



**BÁO CÁO ĐỒ ÁN MÔN HỌC
MÔN THỰC HÀNH PHƯƠNG PHÁP TÍNH**

HK1 - NĂM HỌC: 2025-2026

TÊN ĐỒ ÁN:

**ỨNG DỤNG MATLAB APP DESIGNER TRONG VIỆC MÔ PHỎNG
VÀ GIẢI QUYẾT CÁC BÀI TOÁN PHƯƠNG PHÁP TÍNH**

LỚP: 23DTV_CLC3_NHÓM FIRE

GIẢNG VIÊN HƯỚNG DẪN: THS. HUỲNH QUỐC THỊNH

Họ và tên các thành viên:

| STT | Họ và tên | MSSV |
|-----|---------------------------------|----------|
| 1 | Mai Lý Khải Triều (Trưởng nhóm) | 23207136 |
| 2 | Đỗ Trường Tín | 23207135 |
| 3 | Bùi Anh Phúc | 23207094 |
| 4 | Lý Anh Thư | 23207114 |
| 5 | Phan Phước Sơn | 23207101 |
| 6 | Hà Vĩ Khang | 23207131 |

LỜI CẢM ƠN

Trong quá trình thực hiện đồ án “Ứng dụng Matlab App Designer trong việc mô phỏng và giải quyết các bài toán Phương pháp tính”, nhóm em đã nhận được sự hỗ trợ và hướng dẫn rất tận tình. Nhóm em xin gửi lời cảm ơn chân thành đến thầy ThS. Huỳnh Quốc Thịnh. Thầy đã trực tiếp hướng dẫn, dành thời gian giải đáp những thắc mắc và chia sẻ những kiến thức chuyên môn và kinh nghiệm quý báu để nhóm có thể hoàn thành tốt đê tài này.

Nhóm em mong nhận được những ý kiến đóng góp của thầy để đê tài được hoàn thiện hơn.

TP. Hồ Chí Minh, ngày 01 tháng 12 năm 2025

TÓM TẮT

Đồ án tập trung nghiên cứu và xây dựng ứng dụng trên nền tảng Matlab App Designer nhằm giải quyết các bài toán trọng tâm của môn Phương pháp tính. Mục tiêu của đề tài là số hóa các thuật toán phức tạp thành một công cụ có giao diện trực quan, giúp người dùng dễ dàng tương tác và tính toán chính xác các bài toán giải tích số.

Về mặt nội dung, ứng dụng được thiết kế theo cấu trúc đa Tab, mỗi phân mục tương ứng với một nhóm bài toán cụ thể bao gồm:

- Tìm nghiệm phương trình: Triển khai các phương pháp Chia đôi, Lặp và Newton, kết hợp tính năng vẽ đồ thị hàm số để quan sát khoảng phân ly nghiệm.
- Nội suy: Xây dựng thuật toán nội suy Newton và Lagrange, hỗ trợ tìm đa thức nội suy và dự đoán giá trị tại điểm bất kỳ.
- Hồi quy: Thực hiện các mô hình hồi quy Tuyến tính, Hàm mũ và Logarit, cho phép biểu diễn tương quan giữa dữ liệu thực và đường hồi quy trên đồ thị.
- Đạo hàm: Áp dụng các công thức xấp xỉ tiến, lùi và trung tâm để tính toán từ tập dữ liệu rời rạc hoặc từ hàm số cho trước.
- Tích phân số: Sử dụng các phương pháp Hình thang, Simpson 1/3 và Simpson 3/8 để tính toán diện tích dưới đường cong với độ chính xác cao.

Hệ thống luận lý của ứng dụng được xây dựng dựa trên việc viết các hàm riêng biệt cho từng phương pháp và tổng hợp lại trong giao diện điều khiển chung. Kết quả thực hiện cho thấy ứng dụng đáp ứng đầy đủ các tiêu chí về tính toán, minh họa thuật toán và tối ưu hóa trải nghiệm người dùng trong môi trường Matlab.

MỤC LỤC

| | | |
|--------|--|----|
| I. | THUẬT TOÁN | 7 |
| 1.1. | THUẬT TOÁN TÌM NGHIỆM PHƯƠNG TRÌNH..... | 7 |
| 1.2. | THUẬT TOÁN NỘI SUY | 8 |
| 1.3. | THUẬT TOÁN HÒI QUY..... | 10 |
| 1.4. | THUẬT TOÁN ĐẠO HÀM | 11 |
| 1.5. | THUẬT TOÁN TÍCH PHÂN..... | 13 |
| II. | HOẠT ĐỘNG CỦA ÚNG DỤNG | 15 |
| 2.1. | GIAO DIỆN ÚNG DỤNG..... | 15 |
| 2.2. | HƯỚNG DẪN SỬ DỤNG | 19 |
| III. | KẾT QUẢ THỰC NGHIỆM | 23 |
| 3.1. | CHƯƠNG TRÌNH TÌM NGHIỆM PHƯƠNG TRÌNH..... | 23 |
| 3.2. | CHƯƠNG TRÌNH NỘI SUY | 24 |
| 3.3. | CHƯƠNG TRÌNH HÒI QUY..... | 25 |
| 3.4. | CHƯƠNG TRÌNH ĐẠO HÀM..... | 26 |
| 3.5. | CHƯƠNG TRÌNH TÍCH PHÂN | 27 |
| IV. | ĐÁNH GIÁ, NHẬN XÉT | 28 |
| 4.1. | ĐÁNH GIÁ..... | 28 |
| 4.1.1. | ƯU ĐIỂM | 28 |
| 4.1.2. | HẠN CHẾ..... | 28 |
| 4.2. | NHẬN XÉT | 28 |
| V. | HƯỚNG PHÁT TRIỂN..... | 29 |
| 5.1. | MỞ RỘNG THƯ VIỆN BÀI TOÁN TOÁN HỌC | 29 |
| 5.2. | NÂNG CẤP TRẢI NGHIỆM NGƯỜI DÙNG (UI/UX)..... | 29 |
| 5.3. | TỐI ƯU HÓA THUẬT TOÁN VÀ HIỆU SUẤT | 29 |
| 5.4. | ĐÓNG GÓI VÀ PHÂN PHỐI ÚNG DỤNG | 29 |
| VI. | ĐÁNH GIÁ NHÓM | 30 |
| 6.1. | PHÂN CÔNG CÔNG VIỆC | 30 |
| 6.2. | TIÊU CHÍ ĐÁNH GIÁ..... | 30 |
| 6.3. | KẾT QUẢ ĐÁNH GIÁ..... | 31 |
| VII. | PHỤ LỤC | 31 |

