**ĐẠI HỌC ĐÀ NẴNG**

**ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**PBL1. ĐỒ ÁN**

**LẬP TRÌNH TÍNH TOÁN**

**Đề tài 205:**

**Dùng phương pháp Gauss để giải hệ phương trình**

**gồm n phương trình , n ẩn:  A.X=B**

**GIẢNG VIÊN HƯỚNG DẪN: Đỗ Thị Tuyết Hoa**

**SINH VIÊN THỰC HIỆN:**

**Trần Nhật Nguyên LỚP: 23T\_DT1 NHÓM: 23.Nh15B**

**Vũ Đức Minh LỚP: 23T\_DT2 NHÓM: 23.Nh15B**

**Đà Nẵng, tháng 05/2024**

LỜI MỞ ĐẦU

Hệ phương trình tuyến tính đóng vai trò vô cùng quan trọng trong nhiều lĩnh vực khoa học, kỹ thuật và đời sống. Việc giải quyết các hệ phương trình này là một bài toán cơ bản và cần thiết. Trong số các phương pháp giải hệ phương trình tuyến tính, phương pháp Gauss là một phương pháp phổ biến và hiệu quả, đặc biệt cho các hệ phương trình có nhiều ẩn.

Mục đích thực hiện đề tài “Dùng phương pháp Gauss để giải hệ phương trình gồm n phương trình, n ẩn:  A.X=B” là tạo ra một chương trình máy tính sử dụng phương pháp Gauss để giải các hệ phương trình nói trên.

Mục tiêu của đề tài “Dùng phương pháp Gauss để giải hệ phương trình gồm n phương trình, n ẩn:  A.X=B” là tạo ra một chương trình tính toán chính xác và đạt hiệu quả cao trong quá trình giải hệ phương trình tuyến tính. Cụ thể:

+) Xây dựng giao diện đơn giản.

+) Cho phép nhập các ma trận từ bàn phím người dung và từ các tệp dữ liệu đầu vào.

+) Cho phép thực hiện tính toán và tìm nghiệm trên ma trận nhập vào.

+) Cung cấp tính năng thực hiện các bước biến đổi ma trận về dạng bậc thang.

+) Cung cấp tính năng cộng dồn các phần tử phía sau cột hệ số tự do để tạo thành hệ số tự do mới.

+) Cho phép xuất kết quả tìm được ra tệp dữ liệu đầu ra.

Phạm vi áp dụng của chương trình tính nói riêng và phương pháp Gauss nói chung trải rộng trên mọi hệ phương trình tuyến tính, bất kể số lượng và loại biến số.

Báo cáo này trình bày chi tiết về chương trình tính toán nghiệm của hệ phương trình tuyến tính và các tính năng và giao diện của chương trình.

MỤC LỤC

[LỜI MỞ ĐẦU 2](#_Toc49109152)

[MỤC LỤC 3](#_Toc49109153)

[DANH MỤC HÌNH VẼ 4](#_Toc49109154)

[1. GIỚI THIỆU ĐỀ TÀI 5](#_Toc49109155)

[2. CƠ SỞ LÝ THUYẾT 5](#_Toc49109156)

[2.1. Ý tưởng 5](#_Toc49109157)

[2.2. Cơ sở lý thuyết 5](#_Toc49109158)

[3. CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN 5](#_Toc49109159)

[3.1. Phát biểu bài toán 5](#_Toc49109160)

[3.2. Cấu trúc dữ liệu 5](#_Toc49109161)

[3.3. Thuật toán 5](#_Toc49109162)

[4. CHƯƠNG TRÌNH VÀ KẾT QUẢ 5](#_Toc49109163)

[4.1. Tổ chức chương trình 5](#_Toc49109164)

[4.2. Ngôn ngữ cài đặt 5](#_Toc49109165)

[4.3. Kết quả 5](#_Toc49109166)

[4.3.1. Giao diện chính của chương trình 5](#_Toc49109167)

[4.3.2. Kết quả thực thi của chương trình 5](#_Toc49109168)

[4.3.3. Nhận xét 5](#_Toc49109169)

[5. KẾT LUẬN 5](#_Toc49109170)

[5.1. Kết quả đạt được 5](#_Toc49109171)

[5.2. Hạn chế 5](#_Toc49109172)

[5.3. Hướng phát triển 6](#_Toc49109173)

[TÀI LIỆU THAM KHẢO 7](#_Toc49109174)

DANH MỤC HÌNH VẼ

No table of figures entries found.

# GIỚI THIỆU ĐỀ TÀI

Đề tài “Dùng phương pháp Gauss để giải hệ phương trình gồm n phương trình, n ẩn:  A.X=B” là một đề tài trong lĩnh vực Công nghệ thông tin và Khoa học Máy Tính. Đề tài này tập trung vào việc giải và tìm nghiệm hệ phương trình tuyến tính n ẩn, n phương trình.

