

**ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
**VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC**  
— o0o —



**BÁO CÁO CUỐI KỲ**

**MÔN HỌC: KỸ THUẬT LẬP TRÌNH**

**ĐỀ TÀI: GIẢI GẦN ĐÚNG HỆ PHƯƠNG  
TRÌNH TUYẾN TÍNH BẰNG PHƯƠNG  
PHÁP LẬP ĐƠN VÀ LẬP SEIDEL**

**Giảng viên hướng dẫn: TS.Nguyễn Thị Thanh Huyền**

**Sinh viên thực hiện: Vũ Văn Huy - 20216931**

**Lớp: Hệ thống thông tin quản lý - K66**

Hà Nội, 8/2023.

## Lời cảm ơn

Em xin gửi lời cảm ơn chân thành đến Cô TS.Nguyễn Thị Thanh Huyền vì sự hướng dẫn, hỗ trợ và cống hiến quý báu của Cô trong suốt môn học. Sự tận tâm và kiến thức chuyên sâu của Cô đã giúp định hình và phát triển khả năng lập trình của em một cách toàn diện. Nhờ sự hướng dẫn từ Cô mà em đã có thể áp dụng các kiến thức lập trình vào thực tế và hoàn thiện báo cáo của mình một cách tốt nhất.

Em cũng rất biết ơn vì cô đã tạo điều kiện cho em có một buổi thuyết trình để chia sẻ kiến thức và kinh nghiệm từ những hiểu biết của mình với các bạn trong lớp. Đó là một trải nghiệm thú vị và quý báu, giúp em cải thiện kỹ năng giao tiếp và trình bày, đồng thời ghi nhận được sự tiến bộ của em trong quá trình học tập.

Những kiến thức và kỹ năng em đã học được từ môn kĩ thuật lập trình sẽ luôn là tài sản quý giá và hỗ trợ không nhỏ cho sự phát triển của em trong tương lai. Em rất biết ơn vì đã có cơ hội được học tập và làm việc dưới sự hướng dẫn của Cô.

Một lần nữa, em xin chân thành cảm ơn Cô vì tất cả những điều tốt đẹp mà Cô đã mang đến cho em. Em sẽ luôn tự hào và biết ơn về mối liên kết giữa học trò và Cô trò trong suốt thời gian qua.

Xin chúc Cô sức khỏe dồi dào và tiếp tục thành công trong sự nghiệp giảng dạy cũng như công tác nghiên cứu.

**Trân trọng cảm ơn**

**Người báo cáo**

**Vũ Văn Huy - 20216931**

# Mục lục

<b>Mở đầu</b>	<b>5</b>
<b>1 Giới thiệu bài toán</b>	<b>7</b>
1.1 Hệ phương trình tuyến tính $AX = b$	7
1.2 Lập đơn(lập cổ điển)	7
1.3 Lập Seidel	8
<b>2 Hướng giải quyết bài toán</b>	<b>9</b>
<b>3 Thiết kế chương trình</b>	<b>12</b>
<b>4 Xây dựng chương trình</b>	<b>13</b>
4.1 Tạo menu	13
4.2 Nhập xuất ma trận	17
4.3 Kiểm tra chéo trội	19
4.4 Tính các chuẩn	20
4.5 Hàm tính toán	21
4.6 Sao chép ma trận	23
4.7 Lập đơn k lần	24
4.8 Lập đơn với công thức sai số, e cho trước	25
4.9 Lập đơn với điều kiện	26
4.10 Lập Seidel k lần	27
4.11 Lập Seidel với công thức sai số, e cho trước	29
4.12 Lập Seidel với điều kiện	30
<b>5 Mã nguồn</b>	<b>32</b>
<b>6 Kết quả khi chạy chương trình</b>	<b>55</b>
6.1 <b>Chạy menu</b>	55
6.2 Nhập input cho chương trình	56
6.3 Kiểm tra tính chéo trội và sự hội tụ của phương pháp	59

6.4	Tính chuẩn của các ma trận sau khi biến đổi . . . . .	60
6.5	Tính nghiệm gần đúng với số lần lặp k . . . . .	61
6.6	Tính nghiệm gần đúng với sai số e cho trước . . . . .	63
6.7	Tính nghiệm gần đúng thỏa mãn điều kiện . . . . .	65
6.8	Kết quả thu được trong file output.txt . . . . .	68
<b>7</b>	<b>Tổng kết</b>	<b>72</b>
7.1	Những điều đã học và làm được . . . . .	72
7.2	Đánh giá chi tiết . . . . .	72
7.3	Đánh giá chung . . . . .	73

## Mở đầu

Phương trình tuyến tính là một khái niệm quan trọng trong toán học và khoa học máy tính. Giải hệ phương trình tuyến tính đóng vai trò quan trọng trong nhiều lĩnh vực, bao gồm cả kỹ thuật, vật lý, kinh tế, và nhiều lĩnh vực khác. Trong báo cáo này, chúng ta sẽ tìm hiểu về hai phương pháp giải hệ phương trình tuyến tính: phương pháp lặp đơn và phương pháp lặp Seidel.

Bài báo cáo này sẽ không đi sâu vào việc chứng minh hội tụ của 2 phương pháp này mà chỉ quan tâm đến hướng đi, cách thức hoạt động của từng phương pháp. Báo cáo này không chỉ thể hiện sự nỗ lực và cam kết của em với việc hoàn thành môn học, mà còn là dịp để em chia sẻ những kiến thức hữu ích và kinh nghiệm thực tế trong lĩnh vực thuật toán.

Trên cơ sở đó báo cáo trình bày cách thức thực hiện phương pháp lặp đơn và phương pháp lặp Seidel để giải hệ phương trình tuyến tính. Đồng thời cung cấp các ví dụ minh họa và đánh giá tính hiệu quả của hai phương pháp này. Cuối cùng, chúng ta sẽ so sánh và rút ra những nhận xét về sự khác biệt và ưu điểm của mỗi phương pháp.

Báo cáo này sẽ bao gồm các phần chính sau:

- Trình bày bài toán
- Trình bày quá trình thiết kế chương trình theo phương pháp tính chỉnh dần từng bước
- In mã nguồn
- Chụp và in các hình ảnh giao diện thực hiện chương trình
- In nội dung file text là kết quả ra của chương trình
- Kết luận, đánh giá những kết quả đạt được, những ưu điểm và những thiếu sót (nếu có)

Gồm:

– Đánh giá chi tiết

- Đánh giá chung

## 1

## Giới thiệu bài toán

### 1.1 Hệ phương trình tuyến tính $AX = b$

Trong toán học, hệ phương trình tuyến tính  $AX = b$  là một khái niệm quan trọng trong đại số tuyến tính. Trong đó:  $A$  là ma trận hệ số,  $X$  là vector nghiệm cần tìm và  $b$  là vector kết quả.

Bắt đầu bằng cách nhập ma trận  $A$  và vectơ  $b$  theo định dạng ma trận. Sau đó, chúng ta sẽ kiểm tra tính chéo trội của ma trận  $A$  để đảm bảo tính hội tụ của phương pháp lặp đơn và lặp Seidel. Tính chéo trội là một thuộc tính quan trọng trong quá trình lặp, và nó đảm bảo rằng phương pháp sẽ hội tụ đến nghiệm mong muốn.

Mục tiêu của việc giải hệ phương trình tuyến tính là tìm các giá trị  $x_1, x_2, \dots, x_n$  sao cho thỏa mãn tất cả các phương trình trong hệ. Khi các giá trị này được tìm thấy, chúng ta sẽ có một vector  $X$  chứa giá trị của các biến chưa biết, giúp thỏa mãn hệ phương trình  $AX = b$ .

Có nhiều phương pháp để giải hệ phương trình tuyến tính, bao gồm phương pháp khử Gauss, Gauss - Jordan, lặp đơn, Seidel. Nhờ vào tính chất đặc biệt của các ma trận và vector, chúng ta có thể tìm ra giải pháp chính xác hoặc gần đúng cho hệ phương trình này.

### 1.2 Lặp đơn(lặp cổ điển)

Bắt đầu bằng cách nhập ma trận  $A$  và vectơ  $b$  theo định dạng ma trận. Sau đó, chúng ta sẽ kiểm tra tính chéo trội của ma trận  $A$  để đảm bảo tính hội tụ của phương pháp lặp đơn(tương tự với lặp Seidel). Tính chéo trội là một thuộc tính quan trọng trong quá trình lặp, và nó đảm bảo rằng phương pháp sẽ hội tụ đến nghiệm mong muốn. Quá trình lặp diễn ra liên tục cho đến khi thỏa mãn 1 điều kiện cho trước.

### 1.3 Lặp Seidel

Phương pháp lặp Seidel kế thừa từ phương pháp lặp cổ điển truyền thống bằng cách lặp để tính giá trị của  $x$  ở bên trái phương trình, bằng cách thế các giá trị  $x$  ở phép tính trước vào vế phải phương trình. Nhờ vậy mà nghiệm của phương trình qua các bước lặp sẽ hội tụ về nghiệm  $X^*$  nhanh chóng hơn.



## 2

## Hướng giải quyết bài toán

**1** Giả thiết ban đầu: Cho hệ phương trình tuyến tính  $AX = b$ , trong đó  $A$  là ma trận vuông cỡ  $n \times n$ ,  $b$  là vector có cùng kích thước  $n$  với ma trận  $A$ ,  $X$  là nghiệm gần đúng cần tìm.

**2** Các chuẩn, chéo hóa ma trận

1. Chuẩn của ma trận

- Cho vecto  $X = (X_1, X_2, \dots, X_n)^T$

$$\|X\|_1 = |X_1| + |X_2| + \dots + |X_n| : \text{chuẩn cột}$$

$$\|X\|_\infty = \max\{|X_1|, |X_2|, \dots, |X_n|\} : \text{chuẩn hàng}$$

Định nghĩa: Dãy vecto  $X^k$  hội tụ đến  $X^*$  nếu  $\lim_{k \rightarrow \infty} \|X^k - X^*\| = 0$  ( hội tụ theo chuẩn).

Một số chuẩn thông dụng:

$$\|A\|_1 = |a_{1j}| + |a_{2j}| + \dots + |a_{nj}| : \text{chuẩn cột}$$

$$\|A\|_\infty = \max\{|a_{i1}|, |a_{i2}|, \dots, |a_{in}|\} : \text{chuẩn hàng}$$

2. Tính chéo trội được định nghĩa như sau:

- Chéo trội hàng:

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}|$$

- Chéo trội cột:

$$|a_{jj}| > \sum_{i \neq j} |a_{ij}|$$

**3** Điều kiện hội tụ của phương pháp

Để hệ phương trình  $AX = b \Leftrightarrow X = BX + d$  hội tụ tới nghiệm  $X^*$  thì:

- $A$  là ma trận chéo trội.
- $\|B\|_\infty \leq q < 1$ .

**4** Phương pháp lặp đơn

- Biến đổi hệ về dạng  $X = BX + d$  với  $B$  là ma trận vuông cỡ  $n \times n$ ,  $d$  là vector hệ số tự do.

$$B_{ij} = \begin{cases} 0 & \text{khi } i = j \\ -\frac{a_{ij}}{a_{ii}} & \text{khi } i \neq j \end{cases}; d_i = \frac{b_i}{a_{ii}} \quad (1)$$

- Chọn  $X^{(0)}$  bất kỳ (sắp xỉ đầu). Để đơn giản bài toán và thuận tiện cho quá trình viết code ta chọn luôn sắp xỉ đầu là ma trận  $d$ .
- Xác định dãy  $X^{(k)}$

$$X^{k+1} = BX^k + d_i$$

- Đánh giá sai số:

$$\|X^k - X^*\| \leq \frac{q}{1-q} \|X^k - X^{k-1}\|$$

### 5 Phương pháp lặp Seidel

Tương tự như phương pháp lặp đơn ta cũng biến đổi hệ phương trình về dạng

$$X = BX + d \quad (*)$$

Trong đó:  $\|B\| = q < 1$

$$X = (B_1 + B_2)X + d = B_1X + B_2X + d \quad (**)$$

Trong đó:

- $B_1$  là ma trận tam giác trên
- $B_2$  là ma trận tam giác dưới
- $d$  là ma trận hằng số

$$B_{ij}^1 = \begin{cases} b_{ij} & , \text{nếu } i > j \\ 0 & , \text{nếu } i \leq j \end{cases}$$

$$B_{ij}^2 = \begin{cases} 0 & , \text{nếu } i > j \\ b_{ij} & , \text{nếu } i \leq j \end{cases}$$

Cho  $X^{(0)}$  bất kỳ. Tuy nhiên để đơn giản bài toán ta cũng chọn  $X^{(0)} = d$

$$X^{(k+1)} = B_1X^{(k+1)} + B_2X^{(k)} + d.$$

Dạng tọa độ:

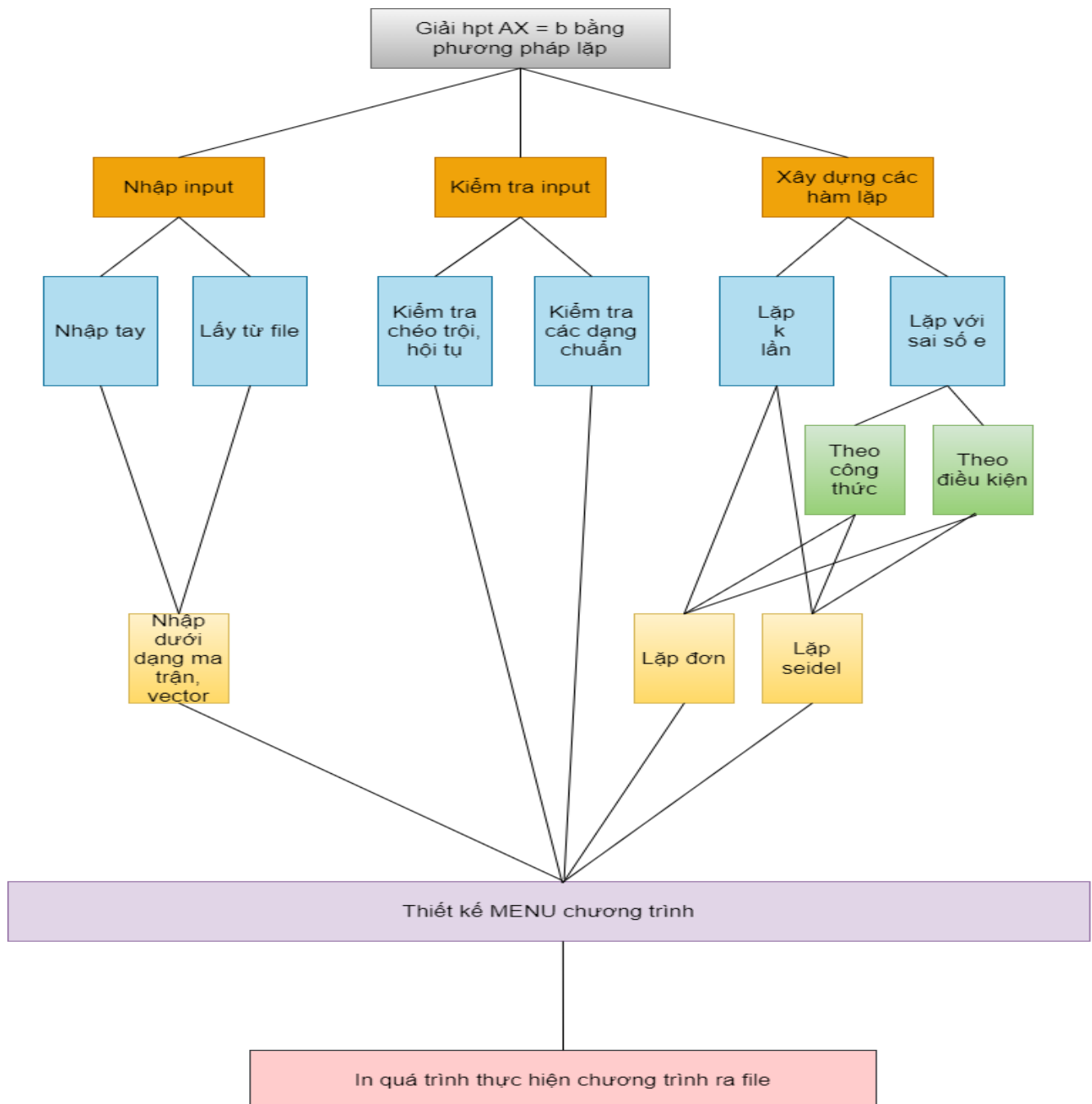
$$\begin{aligned}
 X_1^{k+1} &= b_{11}X_1^k + b_{12}X_2^k + \cdots + b_{1n}X_n^k + d_1 \\
 X_2^{k+1} &= b_{21}X_1^k + b_{22}X_2^k + \cdots + b_{2n}X_n^k + d_2 \\
 X_3^{k+1} &= b_{31}X_1^k + b_{32}X_2^k + \cdots + b_{3n}X_n^k + d_3 \\
 &\vdots \\
 &\vdots \\
 X_n^{k+1} &= b_{n1}X_1^k + b_{n2}X_2^k + \cdots + b_{nn}X_n^k + d_n
 \end{aligned}$$

Công thức xác định sai số tương tự lặp đơn:

$$||X^k - X^*|| \leq \frac{q}{1-q} ||X^k - X^{k-1}||$$

## 3

## Thiết kế chương trình



Hình 1: Sơ đồ thiết kế chương trình

## 4

## Xây dựng chương trình

Dưới đây là các hàm ứng với việc thực hiện các bước trong phương pháp tinh chỉnh từng bước được viết dưới dạng giả mã và cách thức hoạt động của từng hàm trong chương trình:

## 4.1 Tạo menu

**Function 1**

```
void gotoxy(int x, int y)
{
    COORD coord;
    coord.X = x;
    coord.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), coord);
}
```

**Ý tưởng:**

**gotoxy(x, y):** sử dụng để di chuyển con trỏ hiển thị trên giao diện dòng lệnh đến một vị trí cụ thể trên màn hình.

- Đầu vào: tọa độ x, y của vị trí con trỏ di chuyển đến
- Đầu ra: con trỏ hiển thị trên giao diện dòng lệnh sẽ được di chuyển đến vị trí tương ứng với tọa độ x và y.