DANH MỤC HÌNH

| | |
|--|----|
| Hình 1. Giao diện giới thiệu đồ án..... | 15 |
| Hình 2. Giao diện giới thiệu thành viên..... | 15 |
| Hình 3. Giao diện giới thiệu nhiệm vụ..... | 15 |
| Hình 4. Giao diện chương trình tìm nghiệm bằng phương pháp lặp..... | 16 |
| Hình 5. Giao diện chương trình tìm nghiệm bằng phương pháp chia đôi..... | 16 |
| Hình 6. Giao diện chương trình tìm nghiệm bằng phương pháp Newton..... | 16 |
| Hình 7. Giao diện chương trình nội suy..... | 17 |
| Hình 8. Giao diện chương trình hồi quy..... | 17 |
| Hình 9. Giao diện chương trình đạo hàm..... | 18 |
| Hình 10. Giao diện chương trình hồi quy..... | 18 |
| Hình 11. Kết quả thực nghiệm chương trình tính nghiệm bằng phương pháp chia đôi... | 23 |
| Hình 12. Kết quả thực nghiệm chương trình tính nghiệm bằng phương pháp lặp đơn.... | 23 |
| Hình 13. Kết quả thực nghiệm chương trình tính nghiệm bằng phương pháp Newton.... | 23 |
| Hình 14. Kết quả thực nghiệm chương trình nội suy bằng Lagrange..... | 24 |
| Hình 15. Kết quả thực nghiệm chương trình nội suy bằng Newton..... | 24 |
| Hình 16. Kết quả thực nghiệm chương trình hồi quy tuyến tính..... | 25 |
| Hình 17. Kết quả thực nghiệm chương trình hồi quy (Hàm mũ)..... | 25 |
| Hình 18. Kết quả thực nghiệm chương trình hồi quy (Hàm Logarit)..... | 25 |
| Hình 19. Kết quả thực nghiệm chương trình đạo hàm (Xấp xỉ trung tâm)..... | 26 |
| Hình 20. Kết quả thực nghiệm chương trình đạo hàm (Xấp xỉ tiến)..... | 26 |
| Hình 21. Kết quả thực nghiệm chương trình đạo hàm (Xấp xỉ lùi)..... | 26 |
| Hình 22. Kết quả thực nghiệm chương trình tích phân (Hình thang)..... | 27 |

DANH MỤC BẢNG

| | |
|--|----|
| Bảng 1. Phân công nhiệm vụ..... | 30 |
| Bảng 2. Tiêu chí đánh giá..... | 30 |
| Bảng 3. Kết quả đánh giá nhóm..... | 31 |
| Bảng 4. Kết quả đánh giá thành viên..... | 31 |

I. THUẬT TOÁN

1.1. THUẬT TOÁN TÌM NGHIỆM PHƯƠNG TRÌNH

Chương trình

```
% Chia đôi
function [result, n] = chiadoi(f, a, b, eps)
    n = 0;
    while abs(b - a) > eps
        c = (a + b)/2;
        if f(a)*f(c) < 0
            b = c;
        else
            a = c;
        end
        n = n + 1;
    end
    result = (a + b)/2;
end
```

Chương trình

```
% Lặp
function [result, n] = lap(g, x0, eps)
    n = 0;
    while true
        x1 = g(x0);
        n = n + 1;
        if abs(x1 - x0) < eps
            result = x1;
            return;
        end
        x0 = x1;
    end
end
```

Chương trình

```
% Newton
function [result, n] = newton(f, df, x0, eps)
    n = 0;
    while true
        x1 = x0 - f(x0)/df(x0);
        n = n + 1;
        if abs(x1 - x0) < eps
            result = x1;
            return;
        end
        x0 = x1;
    end
```

1.2. THUẬT TOÁN NỘI SUY

Chương trình

```
function p = LagrangeSym(xa, ya)
    syms x
    n = numel(xa);
    p = sym(0);
    for i = 1:n
        Li = sym(1);
        for j = 1:n
            if j ~= i
                Li = Li * (x - xa(j)) / (xa(i) - xa(j));
            end
        end
        p = p + ya(i) * Li;
    end
    p = expand(p);
end
```

Chương trình

```
function y = Lagrange(xa, ya, x)
    n = length(xa)
    y = 0
    for i = 1:n
        Li = 1;
        for j = 1:n
            if j ~= i
                Li = Li*(x - xa(j))/(xa(i) - xa(j));
            end
        end
        y = y + ya(i) * Li;
    end
end
```

Chương trình

```
function p = NewtonSym(xa, ya)
    syms x % khai báo biến x dạng symbol
    d = DividedDifference(xa, ya); % tính mảng hệ số sai phân
    n = numel(xa); % đếm số lượng điểm dữ liệu đầu vào
    p = sym(0); % khởi tạo đa thức với kết quả ban đầu là 0
    term = sym(1); % khởi tạo biến tích lũy ( $x-x_0$ ) ban đầu là 1
    for i = 1:n % Vòng lặp duyệt từng mốc nội suy
        if i > 1 % nếu không phải vòng lặp 1
            term = term * (x - xa(i-1)); % nhân thêm thừa số ( $x - x_{trước}$ ) vào biến tích lũy
        end
    end
```

```

        p = p + d(i) * term; % cộng dòn hאג tử mới ( hệ số * tích luỹ)
và đa thức p
end
p = expand(p); % khai triển và rút gọn biểu thức đa thức cuối cùng
end

```

Chương trình

```

function result = NewtonInterpolation(xa, ya, x)
    d = DividedDifference(xa, ya);
    result = NewtonForm(xa, d, x);
end

function d = DividedDifference(xa, ya)
    xa = xa(:); ya = ya(:);
    n = numel(xa);
    d = ya;
    for j = 2:n
        for i = n:-1:j
            d(i) = (d(i) - d(i-1)) / (xa(i) - xa(i-j+1));
        end
    end
end

```

Chương trình

```

function y = NewtonForm(xa, d, x)
    n = numel(d);
    y = d(n);
    for i = n-1:-1:1
        y = y .* (x - xa(i)) + d(i);
    end
end

```

Chương trình

```

function result = NewtonForm(xa, da, x)
    n = numel(da);
    result = da(n);
    for i = n-1:-1:1
        result = result .* (x - xa(i)) + da(i);
    end
end

```

1.3. THUẬT TOÁN HỒI QUY

Chương trình

```
% Button pushed function: PtrinhHoiQuyButton
function PtrinhHoiQuyButtonPushed(app, event)
% str2num đổi chuỗi thành mảng số
% lấy chuỗi nhập là hàng đầu tiên
x = str2num(app.GiatriXTextArea.Value{1});
y = str2num(app.GiatriYTextArea.Value{1});
% kiểm tra nếu x,y rỗng
if isempty(x) || isempty(y)
    uialert(app.UIFigure, 'Chưa nhập dữ liệu x và y', 'Lỗi');
    return;
end
% chọn phương pháp từ DropDown và lưu vào biến model.
method = app.ChonPPhapDropDown.Value;
model.method = method;
switch method
    case 'Tuyến tính'
        p = polyfit(x, y, 1); % polyfit bậc 1 tìm hệ số a,b
        model.a = p(1);
        model.b = p(2);
        yfit = polyval(p, x); % tính giá trị y dự đoán
    case 'Hàm mũ'
        % lấy log 2 vẽ, dùng polyfit để tìm
        P = polyfit(x, log(y), 1);
        model.b = P(1);
        model.a = exp(P(2)); % khôi phục a = exp(P(2))
        yfit = model.a * exp(model.b * x);
    case 'Logarit'
        P = polyfit(log(x), y, 1);
        model.b = P(1);
        model.a = P(2);
        yfit = model.a + model.b * log(x);
    end
% Lưu toàn bộ hệ số hồi quy để khi nhấn nút "Kết quả dự đoán"
% có thể dùng lại.
app.UIFigure.UserData = model;
% vẽ đồ thị và điểm chính
title(app.UIAxes, 'Đồ thị hàm hồi quy và dữ liệu thực');
plot(app.UIAxes, x, y, 'o', 'LineWidth', 2);
hold(app.UIAxes, 'on');
plot(app.UIAxes, x, yfit, 'LineWidth', 2);
hold(app.UIAxes, 'off');
legend(app.UIAxes, 'Điểm dữ liệu', 'Hồi quy');
end
% Button pushed function: KquaDuDoanButton
function KquaDuDoanButtonPushed(app, event)
% lấy giá trị x cần dự đoán
xp = app.GiatriDudoanEditField.Value;
% lấy mô hình đã lưu, nếu chưa có model -> lỗi
model = app.UIFigure.UserData;
```

```

        if isempty(model)
            uialert(app.UIFigure, 'Chưa có phương trình hồi quy!', 
'Lỗi');
            return;
        end
        a = model.a;
        b = model.b;
        method = model.method;
    % tính y dự đoán
    switch method
        case 'Tuyến tính'
            ypred = a * xp + b;
        case 'Hàm mũ'
            ypred = a * exp(b * xp);
        case 'Logarit'
            ypred = a + b * log(xp);
        end
    % hiện kết quả
    uialert(app.UIFigure, ['Giá trị dự đoán y(x): ' 
num2str(ypred)], 'Kết quả');
    end
end