# CƠ SỞ LÝ THUYẾT

## Ý tưởng

## Cơ sở lý thuyết

Cơ sở lý thuyết cho đề tài “Dùng phương pháp Gauss để giải hệ phương trình gồm n phương trình, n ẩn:  A.X=B” là:

+) File I/O: Sử dụng các thao tác đọc/ghi file để truy xuất dữ liệu ma trận và lưu trữ kết quả tính toán.

+) Các cấu trúc dữ liệu: Sử dụng cấu trúc dữ liệu mảng hai chiều để lưu trữ và quản lý ma trận; Sử dụng cấu trúc dữ liệu danh sách để lưu trữ và quản lý thực đơn của chương trình.

+) Thuật toán giải hệ phương trình tuyến tính bằng phương pháp Gauss để giải và tìm nghiệm của hệ phương trình tuyến tính đã cho.

+) Thiết kế giao diện: Thiết kế giao diện đơn giản.

# CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

## Phát biểu bài toán

Đầu vào (Input) của chương trình bao gồm: Dữ liệu của ma trận đầu vào bao gồm số dòng, số cột và các phần tử ma trận.

Đầu ra (Output) của chương trình bao gồm: Dữ liệu của ma trận được tính toán và nghiệm của hệ phương trình tuyến tính.

Bài toán đề tài “Dùng phương pháp Gauss để giải hệ phương trình gồm n phương trình, n ẩn:  A.X=B” thực hiện các công việc chính sau:

+) Thực hiện đọc ma trận cấp n x m từ tệp đầu vào hoặc từ bàn phím.

+) Cộng dồn các phần tử phía sau kể tử cột n + 1 trở đi để trở thành cột hệ số tự do tương ứng.

+) Thực hiện các tính toán trên ma trận bao gồm: biến đổi ma trận về dạng bậc thang; giải hệ phương trình tuyến tính ứng với ma trận; xuất kết quả của chương trình ra màn hình và tệp đầu ra.

## Cấu trúc dữ liệu

Để xây dựng chương trình cho đề tài “Dùng phương pháp Gauss để giải hệ phương trình gồm n phương trình, n ẩn:  A.X=B” cần có các thành phần:

+) Các định nghĩa, hằng:

|  |
| --- |
| #define ENTER1 10  #define ENTER2 13  #define MAX 100  #define CHAR\_FORMAT "%7c"  #define DATA\_FORMAT "%7.2f"  #define INPUT\_DIR "INPUT/"  #define LOG\_PATH "log/log.txt"  #define INPUT\_PATH "DATA.INP"  #define OUTPUT\_PATH "OUTPUT/DATA.OUT" |

+) Các kiểu dữ liệu được định nghĩa của chương trình.

1. ‘*string*’: kiểu dữ liệu chuỗi.
2. ‘*Matrix*’: kiểu dữ liệu mảng hai chiều biểu diễn ma trận.
3. ‘*Vector*’: kiểu dữ liệu mảng một chiều biểu diễn vector.
4. ‘*bool*’: kiểu dữ liệu logic có hai giá trị là đúng (*true*, 1) và sai (*false*, 0).
5. ‘*func*’: kiểu dữ liệu hàm.
6. ‘*MatrixRecord*’: kiểu dữ liệu cấu trúc của bản ghi các thông tin của ma trận bao gồm ‘*n*’ (số hang), ‘*m*’ (số cột) và ‘*matrix*’ (ma trận).
7. ‘*MatrixCalculationRecord*’: kiểu dữ liệu cấu trúc của bản ghi các thông tin tính toán được từ ma trận bao gồm:
8. ‘*rank*’: hạng của ma trận hệ số.
9. ‘*rank\_mr*’: hạng của ma trận hệ số mở rộng.
10. ‘*nghiem*’: vector nghiệm của hệ phương trình, nếu hệ phương trình vô nghiệm hoặc có vô số nghiệm thì không được gán giá trị.
11. ‘*det*’: giá trị của định thức của ma trận.
12. ‘*MenuOption*’: kiểu dữ liệu cấu trúc của một lựa chọn trên thực đơn, bao gồm các thuộc tính:
13. ‘*label*’: nhãn của lựa chọn, hiển thị cho người dùng nhìn thấy.
14. ‘*action*’: hành động được thực hiện sau khi người dùng chọn.
15. ‘*Menu*’: một List ADT của các lựa chọn trên thực đơn, bao gồm một số thuộc tính cơ bản của List ADT: “*list*” (mảng các lựa chọn) và “*count*” (số lượng các lựa chọn).