**setTextColor(color):** Thiết lập màu sắc cho văn bản được in ra, truyền vào màu mong muốn theo mã màu.

**SetConsoleTextAttribute** để thiết lập màu sắc.

**GetStdHandle(STD\_OUTPUT\_HANDLE)** trả về handle cho đầu ra chuẩn của cửa sổ điều khiển (console).

```
void setTextColor(int color)
{
    SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), color);
}
```

**3. printHighlightedOption(const string& text):** in ra một tùy chọn văn bản được làm nổi bật bằng một màu sắc cụ thể vào chuỗi đại diện cho nội dung tùy chọn in ra

```
void printNormalOption(const string& text)
{
    cout << " " << text << " ";
}
```

**printNormalOption(const string& text):** in ra một tùy chọn văn bản bình thường mà không có màu sắc đặc biệt. Đầu vào là một chuỗi text đại diện cho nội dung tùy chọn cần in ra.

Hàm chỉ đơn giản in ra văn bản "+" text + "trên console mà không thay đổi màu sắc

```
void printHighlightedOption(const string& text)
{
    setTextColor(11);
    cout << " " << text << " ";
    setTextColor(15);
}
```

**setupMenu():** Tạo menu chính chứa các chức năng thực hiện bài toán

- Hàm nhận một tham chiếu ofstream có tên outputFile làm đối số. Tham chiếu này được sử dụng để ghi kết quả ra một tệp.
- Hàm khởi tạo một số biến để quản lý menu, bao gồm choice (tùy chọn menu hiện tại), maxChoice (tùy chọn menu tối đa), và key (để lưu giá trị đầu vào từ người dùng).
- Vòng lặp vô hạn đến khi chọn thoát. Trước tiên xóa màn hình.
- Hàm in tiêu đề menu với tiêu đề và các tùy chọn menu khác nhau. Tùy chọn đang được chọn sẽ được làm nổi bật bằng ký hiệu "»".
- Đầu vào từ người dùng được đọc bằng cách sử dụng hàm getch(), mà chờ đợi một lần nhấn phím mà không cần nhấn Enter.
- **Cài đặt phím:** 'w' hoặc 'W' hoặc phím mũi tên lên (ASCII 72): Giảm giá trị choice nếu nó lớn hơn 0 (di chuyển lên trên menu).  
's' hoặc 'S' hoặc phím mũi tên xuống (ASCII 80): Tăng giá trị choice nếu nó nhỏ hơn maxChoice.  
'\_r' (ASCII của Enter): Thực hiện chức năng được chọn. Khi người dùng nhấn Enter, hàm kiểm tra giá trị choice để xác định tùy chọn đã chọn và sau đó thực hiện các hành động cụ thể dựa trên tùy chọn đã chọn.
- Trong khối switch-case để xử lý tùy chọn đã chọn, mỗi case tương ứng với một tùy chọn menu và nó chứa các chức năng liên quan đến tùy chọn đó.
- Hàm chờ người dùng nhấn phím bất kỳ để tiếp tục (bằng cách gọi hàm getch()) trước khi quay lại menu.
- Vòng lặp tiếp tục hiển thị menu cho đến khi người dùng chọn tùy chọn **"Thoat"** (khi choice bằng maxChoice).

```
void setupMenu(ofstream &outputFile)
{
    while (true) {
        clearScreen();
```

```
TaoHienThi;
key = getch();
DieuChinhLenXuong;
    XoaManHinh;
    if (choice == maxChoice)
        print("Thoat chuong trinh.");
        exit(0);
    else {
        print("Lua chon: ")
        switch (choice)
        {
            case 0:
                NhapMaTran;
            case 1:
                KiemtraCheoTroi;
            case 2:
                TinhCacChuan();
            case 3:
                if chuanHang(D, size)>=1 then:
                    print("He phuong trinh ko giai duoc")
                else chucnang4(outputFile, D, E, size);
            case 4:
                if chuanHang(D, size)>= 1 then:
                    print("He da cho khong giai duoc bang pp
lap don, seidel");
                else
                    chucnang5(outputFile, D, E, size);
            case 5:
                if chuanHang(D, size)>=1 then:
                    print("He da cho khong giai duoc bang lap
don, lap se}else:")
                    chucnang6(outputFile,D, E, size);
```



```

    }
    ThoatMenu;
}
}
Ghi cac thao tac vao file;
}

```

Các menu con tương ứng với các chức năng khác tương tự với menu lớn trên đây.

## 4.2 Nhập xuất ma trận

### Function 2



**Cách 1:** Đọc dữ liệu từ 1 file txt (sắp xếp có thứ tự, các phần tử cách nhau bởi phím space hoặc tab).

- Đầu vào: file text, kích thước ma trận (n), và 2 ma trận cần đọc lấy dữ liệu
- `ifstream file(filename.c_str());` ifstream là lớp để đọc tệp.
- `file » *n`: Hàm đọc một giá trị nguyên từ tệp và lưu trữ vào biến con trỏ n.
- 2 vòng lặp sau lần lượt duyệt từng phần tử của file, đọc nó ra và cũng đồng thời in các phần tử đã đọc vào filename
- `file.close();` Sau khi đọc xong dữ liệu, đóng tệp và giải phóng tài nguyên.

```

void readfromFile(const string& filename, int* n, double A[100][100],
double B[100][100])
{
    ifstream file(filename.c_str());
    if (file.is_open())
    {

```

```
file >> *n;
for (int i = 0; i < *n; i++) {
    for (int j = 0; j < *n; j++) {
        file >> A[i][j];
        cout<<A[i][j] << " ";
    }
    cout<<endl;
}
for (int i = 0; i < *n; i++) {
    file >> B[i][0];
    cout<<B[i][0]<<endl;
}
file.close();
}else{
    cout << "Khong the mo tep!" << std::endl;
}
}
```

**Cách 2:** Nhập trực tiếp từ bàn phím, do truyền vào mảng 2 chiều, số dòng, cột, nên cần 2 vòng for để nhập vào mảng 2 chiều.

```
void NhapMaTran(double a[100][100], int dong, int cot)
{
    for i = 0 to dong - 1:
        for j = 0 to cot - 1:
            cin >> a[i][j];
}
```

**XuatMaTran:** in ma trận đã nhập và ghi ma trận vào file text bằng cách duyệt lần lượt các phần tử của mảng 2 chiều

```
void XuatMaTran(double a[100][100], int dong, int cot, ofstream &outputFile)
{
    for i = 0 to dong - 1:
        for j = 0 to cot - 1:
            outputFile << a[i][j] << " ";
            cout<<a[i][j] << " ";
        outputFile << "\n";
        cout << "\n";
}
```

### 4.3 Kiểm tra chéo trội

#### Function 3



**cheoTroisHang():** Chỉ cần kiểm tra chéo trội hàng hoặc cột là được.

- Sử dụng 2 vòng lặp, lặp qua từng hàng, tính tổng các phần tử không nằm trên đường chéo chính.
- So sánh phần tử trên đường chéo chính với tổng vừa tính, nếu nhỏ hơn thì trả về False
- Kết thúc hàm trả về True

```
bool cheoTroisHang(a[100][100], size)
{
    sum_row := 0;
    for i = 0 to size - 1:
        sum_row := 0;
        for j = 0 to size - 1:
            if j != i then:
```

```
        sum_row += fabs(a[i][j]);
    if sum_row >= fabs(a[i][i]) then:
        return false;
    return true;
}
```

## 4.4 Tính các chuẩn

**Function 4**

```
double chuanHang(A[100][100], size)
{
    maxSum := 0;
    for i = 0 to size - 1:
        rowSum := 0;
        for j = 0 to size - 1:
            rowSum += fabs(A[i][j]);
        if rowSum > maxSum then:
            maxSum := rowSum;
    return maxSum;
}
```

```
double chuanCot(A[100][100], size)
{
    maxSum := 0;
    for i = 0 to size - 1:
        colSum := 0;
```

```
    for j = 0 to size - 1:
        colSum += fabs(A[j][i]);
    if colSum > maxSum then:
        maxSum := colSum;
return maxSum;
}
```

## 4.5 Hàm tính toán

**Function** 5

**congMatrix:** cộng từng phần tử tương ứng của hai mảng vào gán nó vào 1 mảng dùng để lưu trữ thứ 3.

**Input:** 2 ma trận cần cộng.

**Output:** Ma trận chứa kết quả.

```
double congMatrix(A[100][100], B[100][100], C[100][100], size)
{
    for i = 0 to size - 1:
        for j = 0 to size - 1:
            C[i][j] := B[i][j] + A[i][j];
    return C[100][100];
}
```

**Tương tự đối với hàm nhanMatrix:** Hàm này sử dụng ba vòng lặp for lồng nhau để tính giá trị của từng phần tử trong ma trận kết quả C, thực hiện phép nhân hai phần tử tương ứng từ ma trận A và B, và cộng kết quả vào ma trận C.

```
double nhanMatrix(A[100][100], B[100][100], C[100][100], size)
{
    for i = 0 to size - 1:
        for j = 0 to size - 1:
            C[i][j] := 0;
            for k = 0 to size - 1:
                C[i][j] += A[i][k] * B[k][j];
    return C[100][100];
}
```

**Function 6**

**timLamda:** thực chất hàm này là hàm để tìm ra ma trận B trong biến đổi ( $X = BX + d$ )

**Input:** Ma trận A ban đầu, ma trận C để lưu, kích thước ma trận theo công thức (1) thì các phần tử trên đường chéo chính của B đều bằng 0, các phần tử còn lại chia cho trái dấu phần tử trên đường chéo chính

**Output:** Ma trận B theo công thức.

```
double timLamda(a[100][100], c[100][100], size)
{
    for i = 0 to size - 1:
        for j = 0 to size - 1:
            if i == j then:
                c[i][j] := 0;
            else:
                c[i][j] := - a[i][j]/a[i][i];
}
```

**timvecto\_d():** Tương tự như hàm timLamda hàm này để tìm vecto d trong biến đổi

trên tuy nhiên phần tử trên đường chéo chính cột đầu tiên khác khác 0.

- Đầu vào: ma trận A, vecto d và 1 ma trận c để lưu trữ kết quả.
- Cũng áp dụng công thức (1).

```
double timvecto_d(a[100][100], b[100][100], c[100][100])
{
    for i = 0 to len(b) - 1:
        for j = 0 to len(b) - 1:
            c[i][j] := b[i][j] / a[i][i];
    return c[100][100];
}
```

## 4.6 Sao chép ma trận

**copy():** duyệt mảng 2 chiều và lần lượt gán các phần tử của ma trận a cho ma trận b.

**Function 7**



```
void copy(a[100][100], b[100][100], size)
{
    for i = 0 to size - 1:
        for j = 0 to size - 1:
            b[i][j] := a[i][j];
}
```

## 4.7 Lập đơn k lần

Tất cả các hàm làm việc với các phương pháp lập đơn và lập Seidel ta đều chọn xấp xỉ đầu là ma trận  $d$  thay vì nhập từ bàn phím để tối ưu chương trình đồng nhất và dễ dàng so sánh kết quả giữa 2 phương pháp

**Function 8** ★★ 

**loop\_KDon():** sử dụng 1 ma trận  $eva$  để làm ma trận đánh giá sai số,  $K$  là ma trận sao chép từ  $E$ ,  $F$  là ma trận dùng để tính toán trong vòng lặp.

**Trong vòng lặp while:**

- Nhân ma trận  $D$  và  $E$ , kết quả được lưu vào ma trận  $F$ .
- Cộng ma trận  $F$  và  $K$ , kết quả được lưu lại vào ma trận  $F$ .
- In ra ma trận  $F$  và ghi kết quả này vào file `outputFile`.
- Tính toán giá trị  $eva$  bằng cách trừ các phần tử tương ứng của  $E$  và  $F$  và lưu vào ma trận  $eva$ .
- Gán giá trị của  $F$  vào  $E$  để chuẩn bị cho lần lặp tiếp theo

**Sau khi kết thúc vòng lặp, tính toán và in ra sai số:**

Tính giá trị của sai số theo công thức:  $\left(\frac{q}{1-q}\right) \cdot (X_k - X_{k-1})$ , trong đó  $q$  là chuẩn hàng của ma trận  $D$ ,  $X_k$  là ma trận  $F$  sau khi lặp  $k$ , và  $X_{k-1}$  là ma trận  $F$  sau khi lặp  $k-1$ .

Kết quả được in ra màn hình và ghi vào file `outputFile`.

**Input:** Hai ma trận trong biến đổi  $B, d$  ( $X = BX + d$ ), kích thước ma trận, số lần lặp( $k$ ), một tham chiếu đến đối tượng `ofstream` cho phép ghi kết quả vào một file nào đó.

**Ouput:** Ma trận  $X$  sau khi lặp đơn qua  $k$  lần, đánh giá sai số.



```

void loop_KDon(D[100][100], E[100][100], size, k, ofstream &outputFile)
{
    double eva[100][100];
    double K[100][100];
    copy(E, K, size);
    double F[100][100];
    int dem := 1;
    while dem <=k then:
        nhanMatrix(D, E, F, size);
        congMatrix(F, K, F, size);
        XuatMaTran(F, size, 1, outputFile);
        for i = 0 to size -1 :
            eva[i][0] = E[i][0] - F[i][0];
        copy(F, E, size);
        dem++;
    Danh gia sai so;
    print(chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size));
    Ghi tat ca cac thao tac vao file;
}

```

#### 4.8 Lập đơn với công thức sai số, e cho trước

##### Function 9



**loop\_eDon():** Tương tự hàm loop\_KDon() chỉ khác ở chỗ sai số không được in ra mà được đánh giá trong từng vòng lặp. Vòng lặp vô hạn cho đến khi sai số bé hơn sai số e nhập vào ban đầu

```

void loop_KDon(D[100][100], E[100][100], size, e, ofstream &outputFile)
{

```

```

double eva[100][100];
double K[100][100];
copy(E, K, size);
double F[100][100];
int dem := 1;
while 1:
    nhanMatrix(D, E, F, size);
    congMatrix(F, K, F, size);
    XuatMaTran(F, size, 1, outputFile);
    for i = 0 to size -1 :
        eva[i][0] = E[i][0] - F[i][0];
        copy(F, E, size);
        if (chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size))
< e then break;
        dem++;
    Ghi tat ca cac thao tac vao file;
}

```

## 4.9 Lặp đơn với điều kiện

### Function 10



**loop\_Don\_with\_condtition()**: tương tự hàm loop\_eDon khác ở công thức sai số:

$$|X_{k+1} - X_k| \leq e.$$

```

void loop_Don_with_condition(D[100][100], E[100][100], size, e, ofstream
&outputFile)
{
    double eva[100][100];
    double K[100][100];

```

```

copy(E, K, size);
double F[100][100];
int dem := 1;
while 1:
    nhanMatrix(D, E, F, size);
    congMatrix(F, K, F, size);
    XuatMaTran(F, size, 1, outputFile);
    for i = 0 to size -1 :
        eva[i][0] = E[i][0] - F[i][0];
    copy(F, E, size);
    if chuanHang(eva, size) < e then: break;
    dem++;
Ghi tat ca cac thao tac vao file;
}

```

#### 4.10 Lặp Seidel k lần

##### Function 11



**loop\_kSeidel()**: không giống như lặp đơn giải bằng các lặp tuần tự theo công thức nghiệp. Với lặp Seidel cải tiến hơn bằng cách tính được biến nào thì dùng để tính biến mới.

1. Khởi tạo biến  $dem = 1$  để đếm số lần lặp
2. Khởi tạo hai ma trận **eva** để đánh giá sai số và **y** để lưu trữ các nghiệm trong quá trình lặp
3. Trong mỗi lần lặp while:
  - Tính giá trị mới cho nghiệm  $y[i][0]$  của biến thứ  $i$  bằng cách tính tổng các phần

tử trong hàng tương ứng của ma trận  $D$  nhân với các giá trị  $y[j][0]$  của nghiệm ở các lần lặp trước đó, trừ đi giá trị tại vị trí  $i$  của ma trận  $E$ .