```

1.4. THUẬT TOÁN ĐẠO HÀM

Chương trình

```

function df = finiteDiff(f, x0, h, method, order)

switch method
    case 'tien'
        if order == 1
            df = (f(x0+h)-f(x0))/h;
        else
            df = (-3*f(x0)+4*f(x0+h)-f(x0+2*h))/(2*h);
        end

    case 'lui'
        if order == 1
            df = (f(x0)-f(x0-h))/h;
        else
            df = (3*f(x0)-4*f(x0-h)+f(x0-2*h))/(2*h);
        end

    case 'trungtam'
        df = (f(x0+h)-f(x0-h))/(2*h);
    end
end

```

Chương trình

```
function kq = solveDerivative(data, x0, method, order)

    if strcmp(data.type, 'function')
        kq = finiteDiff(data.f, x0, data.h, method, order);

    else
        X = data.X;
        Y = data.Y;
        h = X(2)-X(1);

        idx = find(X==x0,1);
        if isempty(idx)
            error('x0 không tồn tại trong X');
        end

        switch method
            case 'tien'
                kq = (Y(idx+1)-Y(idx))/h;
            case 'lui'
                kq = (Y(idx)-Y(idx-1))/h;
            case 'trungtam'
                kq = (Y(idx+1)-Y(idx-1))/(2*h);
        end
    end
end
```

Chương trình

```
function data = buildInput(Xstr, Ystr, fstr, h)
% Xstr, Ystr: chuỗi nhập từ EditField
% fstr: chuỗi hàm
% h: bước

if ~isempty(Xstr) && ~isempty(Ystr)
    X = str2num(Xstr);
    Y = str2num(Ystr);

    if length(X) ~= length(Y)
        error('X và Y phải cùng kích thước');
    end

    data.type = 'table';
    data.X = X;
    data.Y = Y;

elseif ~isempty(fstr)
    syms x
    f = str2sym(fstr);

    data.type = 'function';
    data.f = matlabFunction(f);
```

```

data.h = h;

else
    error('Chưa nhập dữ liệu đầu vào');
end
end

```

1.5. THUẬT TOÁN TÍCH PHÂN

Chương trình

```

% 1. Phương pháp Hình thang (Trapezoidal) cho dữ liệu vector
function I = Trapz(x, y)
    % x, y là vector dữ liệu
    n = length(x);
    if length(y) ~= n
        error('Kích thước x và y không khớp');
    end
    I = 0;
    for i = 1:n-1
        h = x(i+1) - x(i);
        I = I + (h/2) * (y(i) + y(i+1));
    end
end

```

Chương trình

```

% 2. Phương pháp Hình thang cho Hàm số
function I = Trapz_Func(f, a, b, n)
    h = (b - a) / n;
    x = a:h:b;
    y = f(x);
    I = (h/2) * (y(1) + 2*sum(y(2:end-1)) + y(end));
end

```

Chương trình

```

% 3. Phương pháp Simpson 1/3
function I = Simpson13(f, a, b, n)
    if mod(n, 2) ~= 0
        error('Simpson 1/3 yêu cầu N phải là số chẵn.');
    end
    h = (b - a) / n;
    x = a:h:b;
    y = f(x);

```

```

% Công thức: h/3 * (y0 + 4*y_le + 2*y_chan + yn)
I = (h/3) * (y(1) + 4*sum(y(2:2:end-1)) + 2*sum(y(3:2:end-2))
+ y(end));
end

```

Chương trình

```

% 4. Phương pháp Simpson 3/8
function I = Simpson38(f, a, b, n)
if mod(n, 3) ~= 0
    error('Simpson 3/8 yêu cầu N phải chia hết cho 3.');
end
h = (b - a) / n;
x = a:h:b;
y = f(x);

% Công thức Simpson 3/8
sum_1 = sum(y(2:3:end-1)); % Các vị trí 1, 4, 7...
sum_2 = sum(y(3:3:end-1));
sum_3 = sum(y(4:3:end-2)); % Các vị trí chia hết cho 3

I = (3*h/8) * (y(1) + 3*(sum_1 + sum_2) + 2*sum_3 + y(end));
end

