiem = x1;

end

end % ket thuc function

1. **Phươnng pháp Newton (tiếp tuyến):**

function [nghiem] = tieptuyen(f,a,b,solanlap)

syms x;

fd1 = str2func(['@(x)' char(diff(f(x)))]);

fd2 = str2func(['@(x)' char(diff(fd1(x)))]);

nostop = 1;

temp1 = double(solve(diff(f(x))));

temp2 = double(solve(diff(fd1(x))));

%Kiem tra f'(x) co doi dau trong khoang phan li nghiem hay khong

if ~isempty(temp1)

for i=1:length(temp1)

if (temp1(i)<=b) && (temp1(i)>=a)

disp('Ham f'' doi dau trong khoang phan li nghiem');

nostop=0;

nghiem = 'khong xac dinh';

% solanlap = 'khong xac dinh';

break;

end

end

end

%Kiem tra f"(x) co doi dau trong khoang phan li nghiem hay ko

if ~isempty(temp2)

for i=1:length(temp2)

if (temp2(i)<=b) && (temp2(i)>=a)

disp('Ham f" doi dau trong khoang phan li nghiem');

nostop=0;

nghiem = 'khong xac dinh';

% solanlap = 'khong xac dinh';

break;

end

end

end

if (nostop)

x0=b;

while f(x0)\*fd2(x0)<=0

x0 = (x0+a)/2;

end

if f(x0)\*fd2(x0)>0

for i=1:1:solanlap

x1=x0-f(x0)/fd1(x0);

x0=x1;

nghiem=x1

end

end

end % ket thuc function

1. **Hàm tính gần đúng nghiệm phương trình bằng phương pháp Cholesky:**

**A white paper with black text

Description automatically generated**

function [x] = cholesky(A,b)

n=size(A,1);

L=zeros(n,n);

for i=1:n

L(i, i) = sqrt(A(i, i) - L(i, :)\*L(i, :)');

for j=(i + 1):n

L(j, i) = (A(j, i) - L(i,:)\*L(j ,:)')/L(i, i);

end

end

Lt=L';

y=L\b;

x=Lt\y;

1. **Hàm tính gần đúng nghiệm phương trình bằng đa thức nội suy Newton:**

A close-up of numbers

Description automatically generated

function [result] = noisuy\_newton(xa, ya, x)

n = length(xa);

da = ya;

for i = 1: n

for j = 1: (i-1)

da(i)=(da(j) - da(i))/(xa(j) - xa(i));

end

end

m = length(da);

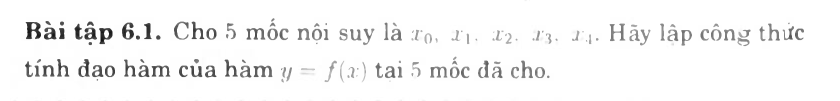
result = da(m);

for k = (m-1):-1:1

result = result \* (x - xa(k)) + da(k);

end

1. **Hàm tính gần đúng nghiệm phương trình bằng đa thức nội suy Larange:**



function [result] = Lagrange(xa, ya, x, h)

x1=x;

syms x;

n = length(xa);

sum = 0;

for i=1:n

product = ya(i);

for j = 1:n

if i ~= j

product = product\*(x - xa(j))/(xa(i) - xa(j));