- Ghi đè nghiệm mới vào  $y[i][0]$  để chuẩn bị cho lần lặp tiếp theo.
- Kiểm tra nếu dem bằng  $k-1$ , tức là lần lặp gần cuối, thì sao chép nghiệm hiện tại vào  $eva$  để lưu trữ giá trị cho việc tính sai số sau này.
- In ra nghiệm của từng biến sau mỗi lần lặp.

4. Sau khi kết thúc vòng lặp, tính và in ra sai số cuối cùng

5. Để tính giá trị của sai số theo công thức:

$$\left( \frac{q}{1-q} \right) \cdot (X_k - X_{k-1})$$

trong đó  $q$  là chuẩn hàng của ma trận  $D$ ,  $X_k$  là ma trận  $y$  sau khi lặp  $k$ , và  $X_{k-1}$  là ma trận **eva** lưu giá trị của  $y$  sau khi lặp  $k-1$ .

6. Kết quả sẽ được in ra màn hình và ghi vào file *outputFile*.

**Input:** Hai ma trận trong biến đổi  $B$ ,  $d$  ( $X = BX + d$ ), kích thước ma trận, số lần lặp( $k$ ), một tham chiếu đến đối tượng ofstream cho phép ghi kết quả vào một file nào đó.

**Ouput:** Ma trận  $X$  sau khi lặp Seidel qua  $k$  lần, đánh giá sai số.

```
void loop_kSeidel(D[100][100], E[100][100], size, k, ofstream &outputFile)
{
    int dem := 1;
    double eva[100][100];
    double y[100][100];
    copy(E, y, size);
    while dem <= k {
        for i = 0 to size -1:
            double sum := 0.0;
            for j = 0 to size -1:
                if j != i then:
                    sum += D[i][j] * y[j][0];
```

```

        y[i][0] := E[i][0] + sum;
    if dem == k- 1 then:
        copy(y, eva, size);
        dem++;
    }
    for i = 0 to size -1:
        eva[i][0] = y[i][0] - eva[i][0];
    Danh gia sai so;
    print(chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size));
    Ghi cac thao tac vao file;
}

```

#### 4.11 Lặp Seidel với công thức sai số, e cho trước

##### Function 12



**loop\_eSeidel():** Tương tự loop\_KSeidel nhưng điều kiện dừng của vòng while là sai số nhỏ hơn sai số nhập vào.

```

void loop_eSeidel(D[100][100], E[100][100], size, e, ofstream &outputFile)
{
    int dem = 1;
    double eva[100][100];
    double y[100][100];
    copy(E, y, size);
    while (1) {
        copy(y, eva, size);
        for i = 0 to size -1:
            double sum := 0.0;
            for j = 0 to size -1:

```

```

        if j != i then:
            sum += D[i][j] * y[j][0];
        y[i][0] = E[i][0] + sum;
    for i = 0 to size - 1:
        eva[i][0] = y[i][0] - eva[i][0];

    if chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size)
< e break;
    dem++;
}
Ghi cac thao tac vao file;
}

```

## 4.12 Lập Seidel với điều kiện

`loop_Seidel_with_condition()`: tương tự hàm `loop_eSeidel` nhưng thay công thức sai số thành  $|X_{k+1} - X_k| \leq e$ .

**Function 13**



```

void loop_Seidel_with_condition(D[100][100], E[100][100], size, e, ofstream
&outputFile)
{
    int dem := 1;
    double eva[100][100];
    double y[100][100];
    copy(E, y, size);
    while (1) {
        copy(y, eva, size);
        for i = 0 to size - 1:
            double sum = 0.0;

```

```
    for j = 0 to size - 1:
        if j != i then:
            sum += D[i][j] * y[j][0];
        y[i][0] := E[i][0] + sum;
    for i = 0 to size - 1:
        eva[i][0] = y[i][0] - eva[i][0];
        if chuanHang(eva, size) < e then: break;
    dem++;
}
Ghi cac thao tac vao file;
}
```

## 5

## Mã nguồn

```
1 #include <iostream>
2 #include <conio.h>
3 #include <windows.h>
4 #include <math.h>
5 #include <fstream>
6
7 using namespace std;
8
9 void clearScreen()
10 {
11     system("cls");
12 }
13 void gotoxy(int x, int y)
14 {
15     COORD coord;
16     coord.X = x;
17     coord.Y = y;
18     SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), coord);
19 }
20 void setTextColor(int color)
21 {
22     SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE), color);
23 }
24 void printHighlightedOption(const string& text)
25 {
26     setTextColor(11);
27     cout << " " << text << " ";
28     setTextColor(15);
29 }
30 void printNormalOption(const string& text)
31 {
32     cout << " " << text << " ";
33 }
34
35 void NhapMaTran(double a[100][100], int dong, int cot)
36 {
37     for (int i = 0; i < dong; i++)
38     {
39         for(int j = 0; j < cot; j++)
40         {
41             cin >> a[i][j];
42         }
43     }
44 }
45
46 void XuatMaTran(double a[100][100], int dong, int cot, ofstream &outputFile)
47 {
48     for (int i = 0; i < dong; i++)
49     {
50         for (int j = 0; j < cot; j++)
51         {
52             outputFile << a[i][j] << " ";
53             cout<<a[i][j] << " ";
54         }
```



```
55     outputFile << endl;
56     cout << endl;
57 }
58 }
59
60 void readfromFile(const string& filename, int* n, double A[100][100], double
    B[100][100])
61 {
62     ifstream file(filename.c_str());
63     if (file.is_open()) {
64         file >> *n;
65         for (int i = 0; i < *n; i++) {
66             for (int j = 0; j < *n; j++)
67             {
68                 file >> A[i][j];
69                 cout<<A[i][j] << " ";
70             }
71             cout<<endl;
72         }
73         for (int i = 0; i < *n; i++)
74         {
75             file >> B[i][0];
76             cout<<B[i][0]<<endl;
77         }
78         file.close();
79     } else {
80         cout << "Khong the mo tep!" << endl;
81     }
82 }
83
84 bool cheoTroiHang(double a[100][100], int size)
85 {
86     double sum_row = 0;
87     for (int i = 0; i < size; i++)
88     {
89         sum_row = 0;
90         for (int j = 0; j < size; j++)
91         {
92             if (j != i)
93             {
94                 sum_row += fabs(a[i][j]);
95             }
96         }
97         if (sum_row >= fabs(a[i][i]))
98         {
99             return false;
100         }
101     }
102     return true;
103 }
104
105 double chuanHang(double A[100][100], int size)
106 {
107     double maxSum = 0.0;
108
109     for(int i = 0; i < size; i++)
110     {
111         double rowSum = 0.0;
```

```
112         for(int j = 0; j < size; j++)
113         {
114             rowSum += fabs(A[i][j]);
115         }
116
117         if(rowSum > maxSum)
118         {
119             maxSum = rowSum;
120         }
121     }
122     return maxSum;
123 }
124
125
126 double chuanCot(double A[100][100], int size)
127 {
128     double maxSum = 0.0;
129     for(int i = 0; i < size; i++)
130     {
131         double colSum = 0.0;
132
133         for(int j = 0; j < size; j++)
134         {
135             colSum += fabs(A[j][i]);
136         }
137
138         if(colSum > maxSum)
139         {
140             maxSum = colSum;
141         }
142     }
143     return maxSum;
144 }
145
146 double congMatrix(double A[100][100], double B[100][100], double C
147 [100][100], int size)
148 {
149     for(int i = 0; i < size; i++)
150     {
151         for(int j = 0; j < size; j++)
152         {
153             C[i][j] = B[i][j] + A[i][j];
154         }
155     }
156     return C[100][100];
157 }
158
159 double nhanMatrix(double A[100][100], double B[100][100], double C
160 [100][100], int size)
161 {
162     for(int i = 0; i < size; i++)
163     {
164         for(int j = 0; j < sizeof(B); j++)
165         {
166             C[i][j] = 0;
167             for(int k = 0; k < size; k++)
168             {
169                 C[i][j] += A[i][k] * B[k][j];
170             }
171         }
172     }
173 }
```

```

168     }
169 }
170 }
171 return C[100][100];
172 }
173
174 double timLamda(double a[100][100], double c[100][100], int size)
175 {
176     for(int i = 0; i < size; i++)
177     {
178         for(int j = 0; j < size; j++)
179         {
180             if(i == j)
181             {
182                 c[i][j] = 0;
183             }
184             else
185             {
186                 c[i][j] = - a[i][j]/a[i][i];
187             }
188         }
189     }
190 }
191
192 double timvecto_d(double a[100][100], double b[100][100], double c[100][100])
193 {
194     for(int i = 0; i < sizeof(b); i++)
195     {
196         for(int j = 0; j < sizeof(b); j++)
197         {
198             c[i][j] = b[i][j]/a[i][i];
199         }
200     }
201     return c[100][100];
202 }
203
204 void copy(double a[100][100], double b[100][100], int size)
205 {
206     for(int i = 0; i < size; i++)
207     {
208         for(int j = 0; j < size; j++)
209         {
210             b[i][j] = a[i][j];
211         }
212     }
213 }
214
215 void loop_KDon(double D[100][100], double E[100][100], int size, int k,
    ofstream &outputFile)
216 {
217     double eva[100][100];
218     cout<<"Nhap xap xi dau: x_0: "<<endl;
219     outputFile<<"Nhap xap xi dau: x_0: "<<endl;
220     cout<<"Chon luon xap xi dau la ma tran d."<<endl;
221     outputFile<<"Chon luon xap xi dau la ma tran d."<<endl;
222     double K[100][100];
223     copy(E, K, size);
224     double F[100][100];

```

```

225     int dem = 1;
226
227     while(dem <=k)
228     {
229         cout<<"Lan " <<dem<<endl;
230         outputFile<<"Lan " <<dem<<endl;
231
232         nhanMatrix(D, E, F, size);
233         congMatrix(F, K, F, size);
234         XuatMaTran(F, size, 1, outputFile);
235
236         for(int i = 0; i < size; i++)
237         {
238             eva[i][0] = E[i][0] - F[i][0];
239         }
240
241         copy(F, E, size);
242         dem++;
243     }
244
245     cout<<endl<<"Danh gia sai so: " <<endl;
246     outputFile<<endl<<"Danh gia sai so: " <<endl;
247     cout<<"sai so: (q/(1-q))" <<"(X_" <<k<<" - " <<"X_" <<k-1<<") = " <<chuanHang
(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size);
248     outputFile<<"sai so: (q/(1-q))" <<"(X_" <<k<<" - " <<"X_" <<k-1<<") = " <<
chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size)<<endl;
249 }
250
251 void loop_eDon(double D[100][100], double E[100][100], int size, double e,
ofstream &outputFile)
252 {
253     double eva[100][100];
254     double K[100][100];
255     copy(E, K, size);
256     double F[100][100];
257
258     int dem = 1;
259     while(1)
260     {
261         cout<<"Lan " <<dem<<endl;
262         outputFile<<"Lan " <<dem<<endl;
263
264         nhanMatrix(D, E, F, size);
265         congMatrix(F, K, F, size);
266         XuatMaTran(F, size, 1, outputFile);
267
268         for(int i = 0; i < size; i++)
269         {
270             eva[i][0] = E[i][0] - F[i][0];
271         }
272
273         copy(F, E, size);
274
275         if((chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size))
< e) break;
276         dem++;
277     }
278     cout << "Da giai xong he phuong trinh trong " << dem << " buoc lap." <<
endl;

```

```
278     outputFile << "Da giai xong he phuong trinh trong " << dem << " buoc lap
    ." << endl;
279 }
280
281 void loop_Don_with_condition(double D[100][100], double E[100][100], int
    size, double e, ofstream &outputFile)
282 {
283     double eva[100][100];
284     double K[100][100];
285     copy(E, K, size);
286     double F[100][100];
287     int dem = 1;
288     while(1)
289     {
290         cout<<"Lan " << dem << endl;
291         outputFile<<"Lan " << dem << endl;
292
293         nhanMatrix(D, E, F, size);
294         congMatrix(F, K, F, size);
295         XuatMaTran(F, size, 1, outputFile);
296
297         for(int i = 0; i < size; i++)
298         {
299             eva[i][0] = E[i][0] - F[i][0];
300         }
301
302         copy(F, E, size);
303
304         if(chuanHang(eva, size) < e) break;
305         dem++;
306     }
307     cout << "Da giai xong he phuong trinh trong " << dem << " buoc lap." <<
    endl;
308     outputFile << "Da giai xong he phuong trinh trong " << dem << " buoc lap
    ." << endl;
309 }
310
311 void loop_kSeidel(double D[100][100], double E[100][100], int size, int k,
    ofstream &outputFile)
312 {
313     int dem = 1;
314     double eva[100][100];
315     double y[100][100];
316     copy(E, y, size);
317     cout << "Nghiem cua he phuong trinh la: " << endl;
318     outputFile << "Nghiem cua he phuong trinh la: " << endl;
319     while (dem <= k) {
320         for (int i = 0; i < size; i++)
321         {
322             double sum = 0.0;
323             for (int j = 0; j < size; j++)
324             {
325                 if (j != i)
326                 {
327                     sum += D[i][j] * y[j][0];
328                 }
329             }
330             y[i][0] = E[i][0] + sum;
```

```

331     }
332     if(dem == k- 1 )
333     {
334         copy(y, eva, size);
335     }
336     cout<<"Lan"<<dem<<endl;
337     for (int i = 0; i < size; i++)
338     {
339         cout << "x[" << i+1 << "]" = " << y[i][0] << endl;
340         outputFile << "x[" << i +1<< "]" = " << y[i][0] << endl;
341     }
342     dem++;
343 }
344
345 for(int i = 0; i < size; i++)
346 {
347     eva[i][0] = y[i][0] - eva[i][0];
348 }
349
350 cout<<endl<<"Danh gia sai so: " <<endl;
351 outputFile<<endl<<"Danh gia sai so: " <<endl;
352 cout<<"Sai so: (q/(1-q))"<<"(X_"<<dem-1<<" - "<<"X_"<<dem-2<<" ) = "<<
chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size)<<endl;
353 outputFile<<"Sai so: (q/(1-q))"<<"(X_"<<dem-1<<" - "<<"X_"<<dem-2<<" ) =
"<<chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size)<<endl;
354 }
355
356 void loop_eSeidel(double D[100][100], double E[100][100], int size, double e
, ofstream &outputFile)
357 {
358     int dem = 1;
359     double eva[100][100];
360     double y[100][100];
361     copy(E, y, size);
362     cout << "Nghiem cua he phuong trinh la: " << endl;
363     outputFile << "Nghiem cua he phuong trinh la: " << endl;
364     while (1) {
365         copy(y, eva, size);
366         for (int i = 0; i < size; i++)
367         {
368             double sum = 0.0;
369             for (int j = 0; j < size; j++)
370             {
371                 if (j != i)
372                 {
373                     sum += D[i][j] * y[j][0];
374                 }
375             }
376             y[i][0] = E[i][0] + sum;
377         }
378         cout<<"Lan"<<dem<<endl;
379         for (int i = 0; i < size; i++)
380         {
381             cout << "x[" << i+1 << "]" = " << y[i][0] << endl;
382             outputFile << "x[" << i+1 << "]" = " << y[i][0] << endl;
383         }
384
385         for(int i = 0; i < size; i++)

```

```

386         {
387             eva[i][0] = y[i][0] - eva[i][0];
388         }
389
390         if(chuanHang(D, size)/(1-chuanHang(D, size))*chuanHang(eva, size) <
e) break;
391         dem++;
392     }
393     cout << "Da giai xong he phuong trinh trong " << dem << " buoc lap." <<
endl;
394     outputFile << "Da giai xong he phuong trinh trong " << dem << " buoc lap
." << endl;
395 }
396
397 void loop_Seidel_with_condition(double D[100][100], double E[100][100], int
size, double e, ofstream &outputFile)
398 {
399     int dem = 1;
400     double eva[100][100];
401     double y[100][100];
402     copy(E, y, size);
403     cout << "Nghiem cua he phuong trinh la: " << endl;
404     outputFile << "Nghiem cua he phuong trinh la: " << endl;
405     while (1) {
406         copy(y, eva, size);
407         for (int i = 0; i < size; i++)
408         {
409             double sum = 0.0;
410             for (int j = 0; j < size; j++)
411             {
412                 if (j != i)
413                 {
414                     sum += D[i][j] * y[j][0];
415                 }
416             }
417             y[i][0] = E[i][0] + sum;
418         }
419         cout<<"Lan"<<dem<<endl;
420         for (int i = 0; i < size; i++)
421         {
422             cout << "x[" << i+1 << "] = " << y[i][0] << endl;
423             outputFile << "x[" << i + 1<< "] = " << y[i][0] << endl;
424         }
425
426         for(int i = 0; i < size; i++)
427         {
428             eva[i][0] = y[i][0] - eva[i][0];
429         }
430
431         if(chuanHang(eva, size) < e) break;
432         dem++;
433     }
434     cout << "Da giai xong he phuong trinh trong " << dem << " buoc lap." <<
endl;
435     outputFile << "Da giai xong he phuong trinh trong " << dem << " buoc lap
." << endl;
436 }
437