+) Các cờ hiệu được định nghĩa để điều khiển chương trình:

1. Cờ ‘*allow\_color\_showing\_in\_matrix*’: nhận hai giá trị đúng hoặc sai, nếu đúng thì chương trình được phép hiển thị màu khi in ma trận. Giá trị mặc định là sai.
2. Cờ ‘*allow\_matrix\_separator*’: nhận hai giá trị đúng hoặc sai, nếu đúng thì chương trình cho phép hiển thị dấu phân cách trước cột hệ số tự do. Giá trị mặc định là đúng.
3. Cờ ‘*pivot\_row*’: nhận các giá trị nguyên, lưu chỉ số hiện tại của dòng biến đổi trong quá trình biến đổi ma trận về dạng bậc thang. Giá trị mặc định là -1.
4. Cờ ‘*dest\_row*’: nhận các giá trị nguyên, lưu chỉ số hiện tại của dòng bị biến đổi trong quá trình biến đổi ma trận về dạng bậc thang. Giá trị mặc định là -1.
5. Cờ ‘*current\_row*’: nhận các giá trị nguyên, lưu chỉ số lớn nhất có thể có của những dòng đã được biến đổi. Giá trị mặc định là -1.
6. Cờ ‘*ENABLE\_COLOR*’: nhận hai giá trị 0 hoặc 1, nếu nhận giá trị 1 thì chương trình cho phép hiển thị các màu sắc khác trên màn hình bao gồm: *RED* (đỏ), *GREEN* (xanh lá), *YELLOW* (vàng), *BLUE* (xanh) và *RESET* (trở về màu bình thường). Giá trị mặc định là 1.

+) Các biến toàn cục được khai báo trong chương trình:

1. ‘*log\_file*’: tệp lưu các thông báo của chương trình nhưng không được in ra màn hình (*stdout*).
2. ‘*menu*’: thực đơn chính của chương trình.
3. ‘*current*’: bản ghi ma trận hiện tại trên chương trình được dùng để tính toán.
4. ‘*calculation*’: bản ghi ma trận đã dược biến đổi của ‘*current*’ (sử dụng trong quá trình nhập ma trận).
5. ‘*calculation\_result*’: bản ghi kết quả tính toán của ma trận (sử dụng trong quá trình nhập ma trận).

## Thuật toán

Thuật toán của bài toán “Dùng phương pháp Gauss để giải hệ phương trình gồm n phương trình, n ẩn:  A.X=B” được thể hiện qua:

### Thuật toán sao chép ma trận.

+) Thuật toán lặp từng gán phần tử của ma trận nguồn (src) sang ma trận đích (dest).

+) Độ phức tạp thuật toán sao chép ma trận là .

### Thuật toán hoán vị hai số và hai dòng.

+) Thuật toán hoán vị hai số bằng phương pháp đặt ẩn phụ có độ phức tạp thuật toán là .

+) Thuật toán hoán vị hai dòng bằng phương pháp hoán vị từng phần tử của hai dòng có độ phức tạp thuật toán là .

### Thuật toán khử Gauss.

+) Thuật toán biến đổi ma trận về ma trận bậc thang:

*Đầu vào*: Ma trận chưa được biến đổi.

**Bước 1**: .

**Bước 2:** Nếu hoặc thì kết thúc thuật toán.

**Bước 3:** Nếu , lặp :

* Nếu , thực hiện và quay lại **bước 3.**
* Nếu , dừng lặp.

**Bước 4:** Thực hiện đổi hai dòng và .

**Bước 5:** Lặp :

* Nếu : **,** tiến hành lặp :

**Bước 6:** Tiến hành, quay lại **bước 2**.

*Đầu ra*: Ma trận a đã được biến đổi.

+) Độ phức tạp thuật toán khử Gauss là .

### Thuật toán tìm hạng ma trận.

+) Thuật toán tìm hạng ma trận đã được biến đổi về dạng bậc thang:

*Đầu vào:* Ma trận đã được biến đổi.

**Bước 1:** .

**Bước 2:** Lặp :

* Lặp :
  + Nếu thì kết thúc lặp.
  + Tăng j thêm một đơn vị .
* Nếu , tăng thêm một đơn vị .

**Bước 3:** Trả về giá trị , kết thúc thuật toán.

*Đầu ra:* Số nguyên r là hạng của ma trận .

+) Độ phức tạp thuật toán tìm hạng ma trận là .

### Một số các hàm khác và phương pháp xử lí lỗi của hàm.

+) Đối với việc nhập ma trận thông qua hai hàm và :

1. Hàm thực hiện chức năng nhập ma trận từ bàn phím hoặc từ tệp đầu vào:
2. Dùng biến dummy để kiểm tra dữ liệu đầu vào có hợp lệ hay không.

# CHƯƠNG TRÌNH VÀ KẾT QUẢ

## Tổ chức chương trình

## Ngôn ngữ cài đặt

## Kết quả

### Giao diện chính của chương trình

### Kết quả thực thi của chương trình

Mô tả kết quả thực hiện chương trình.

### Nhận xét

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết quả đạt được

## Hạn chế

## Hướng phát triển

TÀI LIỆU THAM KHẢO