```

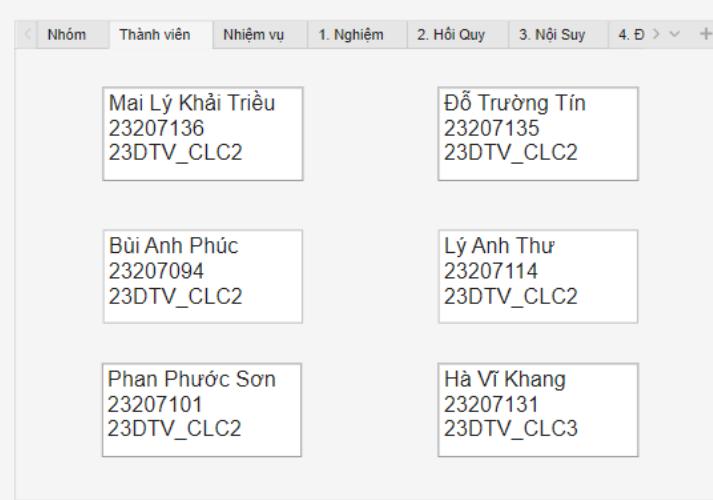
II. HOẠT ĐỘNG CỦA ỨNG DỤNG

2.1. GIAO DIỆN ỨNG DỤNG

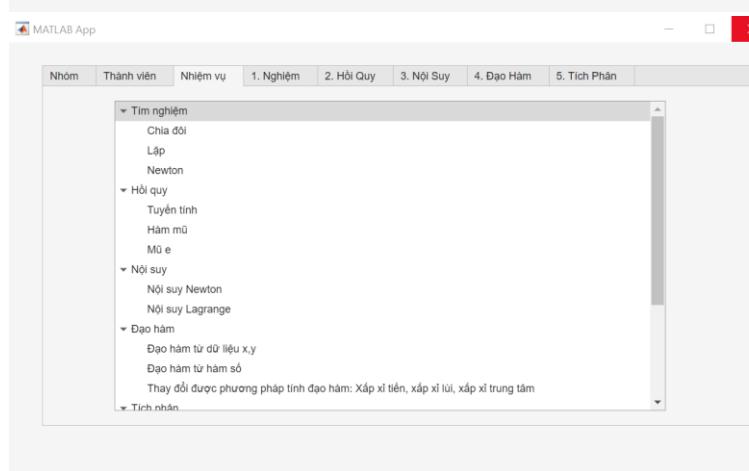
2.1.1. CHƯƠNG TRÌNH GIỚI THIỆU NHÓM



Hình 1. Giao diện giới thiệu đồ án

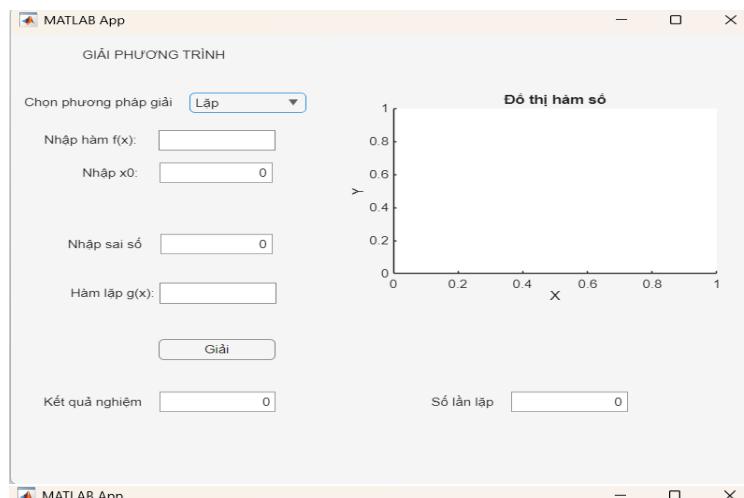


Hình 2. Giao diện giới thiệu thành viên

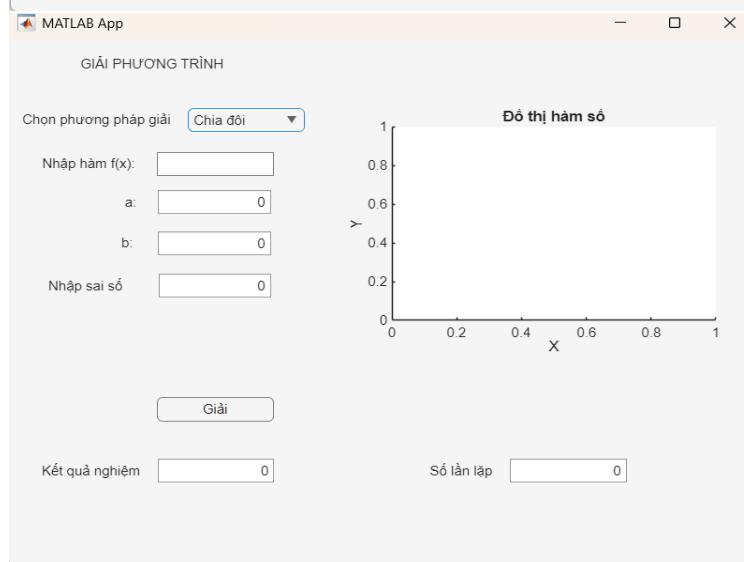


Hình 3. Giao diện giới thiệu nhiệm vụ

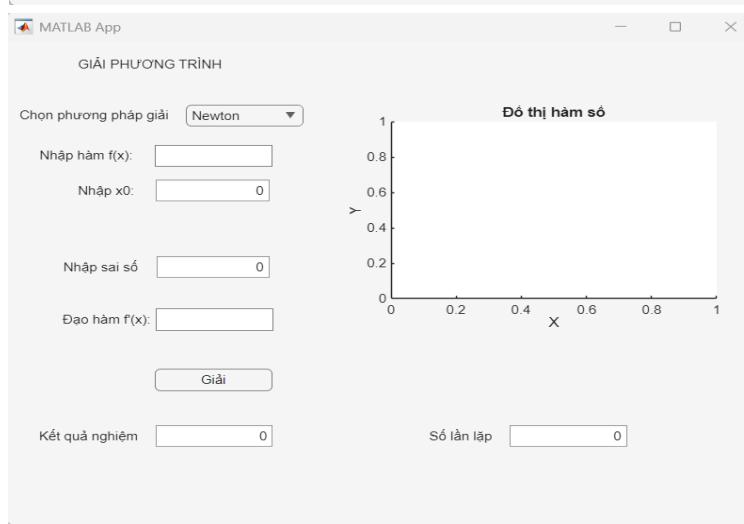
2.1.2. CHƯƠNG TRÌNH TÌM NGHIỆM PHƯƠNG TRÌNH



Hình 4. Giao diện chương trình tìm nghiệm bằng phương pháp lặp



Hình 5. Giao diện chương trình tìm nghiệm bằng phương pháp chia đôi



Hình 6. Giao diện chương trình tìm nghiệm bằng phương pháp Newton

2.1.3. CHƯƠNG TRÌNH NỘI SUY



Hình 7. Giao diện chương trình nội suy

2.1.4. CHƯƠNG TRÌNH HỒI QUY



Hình 8. Giao diện chương trình hồi quy

2.1.5. CHƯƠNG TRÌNH ĐẠO HÀM

The screenshot shows a MATLAB App window titled "MATLAB App". The tab bar at the top includes "Nhóm", "Thành viên", "Nhiệm vụ", "1. Nghiệm", "2. Hồi Quy", "3. Nội Suy", "4. Đạo Hàm", and "5. Tích Phân". The "4. Đạo Hàm" tab is selected. The interface is divided into sections:

- 1. Dữ liệu đầu vào Input:** Contains fields for "X" and "Y".
- 3. Yêu cầu:** Contains fields for "Gia trị cần Dao ham" (value 0) and "Sai so O(h) hoặc O(h^2)".
- hoặc:** Contains fields for "Ham so" and "Buoc H" (step size 0).
- RUN:** A button to execute the calculations.
- Kết quả:** A field showing the result 0.
- 2. Chọn phương pháp tính Đạo hàm:** Contains three buttons: "Xap xi Tien", "Xap xi Trung tam", and "Xap xi Lui".

Hình 9. Giao diện chương
trình đạo hàm

2.1.6. CHƯƠNG TRÌNH TÍCH PHÂN

The screenshot shows a MATLAB App window titled "MATLAB App". The tab bar at the top includes "Nhiệm vụ", "1. Nghiệm", "2. Hồi Quy", "3. Nội Suy", "4. Đạo Hàm", and "5. Tích Phân". The "5. Tích Phân" tab is selected. The interface includes:

- A "Drop Down" menu set to "Hình Thang".
- A toggle switch for "Dữ Liệu" (Data) and "Hàm số" (Function).
- Input fields for "Cận a" (lower limit 0), "Cận b" (upper limit 0), and "Số đoạn N" (number of segments 0).
- A "Tính kết quả" (Calculate result) button.
- A "Kết quả:" (Result:) field showing 0.

Hình 10. Giao diện chương
trình hồi quy

2.2. HƯỚNG DẪN SỬ DỤNG

2.1.2. CHƯƠNG TRÌNH TÌM NGHIỆM PHƯƠNG TRÌNH

Bước 1: Lựa chọn phương pháp giải

- Tại menu thả xuống "Chọn phương pháp giải", hãy chọn một trong các thuật toán: Chia đôi, Lặp, hoặc Newton (Tiếp tuyến).

Bước 2: Nhập thông tin hàm số và thông số tính toán

- Nhập hàm $f(x)$: Nhập biểu thức toán học cần tìm nghiệm (Ví dụ: $x^2 - 2$).
- Nhập khoảng/giá trị ban đầu:
 - Nhập giá trị a và b nếu sử dụng phương pháp Chia đôi (khoảng phân ly nghiệm).
 - Nhập giá trị ban đầu nếu sử dụng phương pháp Newton.
- Nhập sai số: Điền giá trị sai số cho phép vào ô "Nhập sai số" để xác định độ chính xác mong muốn.
- Nhập hàm lặp $g(x)$: (Chỉ dành cho phương pháp Lặp) Nhập hàm lặp tương ứng đã biến đổi từ $f(x) = 0$ thành $x = g(x)$.