```

```

438 void chucnang1(ofstream &outputFile, double A[100][100], double b[100][100],
439               int size)
439 {
440     int choice = 0;
441     int maxChoice = 2;
442     char key;
443     while (true) {
444         clearScreen();
445         cout << "=====
446         " << endl;
447         cout << " |                               Nhap input cho
chuong trinh | " << endl;
448         cout << "=====
449         " << endl;
450
451         for (int i = 0; i <= maxChoice; i++) {
452             gotoxy(0, i + 3);
453             if (i == choice) {
454                 printHighlightedOption(">>");
455                 switch (i) {
456                     case 0:
457                         cout << " |";
458                         printHighlightedOption(" 1. Nhap tu file");
459                         cout << "
460                         |";
461                         break;
462                     case 1:
463                         cout << " |";
464                         printHighlightedOption(" 2. Nhap tay ");
465                         cout << "
466                         |";
467                         break;
468                     case 2:
469                         cout << " |";
470                         printHighlightedOption(" Thoat");
471                         cout << "
472                         |";
473                         break;
474                 }
475             }
476             else {
477                 printNormalOption(" ");
478                 switch (i) {
479                     case 0:
480                         cout << " |";
481                         printNormalOption(" 1. Nhap tu file");
482                         cout << "
483                         |";
484                         break;
485                     case 1:
486                         cout << " |";
487                         printNormalOption(" 2. Nhap tay ");
488                         cout << "
489                         |";
490                         break;
491                     case 2:
492                         cout << " |";
493                         printNormalOption(" Thoat");
494                 }
495             }
496         }
497     }
498 }

```



```

487         cout << "
488             |";
489         break;
490     }
491 }
492 cout << endl;
493 cout << "=====
494 " << endl;
495 key = getch();
496
497 switch (key) {
498     case 'w':
499     case 'W':
500     case 72:
501         if (choice > 0)
502         {
503             choice--;
504             gotoxy(0, choice + 4);
505             printHighlightedOption(">>");
506             gotoxy(0, choice + 5);
507             printNormalOption(" ");
508         }
509         break;
510     case 's':
511     case 'S':
512     case 80:
513         if (choice < maxChoice)
514         {
515             choice++;
516             gotoxy(0, choice + 2);
517             printHighlightedOption(">>");
518             gotoxy(0, choice + 1);
519             printNormalOption(" ");
520         }
521         break;
522     case '\r':
523         clearScreen();
524         if (choice == maxChoice) {
525             return;
526         }
527         else {
528             cout << "Ban da lua chon: ";
529             cout << endl;
530             outputFile << "Giai HPT tuyen tinh AX = b bang lap don, lap
Seidel" << endl;
531             outputFile<<"Lua chon 1:"<<endl;
532             switch (choice) {
533                 case 0:
534                     setTextColor(10);
535                     cout << "Nhap tu file" << endl;
536                     outputFile << "Nhap tu file" << endl;
537                     readfromFile("input.txt", &size, A, b);
538                     break;
539                 case 1:
540                     setTextColor(12);
541                     outputFile<<"Nhap tay"<<endl;
542                     cout << "Nhap kich thuoc ma tran: ";

```

```

542         outputFile << "Nhap kích thước ma tran: ";
543         cin >> size;
544         outputFile << size << endl;
545         cout << "Nhap ma tran A:" << endl;
546         outputFile << "Nhap ma tran A:" << endl;
547
548         NhapMaTran(A, size, size);
549         cout << "Ma tran A:" << endl;
550         XuatMaTran(A, size, size, outputFile);
551
552         cout << "-----" << endl;
553         outputFile << "-----" << endl;
554
555         cout << "Nhap vector b:" << endl;
556         outputFile << "Nhap vector b:" << endl;
557         NhapMaTran(b, size, 1);
558         cout << "Vector b:" << endl;
559         XuatMaTran(b, size, 1, outputFile);
560         break;
561     }
562 }
563 setTextColor(15);
564 cout << endl;
565 cout << "Nhan phim bat ki de tiep tục...";
566 getch();
567 break;
568 }
569 }
570 }
571
572 void chucnang4(ofstream &outputFile, double D[100][100], double E[100][100],
573 int size)
574 {
575     int choice = 0;
576     int maxChoice = 2;
577     char key;
578     while (true) {
579         clearScreen();
580         cout << "===== "
581         << endl;
582         cout << " |                               Giai gan dung voi
583 so lan lap k | " << endl;
584         cout << "===== "
585         << endl;
586
587         for (int i = 0; i <= maxChoice; i++) {
588             gotoxy(0, i + 3);
589             if (i == choice) {
590                 printHighlightedOption(">>");
591                 switch (i) {
592                     case 0:
593                         cout << " |";
594                         printHighlightedOption(" 1. Lap don");
595                         cout << "
596                         |";
597                         break;
598                     case 1:
599                         cout << " |";

```

```

595         printHighlightedOption("    2. Lap Seidel");
596         cout << "
597             |";
598         break;
599     case 2:
600         cout << "            |";
601         printHighlightedOption("    Thoat");
602         cout << "
603             |";
604         break;
605     }
606 }
607 else {
608     printNormalOption(" ");
609     switch (i) {
610     case 0:
611         cout << "            |";
612         printNormalOption("    1. Lap don");
613         cout << "
614             |";
615         break;
616     case 1:
617         cout << "            |";
618         printNormalOption("    2. Lap Seidel");
619         cout << "
620             |";
621         break;
622     case 2:
623         cout << "            |";
624         printNormalOption("    Thoat");
625         cout << "
626             |";
627         break;
628     }
629 }
630 }
631 cout << endl;
632 cout << "===== "
<< endl;
633 key = getch();
634 switch (key) {
635 case 'w':
636 case 'W':
637 case 72:
638     if (choice > 0)
639     {
640         choice--;
641         gotoxy(0, choice + 4);
642         printHighlightedOption(">>");
643         gotoxy(0, choice + 5);
644         printNormalOption(" ");
645     }
646     break;
647 case 's':
648 case 'S':
649 case 80:
650     if (choice < maxChoice)
651     {

```

```

647         choice++;
648         gotoxy(0, choice + 2);
649         printHighlightedOption(">>");
650         gotoxy(0, choice + 1);
651         printNormalOption(" ");
652     }
653     break;
654 case '\r':
655     clearScreen();
656     if (choice == maxChoice) {
657         return;
658     }
659     else {
660         switch (choice) {
661             case 0:
662                 setTextColor(10);
663                 outputFile<<"Lap don k lan:"<<endl;
664                 double D1[100][100], E1[100][100];
665                 copy(D,D1,size);
666                 copy(E,E1,size);
667                 cout<<"Lap don" << endl;
668                 int n_loop;
669                 cout<<"Nhap vao so lan lap: ";cin>>n_loop;
670                 outputFile<<"Nhap vao so lan lap: ";outputFile<<n_loop;
671
672                 loop_KDon(D1, E1, size, n_loop,outputFile);
673                 break;
674             case 1:
675                 setTextColor(12);
676                 outputFile<<"Lap Seidel k lan:"<<endl;
677                 cout<<"Lap Seidel: " << endl;
678                 int n_loop1;
679                 cout<<"Nhap vao so lan lap: ";cin>>n_loop1;
680                 outputFile<<"Nhap vao so lan lap: "<<n_loop<<endl;
681                 double D6[100][100], E6[100][100];
682                 copy(D,D6,size);
683                 copy(E,E6,size);
684
685                 loop_kSeidel(D6, E6, size, n_loop1,outputFile);
686                 break;
687         }
688     }
689     setTextColor(15);
690     cout << endl;
691     cout << "Nhan phim bat ki de tiep tục...";
692     getch();
693     break;
694 }
695 }
696 }
697
698 void chucnang5(ofstream &outputFile, double D[100][100], double E[100][100],
699             int size)
700 {
701     int choice = 0;
702     int maxChoice = 2;
703     char key;
704     while (true) {

```

```

704     clearScreen();
705     cout << "===== "
<< endl;
706     cout << " | Gia hạn dung voi sai
so e cho truoct(tien nghiem) | " << endl;
707     cout << "=====
" << endl;
708     for (int i = 0; i <= maxChoice; i++) {
709         gotoxy(0, i + 3);
710         if (i == choice) {
711             printHighlightedOption(">>");
712             switch (i) {
713                 case 0:
714                     cout << " |";
715                     printHighlightedOption(" 1. Lap don");
716                     cout << "
|";
717                     break;
718                 case 1:
719                     cout << " |";
720                     printHighlightedOption(" 2. Lap Seidel");
721                     cout << "
|";
722                     break;
723                 case 2:
724                     cout << " |";
725                     printHighlightedOption(" Thoat");
726                     cout << "
|";
727                     break;
728             }
729         }
730         else {
731             printNormalOption(" ");
732             switch (i) {
733                 case 0:
734                     cout << " |";
735                     printNormalOption(" 1. Lap don");
736                     cout << "
|";
737                     break;
738                 case 1:
739                     cout << " |";
740                     printNormalOption(" 2. Lap Seidel");
741                     cout << "
|";
742                     break;
743                 case 2:
744                     cout << " |";
745                     printNormalOption(" Thoat");
746                     cout << "
|";
747                     break;
748             }
749         }
750     }
751     cout << endl;
752     cout << "===== "

```

```

753     << endl;
754     key = getch();
755     switch (key) {
756     case 'w':
757     case 'W':
758     case 72:
759         if (choice > 0)
760         {
761             choice--;
762             gotoxy(0, choice + 4);
763             printHighlightedOption(">>");
764             gotoxy(0, choice + 5);
765             printNormalOption(" ");
766         }
767         break;
768     case 's':
769     case 'S':
770     case 80:
771         if (choice < maxChoice)
772         {
773             choice++;
774             gotoxy(0, choice + 2);
775             printHighlightedOption(">>");
776             gotoxy(0, choice + 1);
777             printNormalOption(" ");
778         }
779         break;
780     case '\r':
781         clearScreen();
782         if (choice == maxChoice) {
783             return;
784         }
785         else {
786             switch (choice) {
787             case 0:
788                 setTextColor(11);
789                 outputFile<<"Lap don voi sai so e cho truoc:"<<endl;
790                 double D2[100][100], E2[100][100];
791                 copy(D,D2,size);
792                 copy(E,E2,size);
793                 cout<<"Lap don voi sai so e cho truoc: " << endl;
794                 double e;
795                 cout<<"Nhap vao sai so e: ";cin>>e;
796
797                 loop_eDon(D2, E2, size, e,outputFile);
798                 break;
799             case 1:
800                 setTextColor(10);
801                 outputFile<<endl<<"Lap Seidel voi sai so e cho truoc:"<<
802                 endl;
803                 cout<<"Lap Seidel voi sai so e cho truoc " << endl;
804                 double eps;
805                 cout<<"Nhap vao sai so e: ";cin>>eps;
806                 outputFile<<"Nhap vao sai so e: ";outputFile<<eps<<endl;
807                 double D5[100][100], E5[100][100];
808                 copy(D,D5,size);
809                 copy(E,E5,size);
810                 loop_eSeidel(D5, E5, size, eps,outputFile);

```

```

809         break;
810     }
811 }
812 setTextColor(15);
813 cout << endl;
814 cout << "Nhan phim bat ki de tiep tuc...";
815 getch();
816 break;
817 }
818 }
819 }
820
821 void chucnang6(ofstream &outputFile, double D[100][100], double E[100][100],
822 int size)
823 {
824     int choice = 0;
825     int maxChoice = 2;
826     char key;
827     while (true) {
828         clearScreen();
829         cout << "===== "
830 << endl;
831         cout << " | Gia gan dung voi dieu
832 kien cho truoct(hau nghiem) | " << endl;
833         cout << "=====
834 " << endl;
835         for (int i = 0; i <= maxChoice; i++) {
836             gotoxy(0, i + 3);
837             if (i == choice) {
838                 printHighlightedOption(">>");
839                 switch (i) {
840                     case 0:
841                         cout << " |";
842                         printHighlightedOption(" 1. Lap don");
843                         cout << "
844 |";
845                         break;
846                     case 1:
847                         cout << " |";
848                         printHighlightedOption(" 2. Lap Seidel");
849                         cout << "
850 |";
851                         break;
852                     case 2:
853                         cout << " |";
854                         printHighlightedOption(" Thoat");
855                         cout << "
856 |";
857                         break;
858                 }
859             }
860             else {
861                 printNormalOption(" ");
862                 switch (i) {
863                     case 0:
864                         cout << " |";
865                         printNormalOption(" 1. Lap don");
866                         cout << "

```

```

        |";
860         break;
861     case 1:
862         cout << "        |";
863         printNormalOption("    2. Lap Seidel");
864         cout << "
        |";
865         break;
866     case 2:
867         cout << "        |";
868         printNormalOption("    Thoat");
869         cout << "
        |";
870         break;
871     }
872 }
873 }
874 cout << endl;
875 cout << "===== " <<
endl;
876 key = getch();
877 switch (key) {
878     case 'w':
879     case 'W':
880     case 72:
881         if (choice > 0)
882         {
883             choice--;
884             gotoxy(0, choice + 4);
885             printHighlightedOption(">>");
886             gotoxy(0, choice + 5);
887             printNormalOption(" ");
888         }
889         break;
890     case 's':
891     case 'S':
892     case 80:
893         if (choice < maxChoice)
894         {
895             choice++;
896             gotoxy(0, choice + 2);
897             printHighlightedOption(">>");
898             gotoxy(0, choice + 1);
899             printNormalOption(" ");
900         }
901         break;
902     case '\r':
903         clearScreen();
904         if (choice == maxChoice) {
905             return;
906         }
907         else {
908             switch (choice) {
909                 case 0:
910                     setTextColor(14);
911                     outputFile<<"Lap don co dieu kien voi sai so e cho truoc
: "<<endl;
912                     cout << "Lap don co dieu kien voi sai so e cho truoc

```



```

: " << endl;
913         double exp;
914         double D3[100][100], E3[100][100];
915         copy(D,D3,size);
916         copy(E,E3,size);
917         cout<<"Nhap vao sai so e: ";cin>>exp;
918
919         loop_Don_with_condition(D3, E3, size, exp,outputFile);
920         break;
921     case 1:
922         setTextColor(10);
923         outputFile<<"Lap Seidel co dieu kien voi sai so e cho
truoc:"<<endl;
924         cout<<"Lap Seidel co dieu kien voi sai so e cho truoc: "
<< endl;
925         double eps;
926         cout<<"Nhap vao sai so e: ";cin>>eps;
927         double D4[100][100], E4[100][100];
928         copy(D,D4,size);
929         copy(E,E4,size);
930         loop_Seidel_with_condition(D4, E4, size, eps,outputFile)
;
931         break;
932     }
933 }
934 setTextColor(15);
935 cout << endl;
936 cout << "Nhan phim bat ki de tiep tục...";
937 getch();
938 break;
939 }
940 }
941 }
942
943 void setupMenu(ofstream &outputFile)
944 {
945     int choice = 0;
946     int maxChoice = 6;
947     char key;
948     while (true) {
949         clearScreen();
950
951         cout << "=====
" << endl;
952         cout << "          | GIAI GAN DUNG HE PHUONG TRINH AX = b BANG
PHUONG PHAP LAP DON, LAP SEIDELDAY CUNG |          " << endl;
953         cout << "=====
" << endl;
954         for (int i = 0; i <= maxChoice; i++) {
955             gotoxy(0, i + 3);
956             if (i == choice) {
957
958                 printHighlightedOption(">>");
959                 switch (i) {
960                     case 0:
961                         cout<<"          |";
962                         printHighlightedOption("    1. Nhap vao A, b theo
khuon dang cua ma tran");

```

```

963         cout<<"                |";
964         break;
965     case 1:
966         cout<<"                |";
967         printHighlightedOption("    2. Kiem tra tinh cheo
troi cua ma tran A va su hoi tu cua phuong phap");
968         cout<<"                |";
969         break;
970     case 2:
971         cout<<"                |";
972         printHighlightedOption("    3. Tinh chuan cua ma tran
A doi voi he da cho.");
973         cout<<"                |";
974         break;
975     case 3:
976         cout<<"                |";
977         printHighlightedOption("    4. Tinh nghiem gan dung
voi so lan lap k cho truooc, danh gia sai so.");
978         cout<<"                |";
979         break;
980     case 4:
981         cout<<"                |";
982         printHighlightedOption("    5. Tinh nghiem gan dung
voi sai so e cho truooc.");
983         cout<<"                |";
984         break;
985     case 5:
986         cout<<"                |";
987         printHighlightedOption("    6. Tinh nghiem gan dung X
(k) thoa man: ||X(k)-X(k-1)|| <= e cho truooc");
988         cout<<"                |";
989         break;
990     case 6:
991         cout<<"                |";
992         printHighlightedOption("    Thoat");
993         cout<<"                |";
994         break;
995     }
996 }
997 else {
998
999     printNormalOption(" ");
1000     switch (i) {
1001     case 0:
1002         cout<<"                |";
1003         printNormalOption("    1. Nhap vao A, b theo khuon
dang cua ma tran");
1004         cout<<"                |";
1005         break;
1006     case 1:
1007         cout<<"                |";
1008         printNormalOption("    2. Kiem tra tinh cheo troi cua
ma tran A va su hoi tu cua phuong phap");
1009         cout<<"                |";
1010         break;
1011     case 2:
1012         cout<<"                |";