end

end

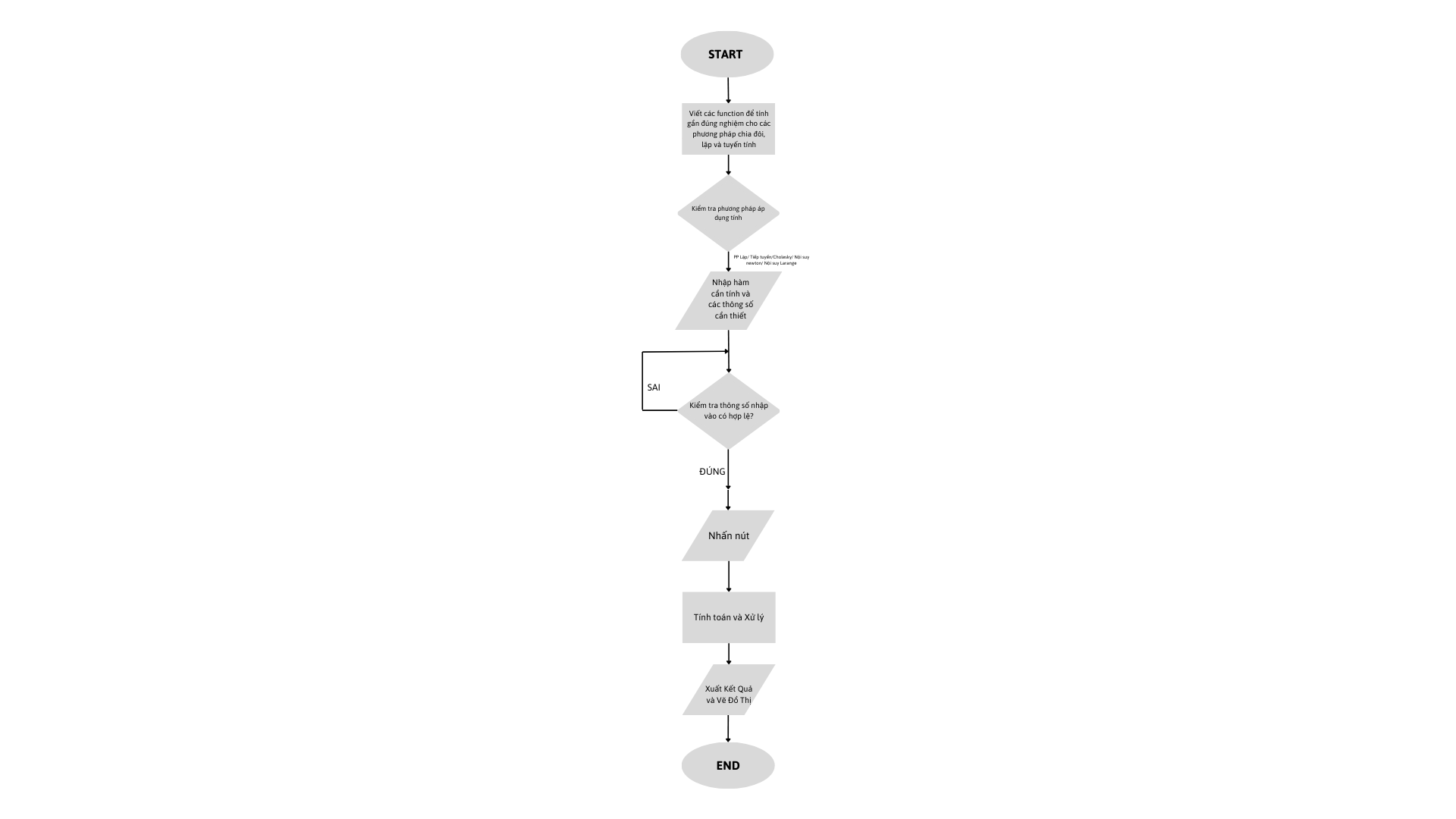
sum = sum + product;

end

rs = diff(sum);

result = double(subs(rs, x, x1));

1. **Phần Code giao diện:**
2. **Lưu đồ giải thuật**

****

1. **Button pushed function: TinhvaVeButton**

% Button pushed function: RUNButton

function RUNButtonPushed(app, event)

%\*\*\*\*\*\*\*\*\*\* Chuong 8 \*\*\*\*\*\*\*\*\*\*

if (app.HmhocdliucnxlEditField.Visible==1 && app.GiihnabsolanlapEditField.Visible==1)

try

sym x;

f = str2func(['@(x)' char(app.HmhocdliucnxlEditField.Value)]);

flap = str2func(['@(x)' char(app.HmbinixtheofxkhisdngPPlpEditField.Value)]);

%gioihan = app.GiihnabsolanlapEditField.Value;

gh = str2num(app.GiihnabsolanlapEditField.Value);

if isempty(gh) || numel(gh) ~= 3

%Kiểm tra số phần tử có trong 'Giới hạn [a b solanlap]'

errordlg('Phải nhập đủ 3 thành phần [a, b, solanlap]. Vui lòng kiểm tra và nhập lại.', 'Lỗi');

else

a = gh(1,1);

b = gh(1,2);

solanlap = gh(1,3);

x\_8 = linspace(a,b,40);

for i=1:40

y\_8(i) = f(x\_8(i));

end

y\_8;

if (strcmp(app.PPxlDropDown.Value, 'Lặp'))

[nghiem] = lap(f,flap,a,b,solanlap);

app.KtqutnhtonTextArea.Value=num2str(nghiem);

plot(app.UIAxes, x\_8, y\_8, '-\*r');

elseif (strcmp(app.PPxlDropDown.Value, 'Tiếp tuyến'))