Bước 3: Thực hiện tính toán

- Nhấn nút "Giải" để hệ thống bắt đầu thực hiện các vòng lặp tính toán.

Bước 4: Quan sát kết quả

- Kết quả nghiệm: Giá trị nghiệm tìm được sẽ hiển thị tại ô này.
- Số lần lặp: Tổng số bước lặp mà thuật toán đã thực hiện để đạt được sai số cho phép.
- Đồ thị hàm số: Vùng bên phải sẽ hiển thị trực quan đường cong của hàm số $f(x)$ giúp người dùng dễ dàng quan sát sự giao cắt với trục hoành.

2.1.3. CHƯƠNG TRÌNH NỘI SUY

Bước 1: Nhập dữ liệu đầu vào

- Dữ liệu x: Nhập tập hợp các giá trị biến độc lập dưới dạng vector
- Dữ liệu y: Nhập tập hợp các giá trị hàm số tương ứng
 - Lưu ý: Số lượng phần tử của x và y phải bằng nhau.

Bước 2: Lựa chọn phương pháp nội suy

- Sử dụng menu thả xuống (Dropdown) để chọn thuật toán phù hợp:

- Lagrange: Sử dụng đa thức nội suy Lagrange.
- Newton: Sử dụng đa thức nội suy Newton

Bước 3: Thực hiện tính toán và xem đa thức

- Sau khi nhập đủ dữ liệu, hệ thống tính toán.
- Đa thức nội suy: Kết quả biểu thức toán học của đa thức sẽ hiển thị

Bước 4: Dự đoán giá trị

- Nhập giá trị cần dự đoán: Nhập một giá trị x_0 bất kỳ vào ô trống để tính giá trị $P(x_0)$ tương ứng.
- Nhấn nút "Kết quả nội suy" để xem giá trị dự đoán cuối cùng.

Bước 5: Quan sát đồ thị

- Vùng đồ thị phía bên phải sẽ hiển thị:
 - Các điểm dữ liệu thực tế
 - Đường cong đa thức nội suy đi qua các điểm đó giúp người dùng đánh giá trực quan độ chính xác của mô hình.

2.1.4. CHƯƠNG TRÌNH HỒI QUY

Bước 1: Nhập dữ liệu đầu vào

Giá trị x: Nhập danh sách các biến độc lập.

Giá trị y: Nhập danh sách các biến phụ thuộc tương ứng với x. Số lượng giá trị y phải bằng số lượng giá trị x.

Bước 2: Chọn phương pháp hồi quy

Tại mục Chọn PP hồi quy, nhấn vào menu thả xuống (dropdown) để chọn mô hình toán học bạn muốn áp dụng

Bước 3: Tính toán và xem đồ thị

Nhấn nút Kết quả phương trình hồi quy:

Hệ thống sẽ tính toán các hệ số và hiển thị phương trình toán học cụ thể.

Đồ thị: Khu vực bên phải sẽ vẽ các điểm dữ liệu thực tế và đường hồi quy để bạn trực quan hóa độ chính xác của mô hình.

Bước 4: Dự đoán giá trị

Nhập giá trị cần dự đoán: Sau khi đã có phương trình, bạn nhập một giá trị x mới vào ô này.

Nhấn nút Kết quả dự đoán: Hệ thống sẽ thay x vào phương trình đã tìm được ở Bước 3 để đưa ra kết quả y tương ứng.

2.1.5. CHƯƠNG TRÌNH ĐẠO HÀM

Bước 1: Nhập dữ liệu đầu vào (Input)

Bạn có hai lựa chọn để cung cấp dữ liệu:

Lựa chọn 1 (Dữ liệu mảng): Nhập các tập giá trị vào ô X và Y

Lựa chọn 2 (Biểu thức hàm số): Nhập biểu thức vào ô Hàm số (Ví dụ: $x^2 + \sin(x)$).

Nhập giá trị khoảng cách giữa các điểm vào ô Bước H (độ phân giải của phép tính).

Bước 2: Chọn phương pháp tính Đạo hàm

Dựa trên vị trí của điểm cần tính đạo hàm trong tập dữ liệu, bạn chọn một trong ba nút:

Xấp xỉ Tiên: Dùng khi muốn tính đạo hàm dựa trên điểm hiện tại và điểm kế tiếp.

Xấp xỉ Trung tâm: Thường cho độ chính xác cao nhất, dùng các điểm xung quanh điểm cần tính.

Xấp xỉ Lùi: Dùng khi muốn tính đạo hàm dựa trên điểm hiện tại và điểm đứng trước nó.

Bước 3: Thiết lập yêu cầu và Chạy chương trình

Giá trị cần Đạo hàm: Nhập giá trị x0 mà tại đó bạn muốn tính đạo hàm.

Sai số O(h) hoặc O(h²): Nhập bậc sai số mong muốn hoặc ngưỡng sai số cho phép để phần mềm tối ưu kết quả.

Nút RUN: Nhấn nút này để thực hiện tính toán.

Bước 4: Xem kết quả

Kết quả cuối cùng của đạo hàm f'(x0) sẽ được hiển thị tại ô Kết quả.

2.1.6. CHƯƠNG TRÌNH TÍCH PHÂN

1. Lựa chọn phương pháp tính

Tại mục Drop Down, nhấn vào mũi tên để chọn phương pháp tính toán.

2. Lựa chọn chế độ nhập liệu (Dữ Liệu / Hàm Số)

Nút gạt phía trên bên phải cho phép chọn một trong hai chế độ:

Chế độ Dữ Liệu (Data): Dùng khi bạn có một bảng giá trị x và y rời rạc.

Chế độ Hàm Số (Function): Dùng khi bạn có biểu thức toán học cụ thể (ví dụ: $f(x) = x^2 + 2x$).

3. Nhập các thông số đầu vào

Điền đầy đủ các thông tin sau vào các ô trống bên phải:

Cận a: Giới hạn dưới của tích phân.

Cận b: Giới hạn trên của tích phân.

Số đoạn N: Số khoảng chia nhỏ để tính toán. Giá trị N càng lớn thì kết quả càng chính xác.

Các ô trống bên trái: Tùy thuộc vào chế độ bạn chọn, đây sẽ là nơi nhập biểu thức hàm số hoặc danh sách các tọa độ điểm.

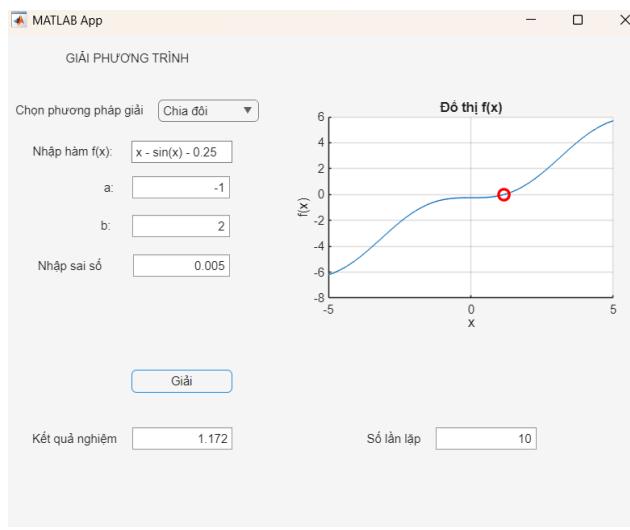
4. Tính toán và Xem kết quả

Sau khi nhập đầy đủ thông số, nhấn nút "Tính kết quả".