```

```

1013         printNormalOption("    3. Tính chuẩn của ma trận A
doi voi he da cho.");
1014         cout<<"                |";
1015         break;
1016     case 3:
1017         cout<<"                |";
1018         printNormalOption("    4. Tính nghiệm gần đúng với số
lan lap k cho truooc, danh gia sai so.");
1019         cout<<"                |";
1020         break;
1021     case 4:
1022         cout<<"                |";
1023         printNormalOption("    5. Tính nghiệm gần đúng với
sai so e cho truooc.");
1024         cout<<"                |";
1025         break;
1026     case 5:
1027         cout<<"                |";
1028         printNormalOption("    6. Tính nghiệm gần đúng X(k)
thoa man: ||X(k)-X(k-1)|| <= e cho truooc");
1029         cout<<"                |";
1030         break;
1031     case 6:
1032         cout<<"                |";
1033         printNormalOption("    Thoat");
1034         cout<<"                |";
1035         break;
1036     }
1037 }
1038 }
1039 cout<<endl;
1040
1041     cout << "                =====
" << endl;
1042     cout<<endl<<endl<<endl<<endl<<endl;
1043     cout<<"                Huong dan:"<<endl;
1044     cout<<"                - Di chuyen bang phim: len, xuong, s, w"
<<endl;
1045     cout<<"                - Chon chuc nang      : Enter"<<endl;
1046
1047     key = getch();
1048     switch (key) {
1049         case 'w':
1050         case 'W':
1051         case 72:
1052             if (choice > 0)
1053             {
1054                 choice--;
1055             }
1056             break;
1057         case 's':
1058         case 'S':
1059         case 80:
1060             if (choice < maxChoice)
1061             {
1062                 choice++;
1063             }

```

```

1064         break;
1065     case '\\r':
1066         clearScreen();
1067
1068         if (choice == maxChoice)
1069         {
1070             cout << "Thoat chuong trinh." << endl;
1071             exit(0);
1072         }
1073         else {
1074             cout << "Ban da lua chon: ";
1075             cout<<endl;
1076             switch (choice)
1077             {
1078                 case 0:
1079                     setTextColor(14);
1080                     double A[100][100];
1081                     double b[100][100];
1082                     int size;
1083                     cout<<"Nhap kích thước ma tran: ";cin>>size;
1084                     chucnang1(outputFile, A, b, size);
1085
1086                     cout << "-----" << endl;
1087                     outputFile << "-----" << endl;
1088
1089                     cout << "Ma tran B la:" << endl;
1090                     outputFile << "Ma tran B la:" << endl;
1091                     double D[100][100];
1092                     timLamda(A, D, size);
1093                     XuatMaTran(D, size, size, outputFile);
1094                     double E[100][100];
1095                     cout << "vetor d la:" << endl;
1096                     outputFile << "vetor d la:" << endl;
1097                     timvecto_d(A, b, E);
1098                     XuatMaTran(E, size, 1, outputFile);
1099                     break;
1100                 case 1:
1101                     outputFile<<"Lua chon 2:"<<endl;
1102                     setTextColor(13);
1103                     cout << "Kiem tra tinh cheo troi cua A: "
1104                     << endl;
1105                     outputFile << "Kiem tra tinh cheo troi cua A: "
1106                     << endl;
1107                     if (cheoTroiHang(A, size)){
1108                         cout << " A cheo troi"<<endl;
1109                         outputFile<< " A cheo troi"<<endl;
1110                     }else{
1111                         cout << " A không cheo troi"<<endl;
1112                         outputFile<< " A không cheo troi"<<endl;
1113                     }
1114                     cout << "-----"<<endl;
1115                     outputFile<<"-----"<<endl;
1116                     if(chuanHang(D, size) < 1.0)
1117                     {
1118                         cout<<"Do || Q || = "<<chuanHang(D, size)<<
1119                         endl;
1120                         outputFile<<"Do || Q || = "<<chuanHang(D,
1121                         size)<<endl;

```

```

1118         cout << "Phuong phap hoi tu";
1119         outputFile<<"Phuong phap hoi tu";
1120     }else{
1121         cout<<"Do ||Q|| = "<<chuanHang(D, size)<<
endl;
1122         cout << "Phuong phap khong hoi tu";
1123         outputFile<<"Phuong phap khong hoi tu";
1124     }
1125     break;
1126 case 2:
1127     outputFile<<"Lua chon 3:"<<endl;
1128     setTextColor(14);
1129     cout<<"Chuan cot cua A: " <<chuanCot(A, size)<<
endl;
1130     cout<<"Chuan hang cua A: " <<chuanHang(A, size)
<< endl;
1131     cout<<"Chuan cot cua B: " <<chuanCot(D, size)<<
endl;
1132     cout<<"Chuan hang cua B: " <<chuanHang(D, size)
<< endl;
1133     cout<<"Chuan cot cua vector d: " << chuanCot(E,
size)<< endl;
1134     cout<<"Chuan hang cua vector d: " << chuanHang(E
, size)<< endl;
1135     outputFile<<"Chuan cot cua A: " <<chuanCot(A,
size)<< endl;
1136     outputFile<<"Chuan hang cua A: " <<chuanHang(A,
size)<< endl;
1137     outputFile<<"Chuan cot cua B: " <<chuanCot(D,
size)<< endl;
1138     outputFile<<"Chuan hang cua B: " <<chuanHang(D,
size)<< endl;
1139     outputFile<<"Chuan cot cua vector d: " <<
chuanCot(E, size)<< endl;
1140     outputFile<<"Chuan hang cua vector d: " <<
chuanHang(E, size)<< endl;
1141     break;
1142 case 3:
1143     outputFile<<"Lua chon 4:"<<endl;
1144     setTextColor(9);
1145     if(chuanHang(D, size)>=1)
1146     {
1147         cout << "He phuong trinh ko giai duoc
bang lap don, lap seidel"<< endl;
1148         outputFile<<"He phuong trinh ko giai duoc
bang lap don, lap seidel"<< endl;
1149     }else
1150     {
1151         chucnang4(outputFile, D, E, size);
1152     }
1153     break;
1154 case 4:
1155     outputFile<<"Lua chon 5:"<<endl;
1156     setTextColor(13);
1157     outputFile<<"Lua chon 5:"<<endl;
1158     if(chuanHang(D, size)>= 1){
1159         cout<<"He da cho khong giai duoc bang pp lap
don, seidel do he khong hoi tu: "<<endl;

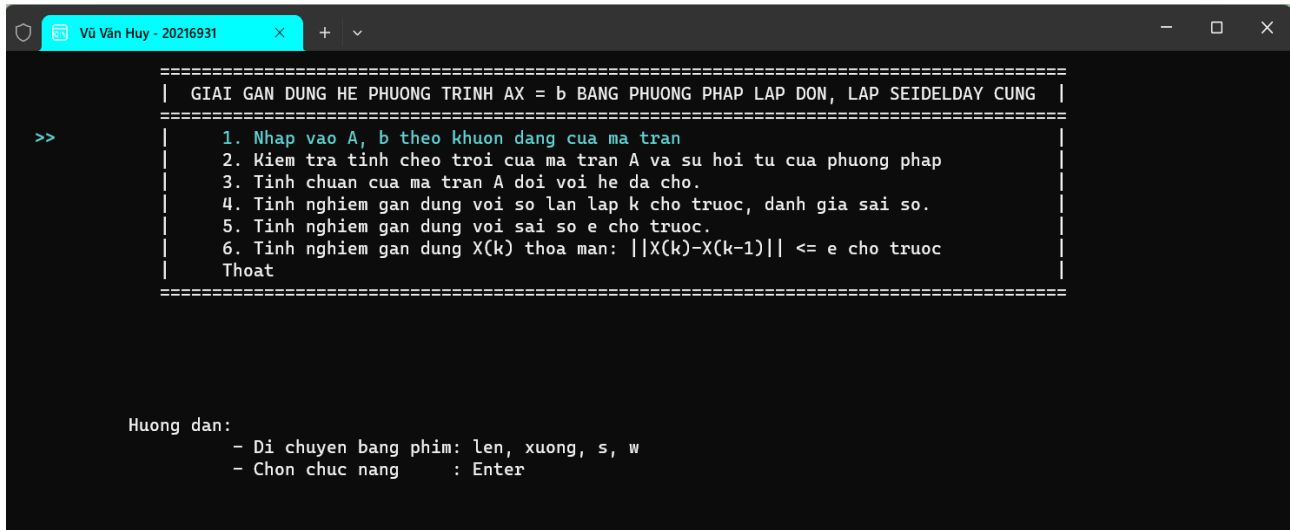
```

```
1160         outputFile<<"He da cho khong giai duoc bang
pp lap don, seidel do he khong hoi tu: "<<endl;
1161     }else
1162     {
1163         chucnang5(outputFile, D, E, size);
1164     }
1165     break;
1166     case 5:
1167         setTextColor(11);
1168         outputFile<<"Lua chon 6:"<<endl;
1169         if(chuanHang(D, size)>=1)
1170         {
1171             cout<<"He da cho khong giai duoc bang lap
don, lap seidel"<<endl;
1172             outputFile<<"He da cho khong giai duoc bang
lap don, lap seidel"<<endl;
1173             }else{
1174                 chucnang6(outputFile,D, E, size);
1175             }
1176         }
1177         setTextColor(15);
1178         cout<<endl;
1179         cout << "Nhan phim bat ki de tiep tuc...";
1180         outputFile <<endl<< " _ _ _ _ _ "<<endl;
1181         getch();
1182     }
1183     break;
1184 }
1185 }
1186 }
1187 int main()
1188 {
1189     ofstream outputFile("output.txt");
1190     setupMenu(outputFile);
1191     outputFile.close();
1192     return 0;
1193 }
```

## 6

## Kết quả khi chạy chương trình

## 6.1 Chạy menu



```
Vũ Văn Huy - 20216931 x + v - □ ×

=====
| GIAI GAN DUNG HE PHUONG TRINH AX = b BANG PHUONG PHAP LAP DON, LAP SEIDELDAY CUNG |
=====
>> 1. Nhập vào A, b theo khuôn dạng của ma trận
    2. Kiểm tra tính chéo trội của ma trận A và sự hội tụ của phương pháp
    3. Tính chuẩn của ma trận A đối với hệ đã cho.
    4. Tính nghiệm gần đúng với số lần lặp k cho trước, đánh giá sai số.
    5. Tính nghiệm gần đúng với sai số e cho trước.
    6. Tính nghiệm gần đúng X(k) thỏa mãn: ||X(k)-X(k-1)|| <= e cho trước
    Thoát

=====

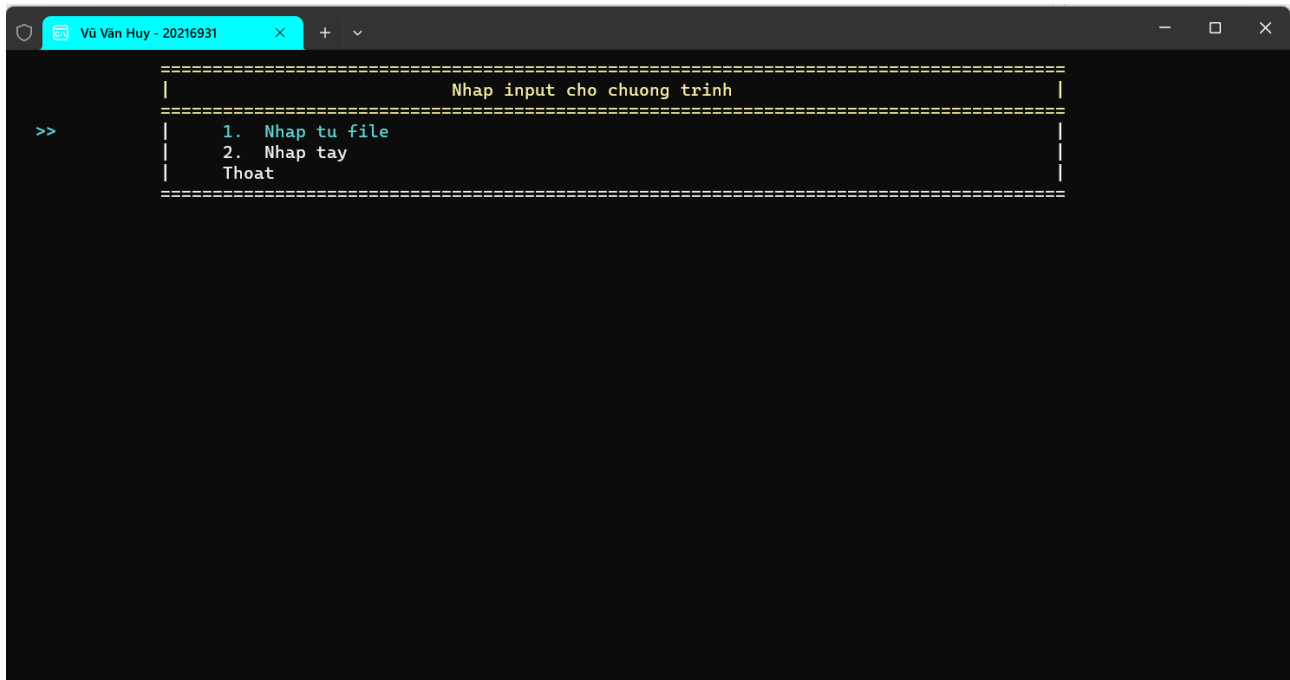
Huong dan:
- Di chuyen bang phim: len, xuong, s, w
- Chon chuc nang      : Enter
```

Hình 2: Menu chương trình

Thao tác trên menu:

- Di chuyển bằng các phím S, W, ↓, ↑
- Chọn chức năng bằng Enter.

## 6.2 Nhập input cho chương trình



Hình 3: Giao diện nhập file

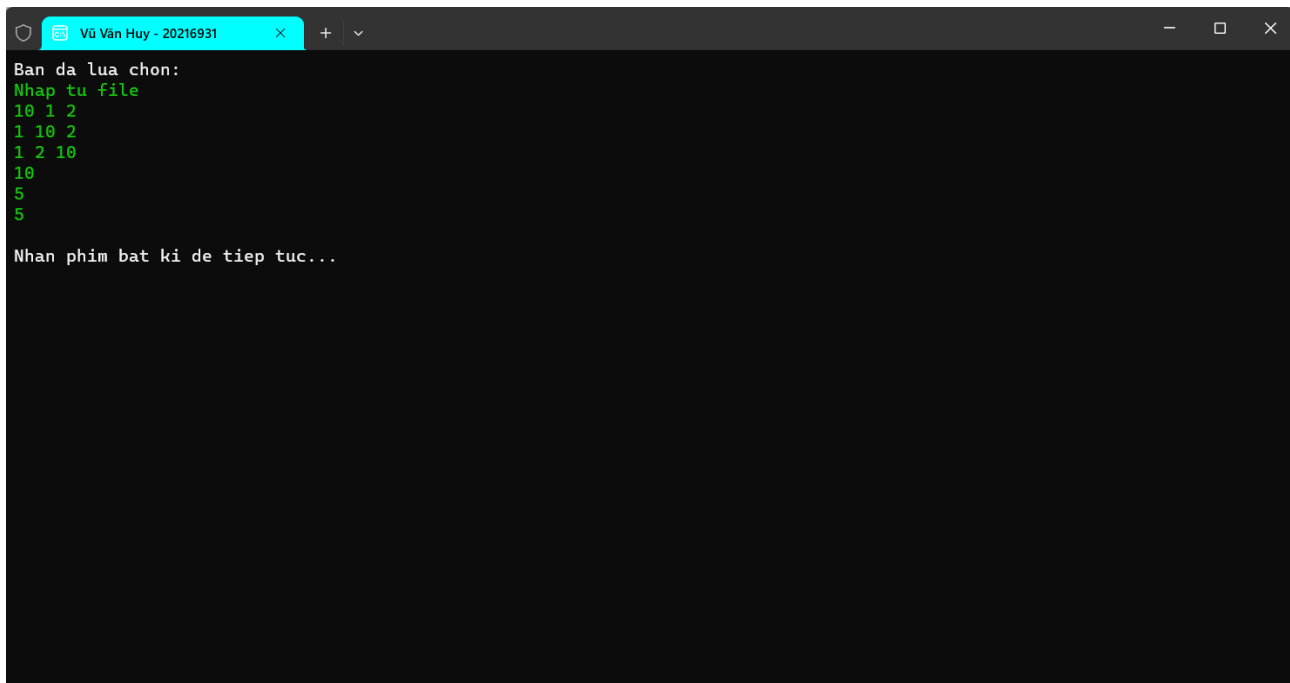
Input 1:

$$A = \begin{bmatrix} 10 & 1 & 2 \\ 1 & 10 & 2 \\ 1 & 2 & 10 \end{bmatrix}; \quad b = \begin{bmatrix} 10 \\ 5 \\ 5 \end{bmatrix}$$