[nghiem] = tieptuyen(f,a,b,solanlap);

app.KtqutnhtonTextArea.Value=num2str(nghiem);

plot(app.UIAxes, x\_8, y\_8, '-og');

end

end

catch

% Nếu có lỗi khi nhập liệu, hiển thị hộp thoại thông báo lỗi

errordlg('Dữ liệu không hợp lệ. Vui lòng kiểm tra và nhập lại.', 'Lỗi');

end

%\*\*\*\*\*\*\*\*\*\* Chuong 11+12 \*\*\*\*\*\*\*\*\*\*

elseif (app.NhphEditField.Visible == 1)

try

xa\_1112 = str2num(app.NhpMngAxaEditField.Value);

ya\_1112 = str2num(app.NhpMngByaEditField.Value);

gtns\_x\_1112 = str2num(app.NhpgitrnisuyxEditField.Value);

gt\_h\_1112 = str2num(app.NhphEditField.Value);

if ~isreal(gtns\_x\_1112) || ~isreal(gt\_h\_1112) || isempty(gtns\_x\_1112) || isempty(gt\_h\_1112) || numel(gtns\_x\_1112) ~= 1 || numel(gt\_h\_1112) ~= 1

%Kiểm tra giá trị của x và h có phải là số thực

errordlg('Giá trị của x or h phải là số thực duy nhất. Vui lòng kiểm tra và nhập lại.', 'Lỗi');

else

[result\_1112] = noisuy(xa\_1112, ya\_1112, gtns\_x\_1112, gt\_h\_1112);

app.KtqutnhtonTextArea.Value = num2str(result\_1112);

end

catch

% Nếu có lỗi khi nhập liệu, hiển thị hộp thoại thông báo lỗi

errordlg('Dữ liệu không hợp lệ. Vui lòng kiểm tra và nhập lại.', 'Lỗi');

end

%\*\*\*\*\*\*\*\*\*\* Chuong 10 \*\*\*\*\*\*\*\*\*\*

elseif (app.NhpgitrnisuyxEditField.Visible==1 )

try

xa\_10 = str2num(app.NhpMngAxaEditField.Value);

ya\_10 = str2num(app.NhpMngByaEditField.Value);

gtns\_x\_10 = str2num(app.NhpgitrnisuyxEditField.Value);

if isempty(gtns\_x\_10) || ~isreal(gtns\_x\_10) || numel(gtns\_x\_10) ~= 1

errordlg('Giá trị của x phải là một số thực duy nhất. Vui lòng kiểm tra và nhập lại.', 'Lỗi');

else

[result] = noisuy\_newton(xa\_10, ya\_10, gtns\_x\_10);

app.KtqutnhtonTextArea.Value = num2str(result);

end

catch

% Nếu có lỗi khi nhập liệu, hiển thị hộp thoại thông báo lỗi

errordlg('Dữ liệu không hợp lệ. Vui lòng kiểm tra và nhập lại.', 'Lỗi');

end

%\*\*\*\*\*\*\*\*\*\* Chuong 9 \*\*\*\*\*\*\*\*\*\*

elseif (app.NhpMngAxaEditField.Visible == 1 && app.NhpMngByaEditField.Visible == 1)

try

A = str2num(app.NhpMngAxaEditField.Value);

b = str2num(app.NhpMngByaEditField.Value);

[x] = cholesky(A, b);

% Tạo chuỗi định dạng "[a1; a2; a3; ...]"

xString = sprintf('[%s ]', strjoin(arrayfun(@num2str, x, 'UniformOutput', false), '; '));

app.KtqutnhtonTextArea.Value = xString;

catch

% Nếu có lỗi khi nhập liệu, hiển thị hộp thoại thông báo lỗi

errordlg('Dữ liệu không hợp lệ. Vui lòng kiểm tra và nhập lại.', 'Lỗi');

end

end

Phần code này sẽ thực hiện các thao tác đọc dữ liệu vào, bắt đầu tính toán và xuất kết quả nghiệm tính được và số lần lặp ra, đồng thời vẽ đồ thị của hàm f(x) trong khoảng phân li nghiệm.

1. **Giao diện và chạy mô phỏng:**
2. **Giao diện:**

Sử dụng các **Edit Field (Text)** để nhập hàm f(x); nhập hàm x biến đổi theo f(x) phục vụ cho quá trình sử dụng phương pháp lặp; nhập khoảng phân li nghiệm a và b; nhập sai số.