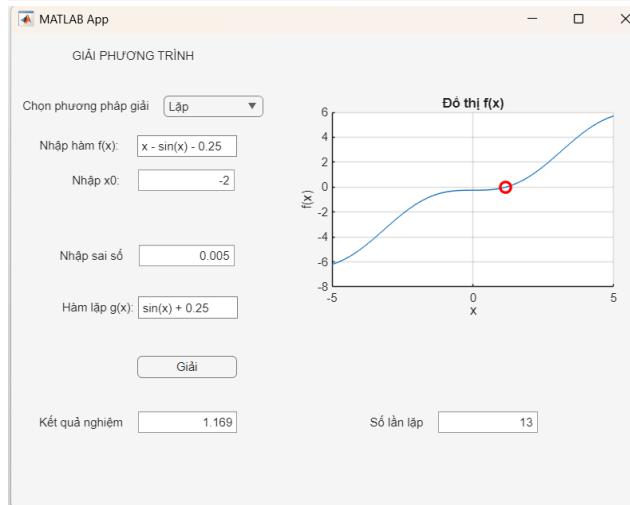
Giá trị cuối cùng sẽ hiển thị tại ô "Kết quả:" ở góc dưới cùng bên phải

III. KẾT QUẢ THỰC NGHIỆM

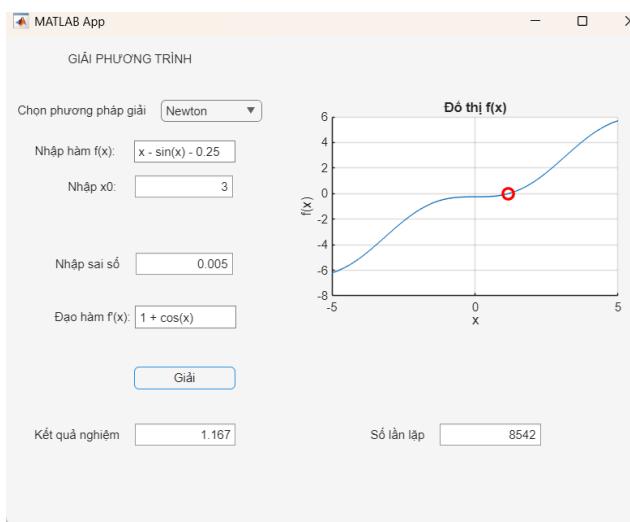
3.1. CHƯƠNG TRÌNH TÌM NGHIỆM PHƯƠNG TRÌNH