Input 2:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 1 & 1 \end{bmatrix}; \quad b = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$



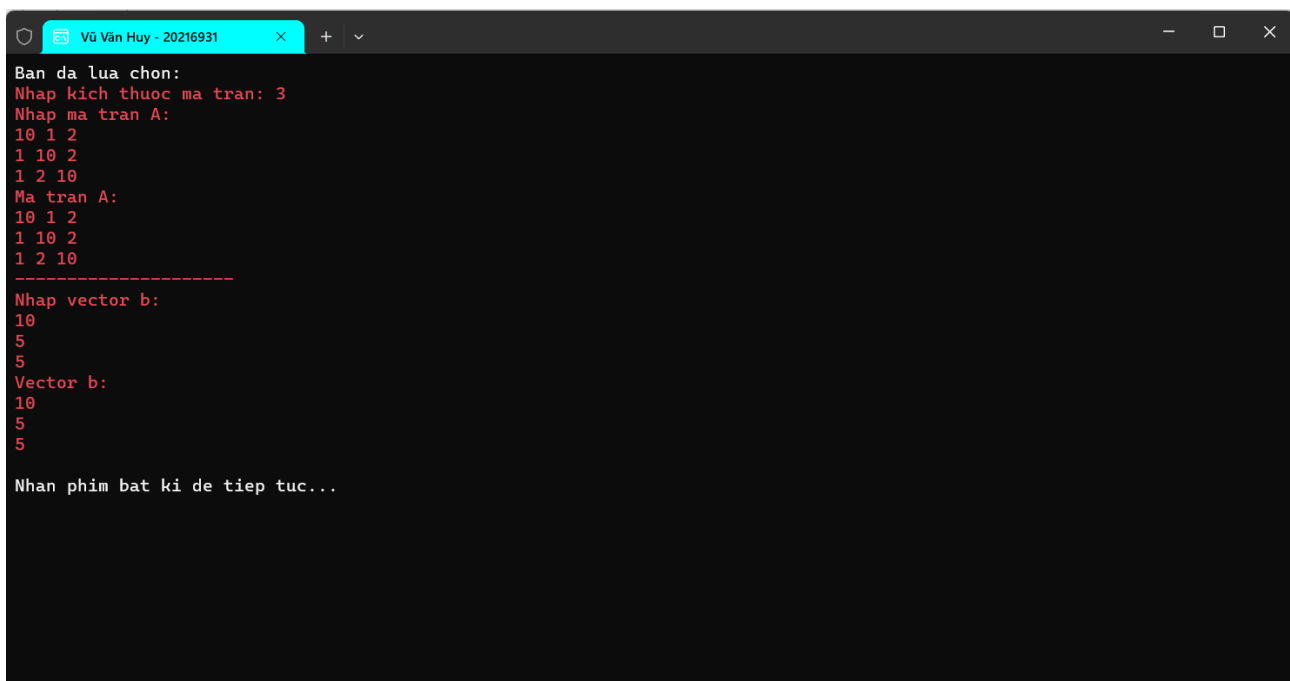


A screenshot of a terminal window with a dark background. The title bar at the top shows a shield icon, a file icon, and the text 'Vũ Văn Huy - 20216931'. The terminal content is as follows:

```
Ban da lua chon:  
Nhap tu file  
10 1 2  
1 10 2  
1 2 10  
10  
5  
5  
  
Nhan phim bat ki de tiep tục...
```

Hình 4: Nhập từ file input 1

Hoặc nhập tay:



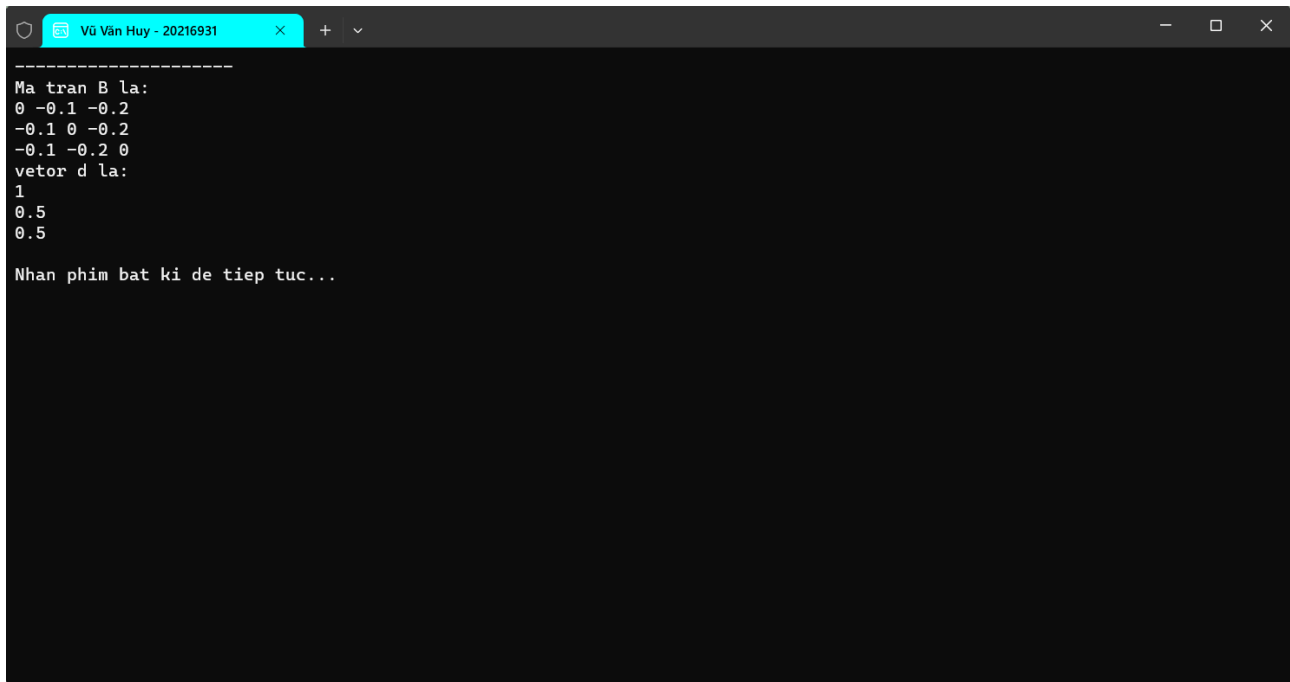
A screenshot of a terminal window with a dark background. The title bar at the top shows a shield icon, a file icon, and the text 'Vũ Văn Huy - 20216931'. The terminal content is as follows:

```
Ban da lua chon:  
Nhap kích thước ma tran: 3  
Nhap ma tran A:  
10 1 2  
1 10 2  
1 2 10  
Ma tran A:  
10 1 2  
1 10 2  
1 2 10  
-----  
Nhap vector b:  
10  
5  
5  
Vector b:  
10  
5  
5  
  
Nhan phim bat ki de tiep tục...
```

Hình 5: Nhập tay input 1

Khi chạy đồng thời cho kết quả ma trận B, d sau khi biến đổi hệ ban đầu

$$X = BX + d$$

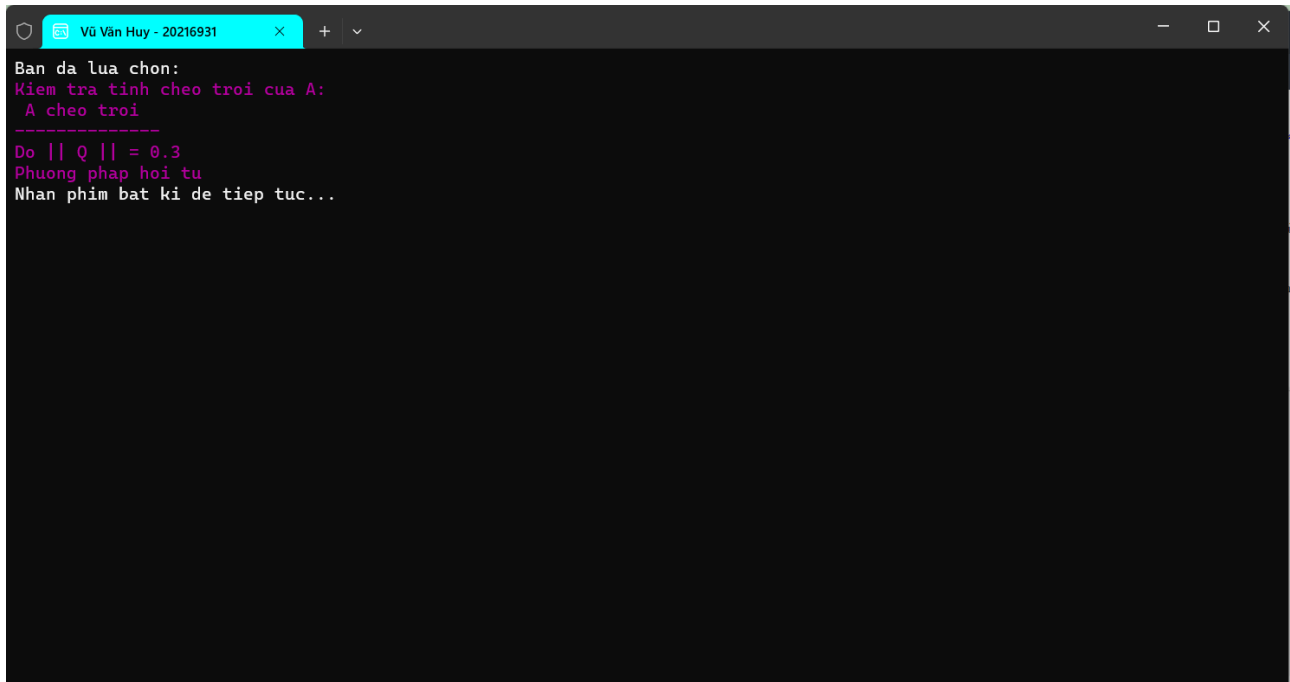


```
-----  
Ma tran B la:  
0 -0.1 -0.2  
-0.1 0 -0.2  
-0.1 -0.2 0  
vetor d la:  
1  
0.5  
0.5  
  
Nhan phim bat ki de tiep tục...
```

Hình 6: Xuất ra ma trận B, d(từ input1) sau khi thoát chức năng 1

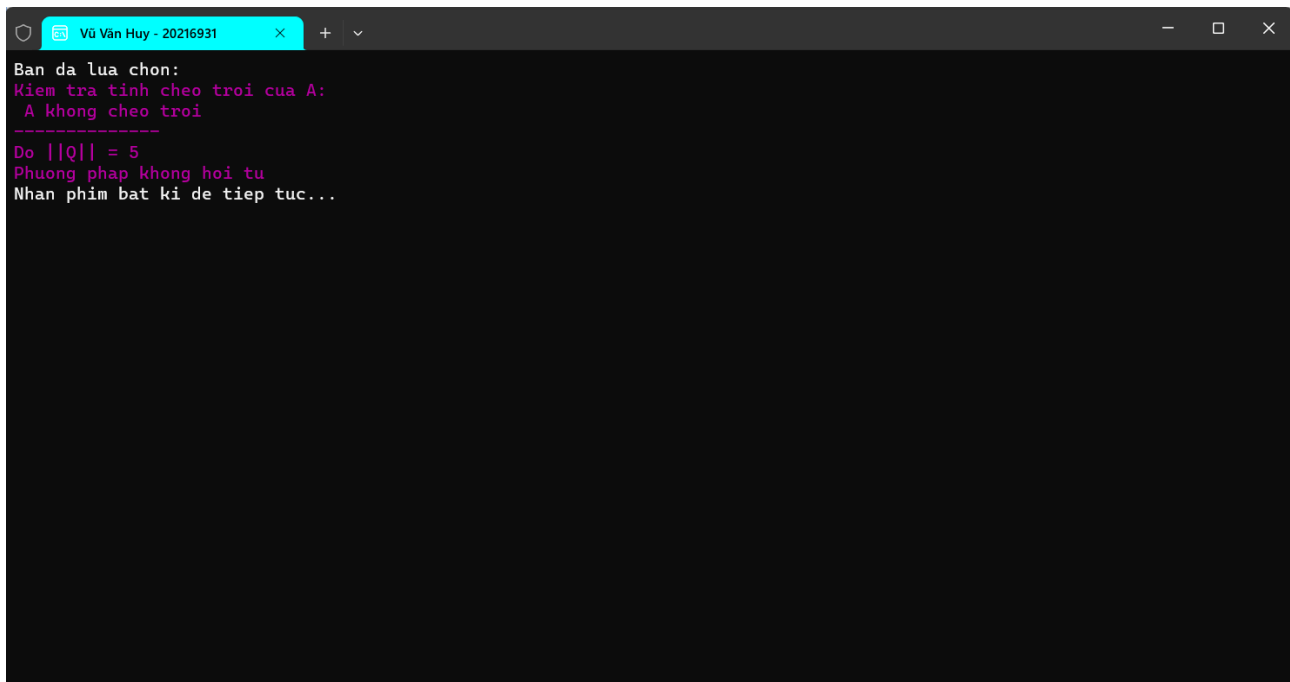
Tương tự đối với input 2.

### 6.3 Kiểm tra tính chéo trội và sự hội tụ của phương pháp



```
Vũ Văn Huy - 20216931
Ban da lua chon:
Kiem tra tinh chéo trội của A:
A chéo trội
-----
Do || Q || = 0.3
Phuong phap hội tụ
Nhan phim bat ki de tiếp tục...
```

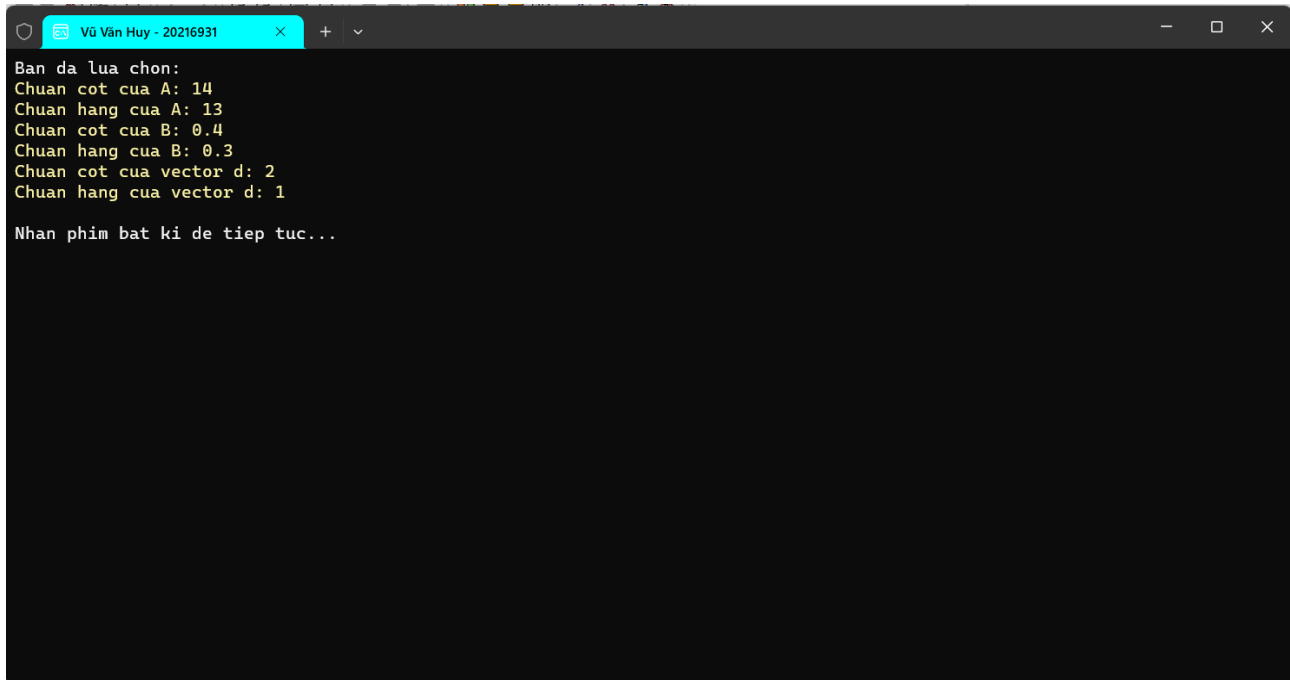
Hình 7: Xác định chéo trội, tính hội tụ input 1



```
Vũ Văn Huy - 20216931
Ban da lua chon:
Kiem tra tinh chéo trội của A:
A không chéo trội
-----
Do ||Q|| = 5
Phuong phap không hội tụ
Nhan phim bat ki de tiếp tục...
```