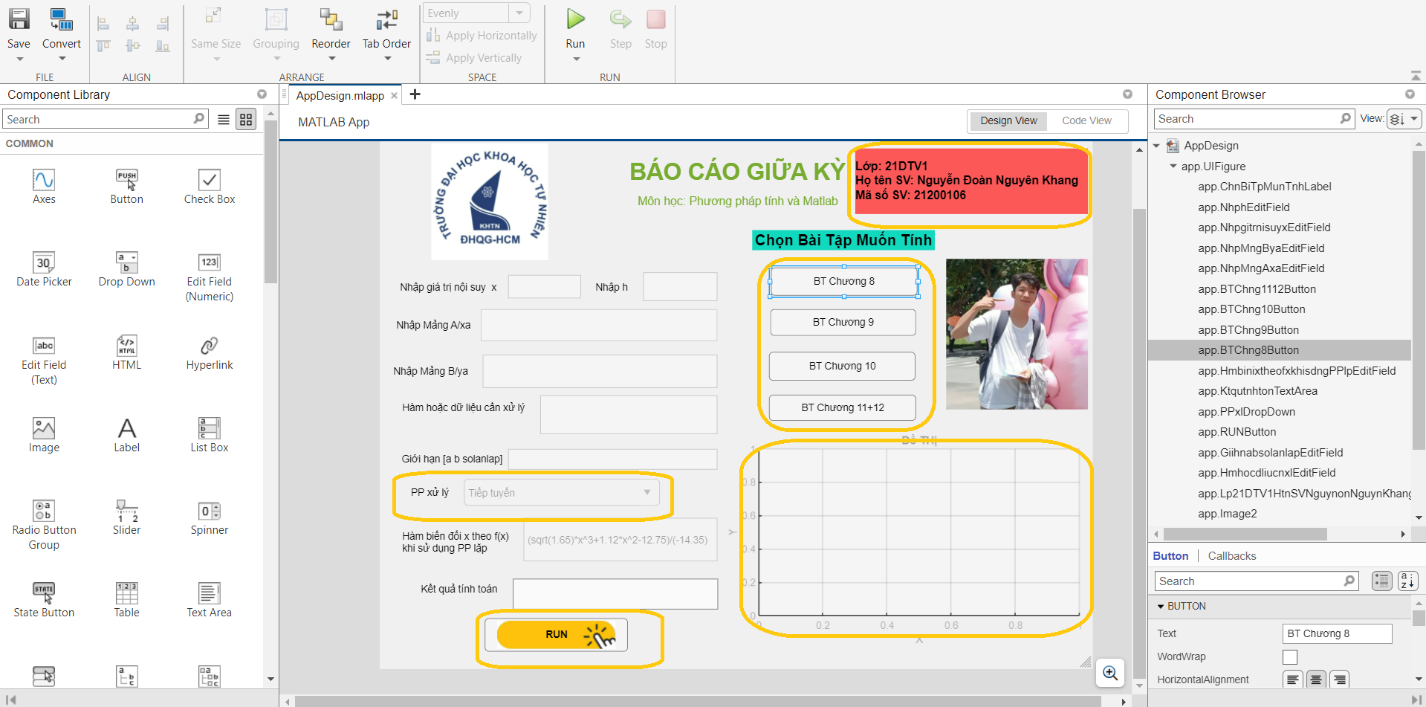
Sử dụng **Radio Button Group** để chọn 1 trong 4 bài tập của từng chương lần lượt tương ứng với các phương pháp (lặp/tiếp tuyến; Cholesky; nội suy newton; nội suy Larange) để tính.

**Drop Down** để chọn phương pháp ở bài tập chương 8.

Một nút **Button** để tính kết quả nghiệm và số lần lặp tối đa.

Sử dụng các **Text Area** để hiển thị kết quả thu được là nghiệm và sai số tối đa sau khi nhấn nút nhấn.

Sử dụng **Axes** để vẽ đồ thị f(x) trong khoảng phân li nghiệm sau khi nhấn nút.



1. **Chạy mô phỏng:**

* Giao diện khi khởi động:

A screenshot of a computer

Description automatically generated

* Trường hợp không nhập hoặc nhập sai dữ liệu:



* Thử mô phỏng bài tập chương 8:

A screenshot of a computer

Description automatically generated

* Thử mô phỏng bài tập chương 10:



1. **Tổng kết:**

Bằng việc vận dụng các kiến thức đã học được từ các buổi lý thuyết cũng như thực hành trên trường và tìm hiểu trên internet. Đề tài thiết kế giao diện triển khai các phương pháp tính gần đúng để hoàn thành các bài tập lý thuyết ở BTVN1 đã được hoàn thành.

Giao diện hoạt động đúng theo các yêu cầu được giao của đề tài, tính toán và vẽ đồ thị một các chính xác, kiểm soát tốt một số lỗi nhập liệu từ người dùng và cho ra kết quả chính xác.

**🙣 End 🙡**