Hình 11. Kết quả thực nghiệm chương trình tính nghiệm bằng phương pháp chia đôi

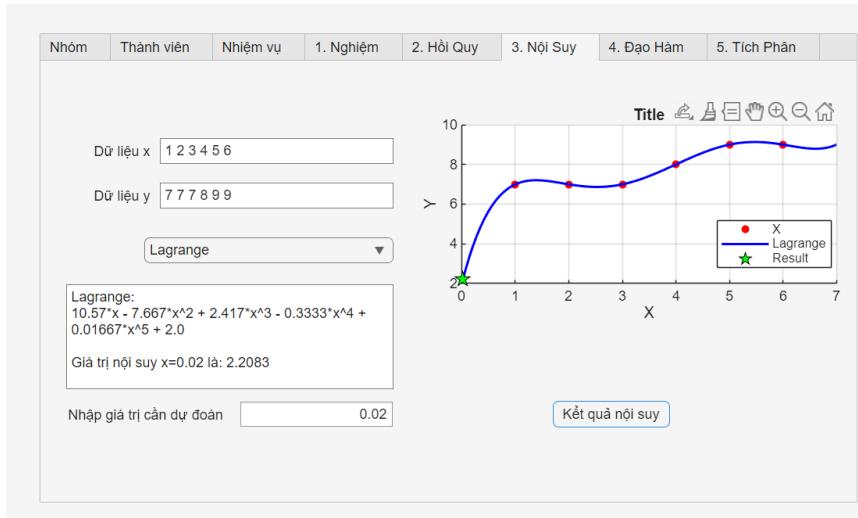


Hình 12. Kết quả thực nghiệm chương trình tính nghiệm bằng phương pháp lập đơn

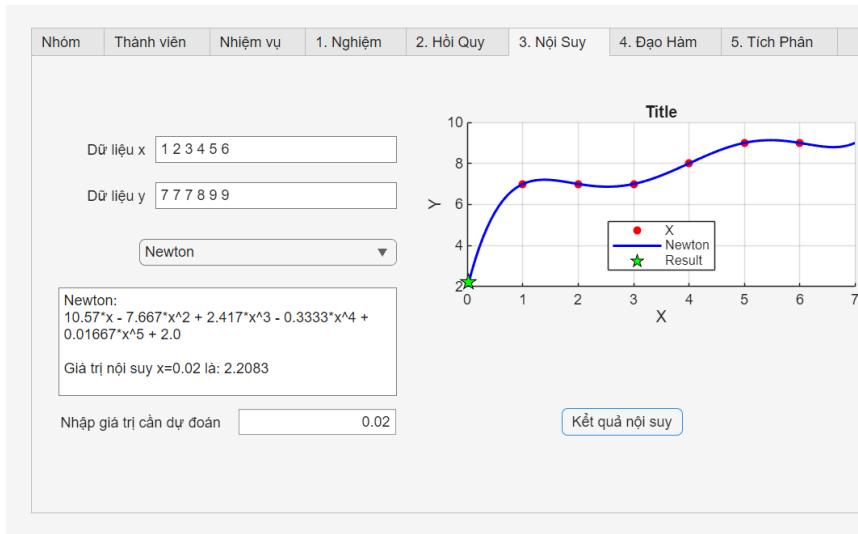


Hình 13. Kết quả thực nghiệm chương trình tính nghiệm bằng phương pháp Newton

3.2. CHƯƠNG TRÌNH NỘI SUY

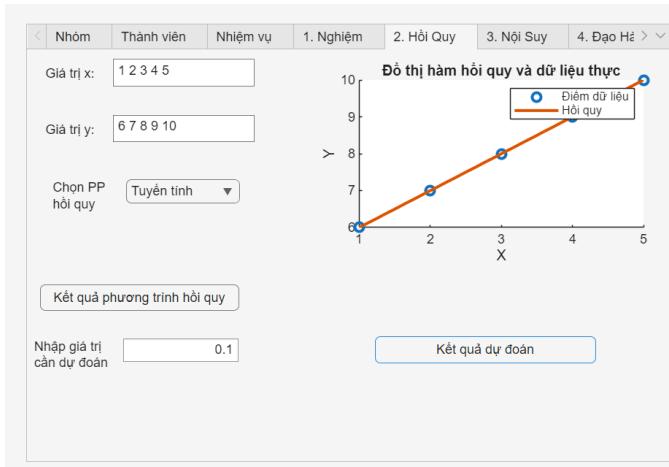


Hình 14. Kết quả thực nghiệm
chương trình nội suy bằng
Lagrange

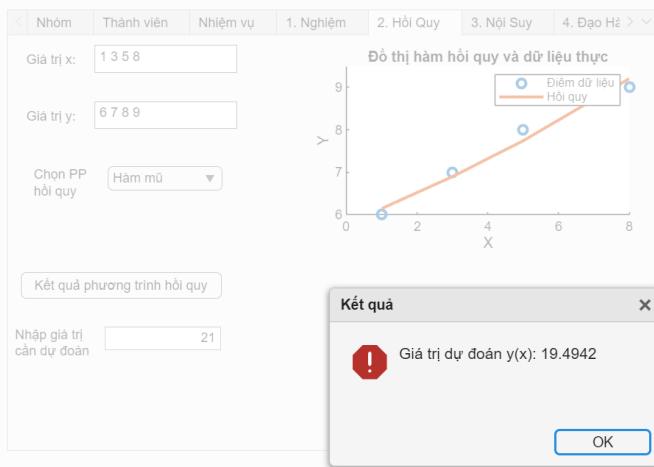


Hình 15. Kết quả thực nghiệm
chương trình nội suy bằng Newton

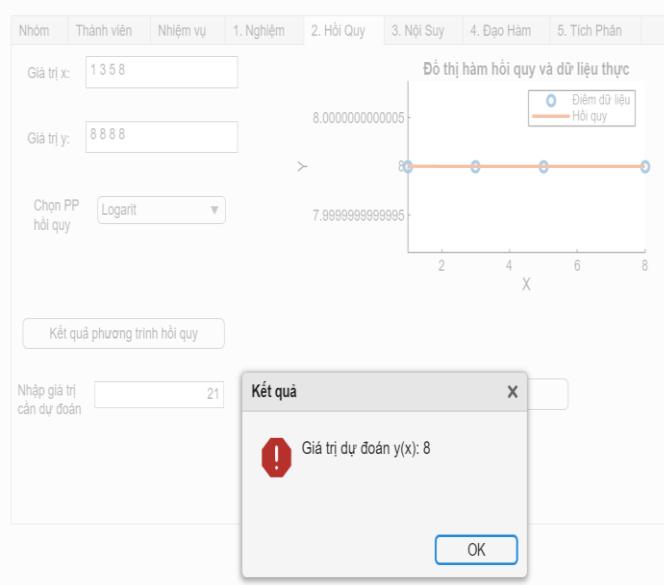
3.3. CHƯƠNG TRÌNH HỒI QUY



Hình 16. Kết quả thực nghiệm chương trình hồi quy tuyến tính



Hình 17. Kết quả thực nghiệm chương trình hồi quy (Hàm mũ)



Hình 18. Kết quả thực nghiệm chương trình hồi quy (Hàm Logarit)

3.4. CHƯƠNG TRÌNH ĐẠO HÀM

| Nhóm | Thành viên | Nhiệm vụ | 1. Nghiệm | 2. Hồi Quy | 3. Nội Suy | 4. Đạo Hàm | 5. Tích Phản |
|--|--------------------------------------|----------|---------------------|------------------------------------|------------|------------|--------------|
| 1.Dữ liệu đầu vào Input | | | | | | | |
| X | <input type="text" value="1 2 3 4"/> | | 3. Yêu cầu | | | | |
| Y | <input type="text" value="4 5 6 5"/> | | Gia trị cần Đạo hàm | <input type="text" value="2"/> | | | |
| hoặc | | | | | | | |
| Ham so | <input type="text"/> | | | <input type="button" value="RUN"/> | | | |
| Buoc H | <input type="text" value="0"/> | | Ket qua | <input type="text" value="1"/> | | | |
| 2.Chọn phương pháp tính Đạo hàm | | | | | | | |
| <input type="button" value="Xáp xi Tiên"/> <input type="button" value="Xáp xi Trung tam"/> <input type="button" value="Xáp xi Lui"/> | | | | | | | |

Hình 19. Kết quả thực nghiệm chương trình đạo hàm (Xáp xi trung tâm)

| Nhóm | Thành viên | Nhiệm vụ | 1. Nghiệm | 2. Hồi Quy | 3. Nội Suy | 4. Đạo Hàm | 5. Tích Phản |
|--|--------------------------------------|----------|------------------------------------|----------------------------------|------------|------------|--------------|
| 1.Dữ liệu đầu vào Input | | | | | | | |
| X | <input type="text"/> | | 3. Yêu cầu | | | | |
| Y | <input type="text"/> | | Gia trị cần Đạo hàm | <input type="text" value="2"/> | | | |
| hoặc | | | | | | | |
| Ham so | <input type="text" value="x^2 + 2"/> | | <input type="button" value="RUN"/> | | | | |
| Buoc H | <input type="text" value="0.2"/> | | Ket qua | <input type="text" value="4.2"/> | | | |
| 2.Chọn phương pháp tính Đạo hàm | | | | | | | |
| <input type="button" value="Xáp xi Tiên"/> <input type="button" value="Xáp xi Trung tam"/> <input type="button" value="Xáp xi Lui"/> | | | | | | | |

Hình 20. Kết quả thực nghiệm chương trình đạo hàm (Xáp xi tiên)

| Nhóm | Thành viên | Nhiệm vụ | 1. Nghiệm | 2. Hồi Quy | 3. Nội Suy | 4. Đạo Hàm | 5. Tích Phản |
|--|--------------------------------------|----------|------------------------------------|----------------------------------|------------|------------|--------------|
| 1.Dữ liệu đầu vào Input | | | | | | | |
| X | <input type="text"/> | | 3. Yêu cầu | | | | |
| Y | <input type="text"/> | | Gia trị cần Đạo hàm | <input type="text" value="2"/> | | | |
| hoặc | | | | | | | |
| Ham so | <input type="text" value="x^2 + 2"/> | | <input type="button" value="RUN"/> | | | | |
| Buoc H | <input type="text" value="0.2"/> | | Ket qua | <input type="text" value="3.8"/> | | | |
| 2.Chọn phương pháp tính Đạo hàm | | | | | | | |
| <input type="button" value="Xáp xi Tiên"/> <input type="button" value="Xáp xi Trung tam"/> <input type="button" value="Xáp xi Lui"/> | | | | | | | |

Hình 21. Kết quả thực nghiệm chương trình đạo hàm (Xáp xi lùi)

3.5. CHƯƠNG TRÌNH TÍCH PHÂN

The screenshot shows a software window with a tab bar at the top: 'Nhiệm vụ' (Task), '1. Nghiệm', '2. Hồi Quy', '3. Nội Suy', '4. Đạo Hàm', and '5. Tích Phân'. The '5. Tích Phân' tab is selected. Below the tabs, there is a dropdown menu labeled 'Drop Down' with 'Hình Thang' selected. A toggle switch labeled 'Dữ Liệu' is set to 'Hàm số'. The interface contains several input fields:

- Mảng X: Input field containing '1 2 3 4'.
- Cận a: Input field containing '1'.
- Mảng Y: Input field containing '4 3 2 1'.
- Cận b: Input field containing '2'.
- f(x): Input field containing an empty box.
- Số đoạn N: Input field containing '10'.
- A button labeled 'Tính kết quả' (Calculate result).
- An output field labeled 'Kết quả:' containing '7.5'.

Hình 22. Kết quả thực nghiệm chương
trình tích phân (Hình thang)

IV. ĐÁNH GIÁ, NHẬN XÉT

4.1. ĐÁNH GIÁ

4.1.1. ƯU ĐIỂM

- **Giao diện trực quan:** Ứng dụng được thiết kế theo cấu trúc Tab hiện đại, giúp phân loại rõ ràng các chủ đề toán học như Nghiệm, Nội suy, Hồi quy, Đạo hàm và Tích phân.
- **Tính tương tác cao:** Người dùng có thể dễ dàng nhập phương trình, dữ liệu x, y và nhận kết quả tức thời cùng với đồ thị minh họa sinh động.
- **Đa dạng phương pháp:** Tích hợp đầy đủ các thuật toán cốt lõi từ Chia đôi, Newton đến các quy tắc phức tạp như Simpson 1/3 và 3/8.
- **Tính module hóa:** Các phương pháp được viết dưới dạng hàm riêng biệt, giúp mã nguồn dễ quản lý, bảo trì và tối ưu hóa hiệu suất tính toán.
- **Quản lý chuyên nghiệp:** Toàn bộ mã nguồn được lưu trữ và quản lý phiên bản qua GitHub, đảm bảo tính cộng tác hiệu quả trong nhóm.