Hình 8: Xác định chéo trội, tính hội tụ input 2

## 6.4 Tính chuẩn của các ma trận sau khi biến đổi

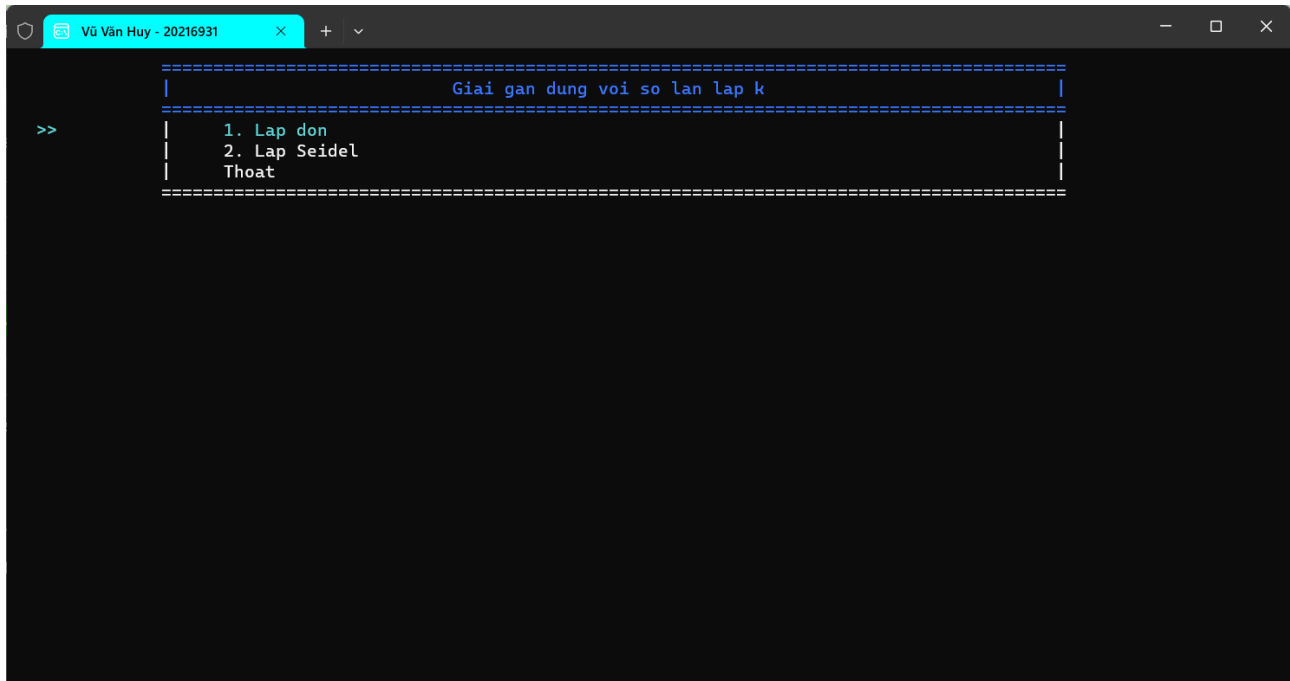


```
Ban da lua chon:  
Chuan cot cua A: 14  
Chuan hang cua A: 13  
Chuan cot cua B: 0.4  
Chuan hang cua B: 0.3  
Chuan cot cua vector d: 2  
Chuan hang cua vector d: 1  
  
Nhan phim bat ki de tiep tuc...
```

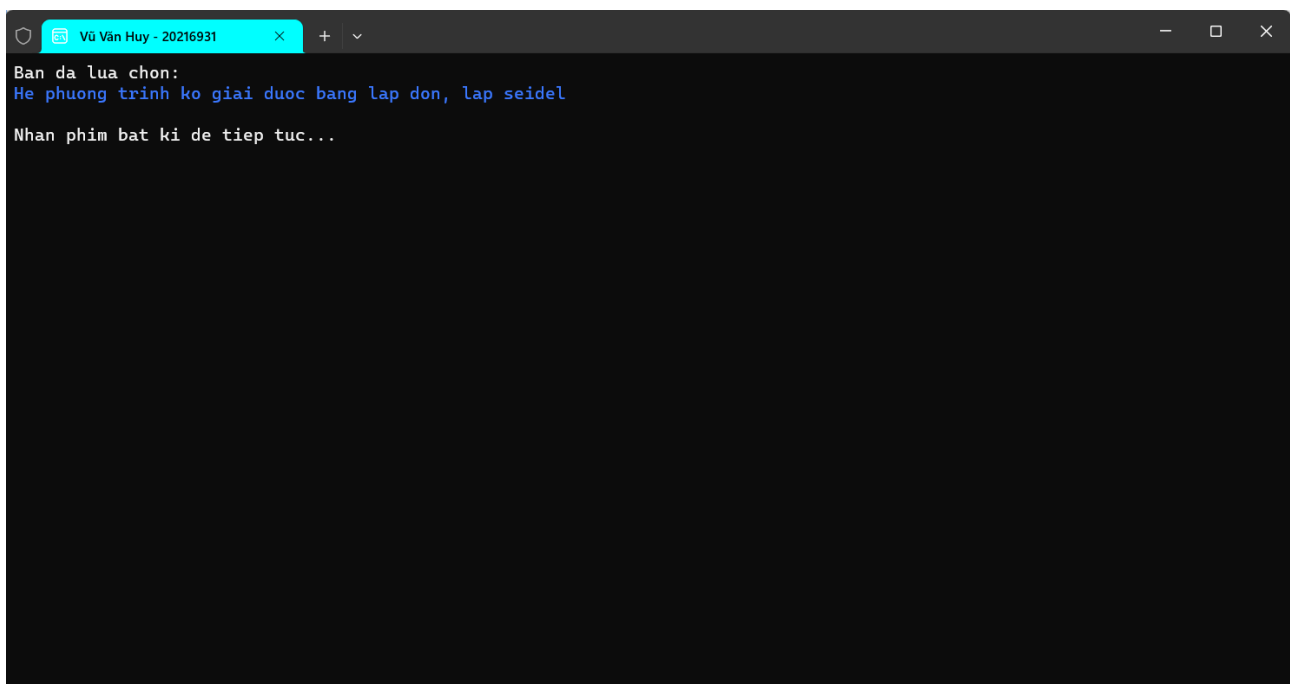
Hình 9: Tính các chuẩn

Viết cái gì đó ra đây đi

## 6.5 Tính nghiệm gần đúng với số lần lặp k



Hình 10: Menu con lặp k lần



Hình 11: Hiển thị thông báo nếu phương pháp không hội tụ.

```
Vũ Văn Huy - 20216931
Lap đơn
Nhap vao so lan lap: 5
Nhap xap xi dau: x_0:
Chon luon xap xi dau la ma tran d.
Lan 1
0.85
0.3
0.3
Lan 2
0.91
0.355
0.355
Lan 3
0.8935
0.338
0.338
Lan 4
0.8986
0.34305
0.34305
Lan 5
0.897085
0.34153
0.34153

Danh gia sai so:
sai so: (q/(1-q))(X_5 - X_4) = 0.000651429
Nhan phim bat ki de tiep tục...
```

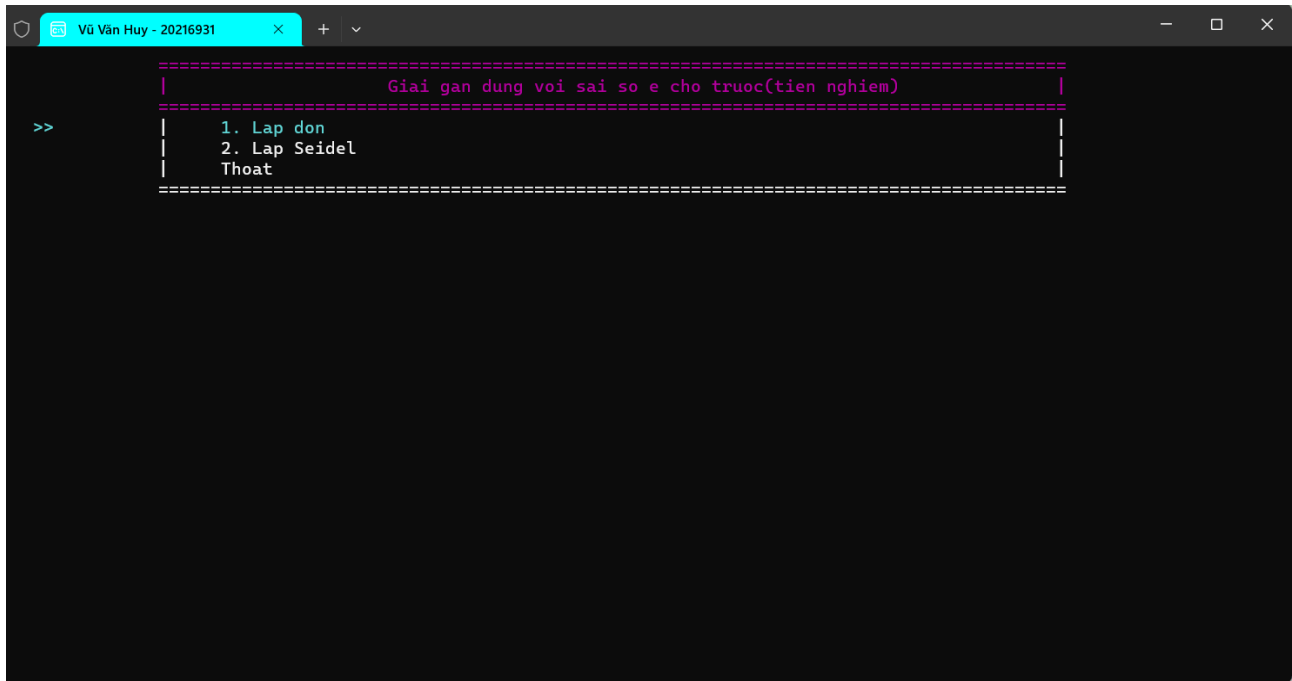
Hình 12: Lặp đơn k lần

```
Vũ Văn Huy - 20216931
Lap Seidel:
Nhap vao so lan lap: 5
Nghiem cua he phuong trinh la:
Lan1
x[1] = 0.85
x[2] = 0.315
x[3] = 0.352
Lan2
x[1] = 0.8981
x[2] = 0.33979
x[3] = 0.342232
Lan3
x[1] = 0.897575
x[2] = 0.341796
x[3] = 0.341883
Lan4
x[1] = 0.897444
x[2] = 0.341879
x[3] = 0.34188
Lan5
x[1] = 0.897436
x[2] = 0.34188
x[3] = 0.34188

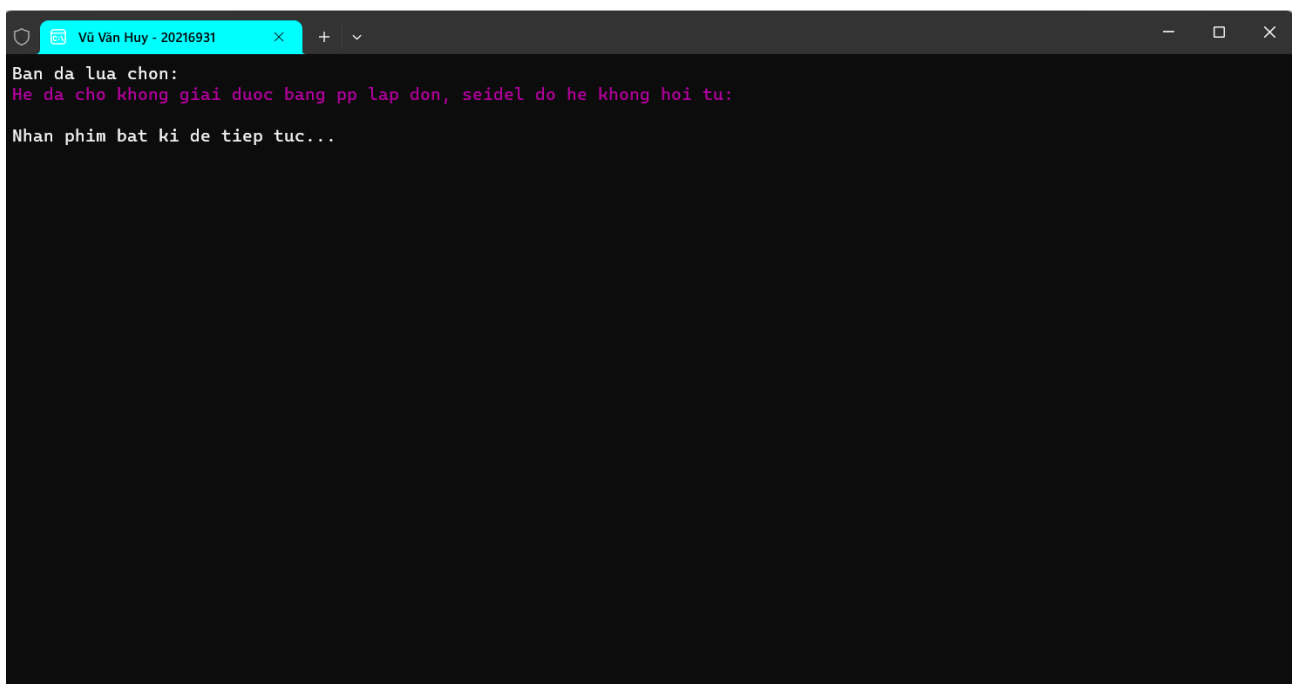
Danh gia sai so:
Sai so: (q/(1-q))(X_5 - X_4) = 3.25159e-006
Nhan phim bat ki de tiep tục...
```

Hình 13: Lặp Seidel k lần

## 6.6 Tính nghiệm gần đúng với sai số $\epsilon$ cho trước



Hình 14: Menu con lặp với sai số  $\epsilon$



Hình 15: Hiển thị thông báo nếu phương pháp không hội tụ

```
Lap don voi sai so e cho truooc:
Nhap vao sai so e: 0.0001
Lan 1
0.85
0.3
0.3
Lan 2
0.91
0.355
0.355
Lan 3
0.8935
0.338
0.338
Lan 4
0.8986
0.34305
0.34305
Lan 5
0.897085
0.34153
0.34153
Lan 6
0.897541
0.341985
0.341985
Lan 7
0.897404
0.341849
0.341849
Da giai xong he phuong trinh trong 7 buoc lap.

Nhan phim bat ki de tiep tục...
```

Hình 16: Lặp đơn với sai số  $\epsilon$  cho trước

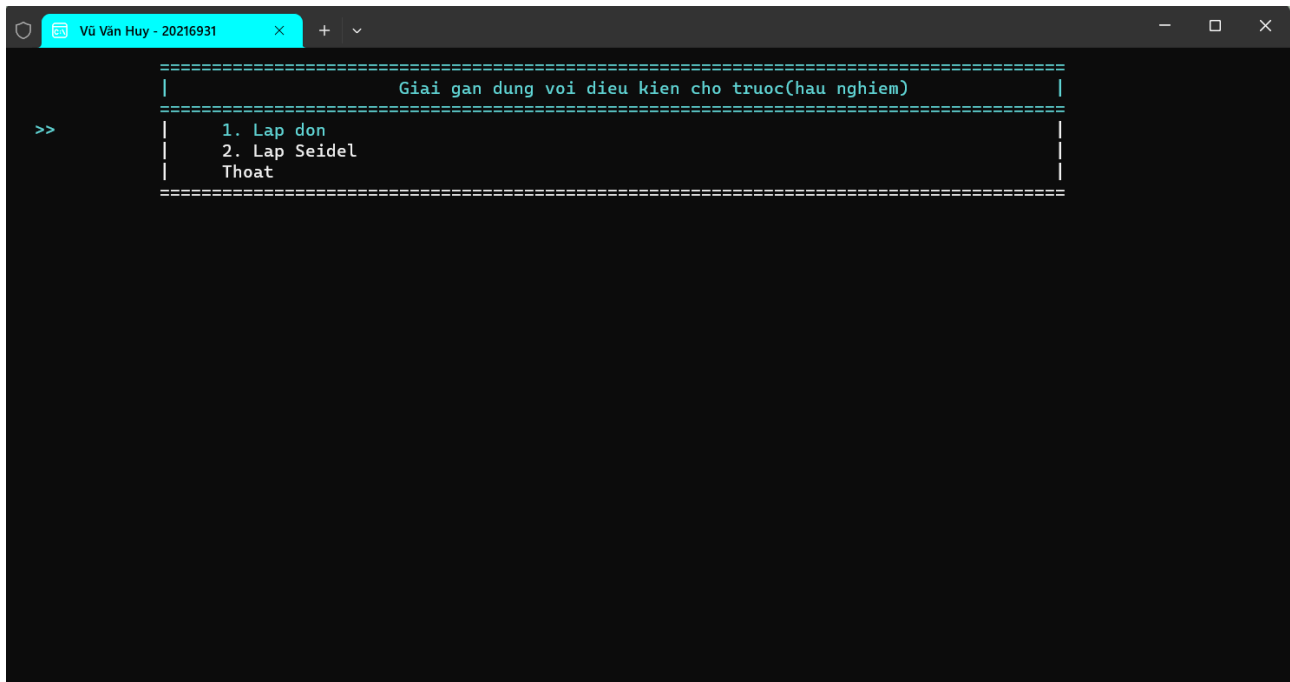
```
Lap Seidel voi sai so e cho truooc
Nhap vao sai so e: 0.0001
Nghiem cua he phuong trinh la:
Lan1
x[1] = 0.85
x[2] = 0.315
x[3] = 0.352
Lan2
x[1] = 0.8981
x[2] = 0.33979
x[3] = 0.342232
Lan3
x[1] = 0.897575
x[2] = 0.341796
x[3] = 0.341883
Lan4
x[1] = 0.897444
x[2] = 0.341879
x[3] = 0.34188
Da giai xong he phuong trinh trong 4 buoc lap.

Nhan phim bat ki de tiep tục...
```