4.1.2. HẠN CHẾ

- **Xử lý lỗi nhập liệu:** Ứng dụng đôi khi còn gặp lỗi khi người dùng nhập các hàm số quá phức tạp hoặc định dạng dữ liệu không đồng nhất.
- **Phụ thuộc nền tảng:** Hiện tại ứng dụng chỉ chạy được trong môi trường Matlab, chưa được đóng gói thành phần mềm độc lập (.exe) cho người dùng không có Matlab.
- **Tốc độ với dữ liệu lớn:** Khi xử lý các bài toán hồi quy hoặc nội suy với hàng nghìn điểm dữ liệu, tốc độ phản hồi của giao diện có thể bị chậm lại.

4.2. NHẬN XÉT

Through the process of implementing the project, the group found that Matlab App Designer is a powerful tool for integrating mathematical concepts with user interface design. The project does not only help students understand various numerical methods but also provides them with practical experience in software development. However, there are still some limitations such as dependency on Matlab, slow performance with large data sets, and lack of portability. Overall, the application has met most of the specified requirements and has a promising future in education and research.

V. HƯỚNG PHÁT TRIỂN

Dù ứng dụng đã hoàn thành các mục tiêu cơ bản về tính toán và mô phỏng giao diện, nhóm nhận thấy vẫn còn nhiều tiềm năng để mở rộng và nâng cấp. Dưới đây là các hướng phát triển chính mà nhóm đề xuất:

5.1. MỞ RỘNG THƯ VIỆN BÀI TOÁN TOÁN HỌC

- Giải hệ phương trình vi phân: Tích hợp thêm các phương pháp như Runge-Kutta hoặc Euler để giải quyết các bài toán động lực học thực tế.
- Tối ưu hóa: Bổ sung các thuật toán tìm giá trị cực đại, cực tiểu của hàm số đa biến.
- Đại số tuyến tính nâng cao: Phát triển thêm Tab giải hệ phương trình tuyến tính quy mô lớn bằng các phương pháp lặp Gauss-Seidel hoặc Jacobi.

5.2. NÂNG CẤP TRẢI NGHIỆM NGƯỜI DÙNG (UI/UX)

- Tính năng Import/Export dữ liệu: Cho phép người dùng nhập dữ liệu trực tiếp từ các file Excel (.xlsx) hoặc Text (.txt) thay vì nhập thủ công từng điểm dữ liệu.
- Xuất báo cáo kết quả: Xây dựng chức năng xuất kết quả tính toán và đồ thị ra file PDF hoặc Word để phục vụ việc lưu trữ và học tập.
- Giao diện đáp ứng (Responsive Layout): Tối ưu hóa giao diện App để có thể hiển thị tốt trên nhiều kích thước màn hình khác nhau mà không bị vỡ bô cục.

5.3. TỐI ƯU HÓA THUẬT TOÁN VÀ HIỆU SUẤT

- Xử lý sai số thông minh: Tự động đưa ra cảnh báo hoặc gợi ý phương pháp phù hợp khi hàm số khó hội tụ hoặc dữ liệu đầu vào có sai số lớn.
- Tính toán song song (Parallel Computing): Sử dụng các thư viện hỗ trợ của Matlab để tăng tốc độ tính toán đối với các bài toán nội suy hoặc hồi quy trên tập dữ liệu lớn.

5.4. ĐÓNG GÓI VÀ PHÂN PHỐI ỨNG DỤNG

- Matlab Web App Server: Triển khai ứng dụng lên Web để người dùng có thể truy cập và sử dụng trực tiếp qua trình duyệt mà không cần cài đặt phần mềm Matlab.
- Standalone Application: Đóng gói ứng dụng thành file thực thi (.exe) chạy độc lập trên hệ điều hành Windows.

VI. ĐÁNH GIÁ NHÓM

6.1. PHÂN CÔNG CÔNG VIỆC

| STT | Họ và tên | MSSV | Nhiệm vụ |
|-----|-------------------|----------|--|
| 1 | Mai Lý Khải Triều | 23207136 | Trưởng nhóm, Thiết kế chương trình đạo hàm |
| 2 | Đỗ Trường Tín | 23207135 | Thiết kế chương trình tìm nghiệm phương trình |
| 3 | Bùi Anh Phúc | 23207094 | Thiết kế chương trình tìm nghiệm nội suy |
| 4 | Lý Anh Thư | 23207114 | Thiết kế chương trình tìm nghiệm hồi quy |
| 5 | Phan Phước Sơn | 23207101 | Thiết kế chương trình tích phân |
| 6 | Hà Vĩ Khang | 23207131 | Viết báo cáo, Thiết kế chương trình nhóm |

Bảng 1. Phân công nhiệm vụ

6.2. TIÊU CHÍ ĐÁNH GIÁ

| STT | Các kỹ năng |
|-----|--|
| 1 | Kỹ năng làm việc nhóm, giao tiếp, hành xử chuyên nghiệp, khả năng lãnh đạo và làm việc độc lập |
| 2 | Kỹ năng tư duy phản biện |
| 3 | Kỹ năng thuyết trình |
| 4 | Giao tiếp kỹ thuật (viết báo cáo kỹ thuật) |
| 5 | Kỹ năng tư duy sáng tạo |
| 6 | Kỹ năng quản lý dự án/thời gian thực hiện dự án |
| 7 | Hình thành nội dung, xác định vấn đề và kỹ năng giải quyết vấn đề |
| 8 | Kiến thức, thực nghiệm qua đồ án môn học |

Bảng 2. Tiêu chí đánh giá

6.3. KẾT QUẢ ĐÁNH GIÁ

| STT | Các kỹ năng | Đánh giá (*) |
|-----|--|--------------|
| 1 | Kỹ năng làm việc nhóm, giao tiếp, hành xử chuyên nghiệp, khả năng lãnh đạo và làm việc độc lập | A |
| 2 | Kỹ năng tư duy phản biện | A |
| 3 | Kỹ năng thuyết trình | B |
| 4 | Giao tiếp kỹ thuật (viết báo cáo kỹ thuật) | B |
| 5 | Kỹ năng tư duy sáng tạo | A |
| 6 | Kỹ năng quản lý dự án/thời gian thực hiện dự án | A |
| 7 | Hình thành nội dung, xác định vấn đề và kỹ năng giải quyết vấn đề | A |
| 8 | Kiến thức, thực nghiệm qua đồ án môn học | A |

Bảng 3. Kết quả đánh giá nhóm

| Họ và tên | MSSV | Mức độ hoàn thành công việc | Tỷ lệ đóng góp (%) |
|-------------------|----------|-----------------------------|--------------------|
| Mai Lý Khải Triều | 23207136 | 90% | 15% |
| Đỗ Trường Tín | 23207135 | 100% | 15% |
| Bùi Anh Phúc | 23207094 | 100% | 20% |
| Lý Anh Thư | 23207114 | 100% | 20% |
| Phan Phước Sơn | 23207101 | 100% | 15% |
| Hà Vĩ Khang | 23207131 | 90% | 15% |

Bảng 4. Kết quả đánh giá thành viên

VII. PHỤ LỤC

7.1. Thông tin lưu trữ trực tuyến (GitHub)

Đường dẫn: https://github.com/khaitrieu/CK_FIRE