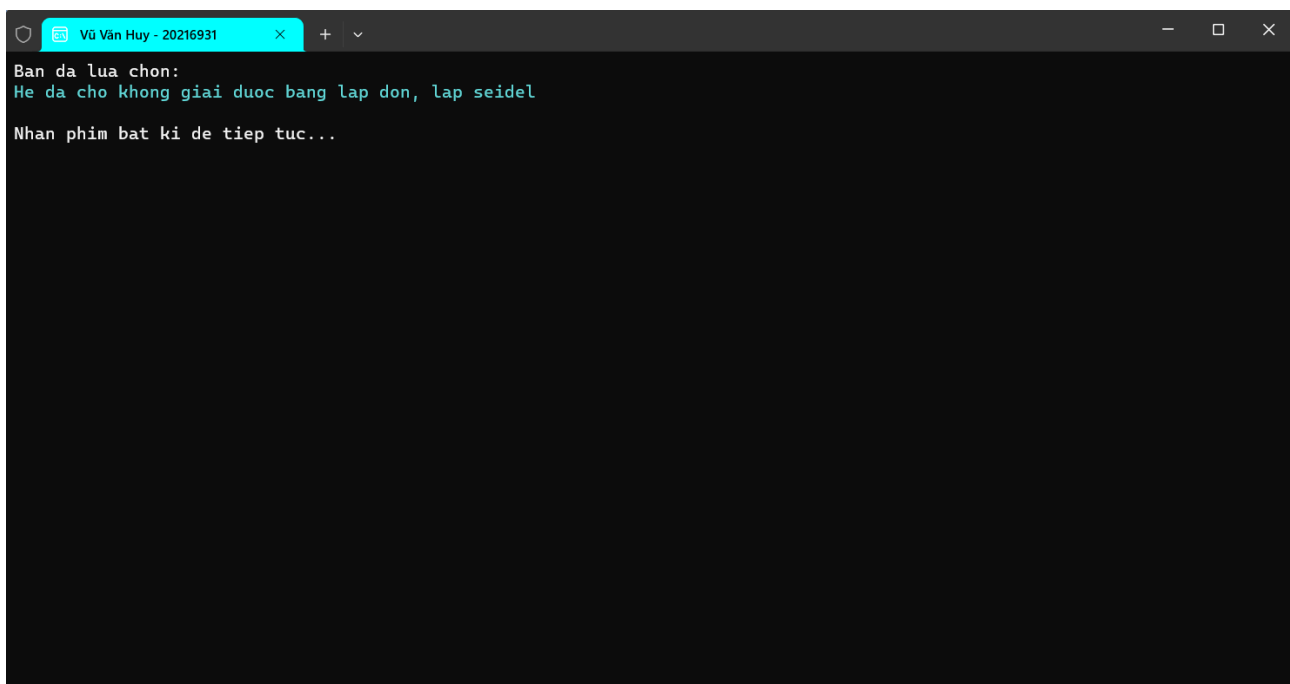
Hình 17: Lặp Seidel với sai số  $\epsilon$  cho trước



## 6.7 Tính nghiệm gần đúng thỏa mãn điều kiện



Hình 18: Menu con lặp với điều kiện



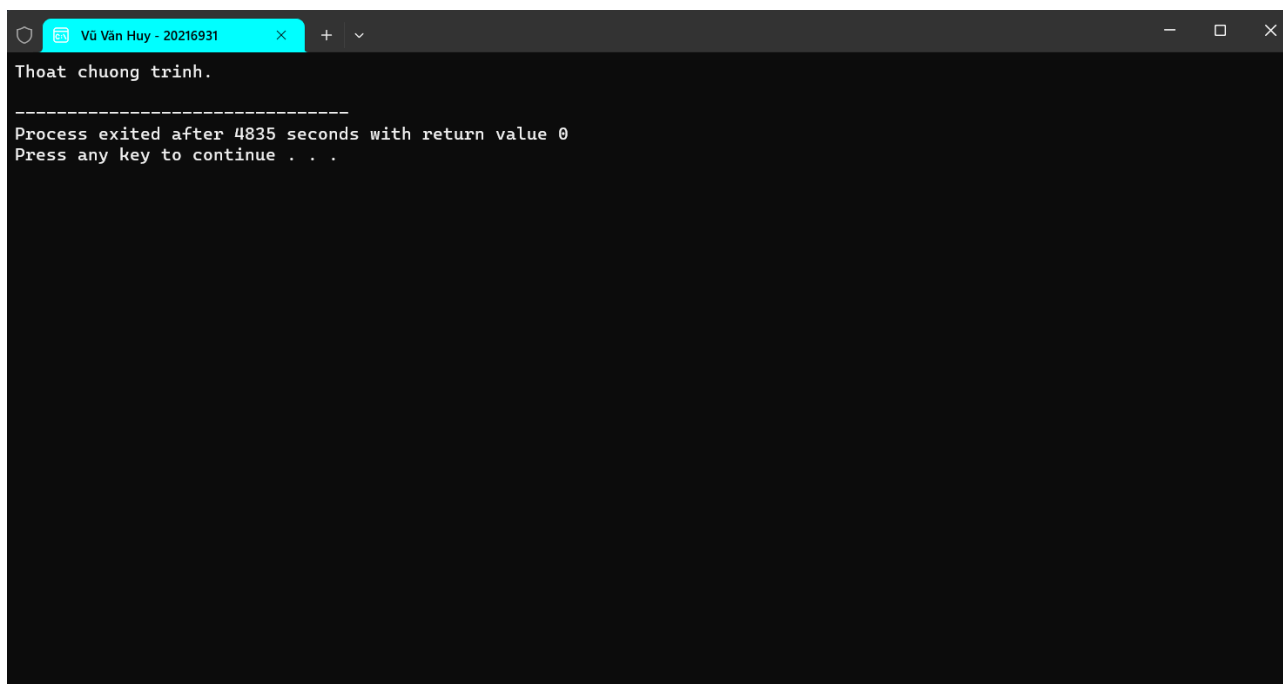
Hình 19: Hiện thị thông báo nếu phương pháp không hội tụ

```
Lap đơn co điều kiện với sai số e cho trước:  
Nhập vào sai số e: 0.001  
Lan 1  
0.85  
0.3  
0.3  
Lan 2  
0.91  
0.355  
0.355  
Lan 3  
0.8935  
0.338  
0.338  
Lan 4  
0.8986  
0.34305  
0.34305  
Lan 5  
0.897085  
0.34153  
0.34153  
Lan 6  
0.897541  
0.341985  
0.341985  
Đã giải xong hệ phương trình trong 6 bước lặp.  
Nhấn phím bất kỳ để tiếp tục...
```

Hình 20: Lặp đơn với điều kiện sai số

```
Lap Seidel co điều kiện với sai số e cho trước:  
Nhập vào sai số e: 0.001  
Nghiệm của hệ phương trình là:  
Lan1  
x[1] = 0.85  
x[2] = 0.315  
x[3] = 0.352  
Lan2  
x[1] = 0.8981  
x[2] = 0.33979  
x[3] = 0.342232  
Lan3  
x[1] = 0.897575  
x[2] = 0.341796  
x[3] = 0.341883  
Lan4  
x[1] = 0.897444  
x[2] = 0.341879  
x[3] = 0.34188  
Đã giải xong hệ phương trình trong 4 bước lặp.  
Nhấn phím bất kỳ để tiếp tục...
```

Hình 21: Lặp Seidel với điều kiện sai số sai số e



Hình 22: Kết thúc chương trình

## 6.8 Kết quả thu được trong file output.txt

Tất cả thao tác khi thực hiện chương trình đều được ghi vào file ouput.txt, dưới đây là nội dung file

```
1 Giai HPT tuyến tính  $AX = b$  bằng lap đơn, lap Seidel
2 Lua chọn 1:
3 Nhập từ file
4 -----
5 Ma trận B là:
6 0 -0.1 -0.2
7 -0.1 0 -0.2
8 -0.1 -0.2 0
9 vector d là:
10 1
11 0.5
12 0.5
13
14 -----
15 Lua chọn 2:
16 Kiểm tra tính chéo trội của A:
17 A chéo trội
18 -----
19 Do  $\|Q\| = 0.3$ 
20 Phương pháp hồi tu
21 -----
22 Lua chọn 3:
23 Chuẩn cột của A: 14
24 Chuẩn hàng của A: 13
25 Chuẩn cột của B: 0.4
26 Chuẩn hàng của B: 0.3
27 Chuẩn cột của vector d: 2
28 Chuẩn hàng của vector d: 1
29
30 -----
31 Lua chọn 4:
32 Lap đơn k lần:
33 Nhập vào số lần lap: 5 Nhập xấp xỉ đầu: x_0:
34 Chọn luôn xấp xỉ đầu là ma trận d.
35 Lan 1
36 0.85
37 0.3
38 0.3
39 Lan 2
40 0.91
41 0.355
42 0.355
43 Lan 3
44 0.8935
45 0.338
46 0.338
47 Lan 4
48 0.8986
49 0.34305
50 0.34305
51 Lan 5
```

```
52 0.897085
53 0.34153
54 0.34153
55
56 Danh gia sai so:
57 sai so: (q/(1-q))(X_5 - X_4) = 0.000651429
58 Lap Seidel k lan:
59 Nhap vao so lan lap: 5
60 Nghiem cua he phuong trinh la:
61 x[1] = 0.85
62 x[2] = 0.315
63 x[3] = 0.352
64 x[1] = 0.8981
65 x[2] = 0.33979
66 x[3] = 0.342232
67 x[1] = 0.897575
68 x[2] = 0.341796
69 x[3] = 0.341883
70 x[1] = 0.897444
71 x[2] = 0.341879
72 x[3] = 0.34188
73 x[1] = 0.897436
74 x[2] = 0.34188
75 x[3] = 0.34188
76
77 Danh gia sai so:
78 Sai so: (q/(1-q))(X_5 - X_4) = 3.25159e-006
79
80 -----
81 Lua chon 5:
82 Lua chon 5:
83 Lap don voi sai so e cho truoc:
84 Lan 1
85 0.85
86 0.3
87 0.3
88 Lan 2
89 0.91
90 0.355
91 0.355
92 Lan 3
93 0.8935
94 0.338
95 0.338
96 Lan 4
97 0.8986
98 0.34305
99 0.34305
100 Lan 5
101 0.897085
102 0.34153
103 0.34153
104 Lan 6
105 0.897541
106 0.341985
107 0.341985
108 Lan 7
109 0.897404
```

```
110 0.341849
111 0.341849
112 Da giai xong he phuong trinh trong 7 buoc lap.
113
114 Lap Seidel voi sai so e cho truooc:
115 Nhap vao sai so e: 0.0001
116 Nghiem cua he phuong trinh la:
117 x[1] = 0.85
118 x[2] = 0.315
119 x[3] = 0.352
120 x[1] = 0.8981
121 x[2] = 0.33979
122 x[3] = 0.342232
123 x[1] = 0.897575
124 x[2] = 0.341796
125 x[3] = 0.341883
126 x[1] = 0.897444
127 x[2] = 0.341879
128 x[3] = 0.34188
129 Da giai xong he phuong trinh trong 4 buoc lap.
130
131 -----
132 Lua chon 6:
133 Lap don co dieu kien voi sai so e cho truooc:
134 Lan 1
135 0.85
136 0.3
137 0.3
138 Lan 2
139 0.91
140 0.355
141 0.355
142 Lan 3
143 0.8935
144 0.338
145 0.338
146 Lan 4
147 0.8986
148 0.34305
149 0.34305
150 Lan 5
151 0.897085
152 0.34153
153 0.34153
154 Lan 6
155 0.897541
156 0.341985
157 0.341985
158 Da giai xong he phuong trinh trong 6 buoc lap.
159 Lap Seidel co dieu kien voi sai so e cho truooc:
160 Nghiem cua he phuong trinh la:
161 x[1] = 0.85
162 x[2] = 0.315
163 x[3] = 0.352
164 x[1] = 0.8981
165 x[2] = 0.33979
166 x[3] = 0.342232
167 x[1] = 0.897575
```

```
168 x[2] = 0.341796
169 x[3] = 0.341883
170 x[1] = 0.897444
171 x[2] = 0.341879
172 x[3] = 0.34188
173 Da giai xong he phuong trinh trong 4 buoc lap.
174
175 -----
```

→ Qua việc thực thi chương trình cho thấy phương pháp lặp Seidel hiệu quả hơn so với phương pháp lặp đơn trong việc tìm nghiệm của phương trình  $AX = B$ .

## 7 Tổng kết

### 7.1 Những điều đã học và làm được

1. Sử dụng và làm việc tốt với ngôn ngữ lập trình C++.
2. Áp dụng tốt các thuật toán giải gần đúng hệ phương trình tuyến tính.
3. Viết code thành thạo, sử lý in kết quả ra file.
4. Định hướng viết chương trình theo phương pháp tinh chỉnh từng bước.
5. Sửa lỗi chương trình khi viết code.
6. Áp dụng tốt các kiến thức được dạy trên lớp.
7. Sử dụng các công cụ mã nguồn mở để lưu trữ code.
8. Phần nào hiểu được tầm quan trọng giữa toán và tin.

### 7.2 Đánh giá chi tiết

Chương trình đơn giản dễ sử dụng với người dùng. Tuy nhiên chương trình vẫn còn khá dài và nhiều điểm chưa tối ưu. Cụ thể:

- Các menu con bị lặp lại 1 lần thay vì viết chung 1 vào 1 menu lớn.
- Khi nhập tay phải nhập lại kích thước ma trận 2 lần do chia làm nhiều hàm menu, truyền tham trị. Lỗi này có thể sửa được bằng cách gộp lại các hàm nhưng vẫn để nhằm mục đích chương trình chia thành các hàm con để dàng bảo trì nâng cấp phần mềm sau này.
- Thực hiện xử lý tuần tự đối với các hàm con.
- Code còn khá dài, nhiều đoạn menu con bị lặp lại chưa thể tối ưu bằng các chương trình con. Do gặp vấn đề về truyền tham trị.



### 7.3 Đánh giá chung

BẢNG ĐÁNH GIÁ CHUNG				
MÔN HỌC: KỸ THUẬT LẬP TRÌNH				
Họ và tên: VŨ VĂN HUY		-	Mã sinh viên: 20216931	
STT	Nội dung yêu cầu theo đề	Đã viết code? (Y/N)	Đã thực hiện đúng? (Y/N)	Tự đánh giá, nhận xét
1	Nhập vào ma trận A, vector b theo khuôn dạng của ma trận	Y	Y	Khi chọn nhập bằng tay thì phải nhập 2 lần
2	Kiểm tra tính chéo trội của ma trận A	Y	Y	Đúng yêu cầu
3	Tính chuẩn của ma trận A và kiểm tra sự hội tụ của phương pháp lặp đơn và lặp Seidel đối với hệ đã cho	Y	Y	Đúng yêu cầu
4	Tính nghiệm gần đúng với số lần lặp k cho trước và đánh giá sai số, sử dụng cả hai công thức	Y	Y	Báo lỗi khi ma trận không hội tụ
5	Tính nghiệm gần đúng với sai số e cho trước, sử dụng cả hai phương pháp áp dụng công thức	Y	Y	Báo lỗi khi ma trận không hội tụ
6	Tính nghiệm gần đúng $X^k$ thoả mãn: $\ X^k - X^{k-1}\  \leq e$ cho trước	Y	Y	Báo lỗi khi ma trận không hội tụ
7	Ghi vào tệp văn bản thể hiện quá trình thực hiện chương trình và các kết quả	Y	Y	Đúng yêu cầu
8	Thực hiện chương trình bằng menu điều khiển bằng các phím chức năng	Y	Y	Menu còn chưa đẹp, dài

Bảng 1: Đánh giá chung

Xem toàn bộ mã nguồn và mô tả tại: 