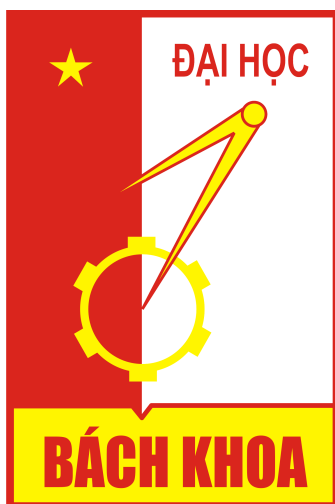


ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC



BÁO CÁO CUỐI KỲ

MÔN HỌC: KỸ THUẬT LẬP TRÌNH

CHỦ ĐỀ: XÂY DỰNG "THƯ VIỆN/ TIỆN ÍCH" VỀ MA TRẬN

Giảng viên hướng dẫn: TS. Vũ Thành Nam

Sinh viên thực hiện: Nguyễn Thị Linh Chi

MSSV: 20216913

Mã lớp: 142297

Hà Nội, 6/2023

Lời cảm ơn

Lời đầu tiên, em xin gửi lời cảm ơn đến Viện Toán ứng dụng và Tin học, Đại học Bách Khoa Hà Nội đã tạo cơ hội để em thực hiện báo cáo môn học trong một môi trường tốt nhất, nhằm đáp ứng yêu cầu hoàn thành báo cáo của em.

Đặc biệt, em xin bày tỏ lòng biết ơn sâu sắc đến thầy giáo TS. Vũ Thành Nam, người đã trực tiếp giảng dạy, hướng dẫn và hỗ trợ em rất nhiều ở môn học Kỹ thuật lập trình trong suốt thời gian qua.

Cuối cùng, em xin gửi lời cảm ơn tới các anh chị cựu sinh viên viện Toán ứng dụng và Tin học, các thầy cô giáo bộ môn liên quan, sự hỗ trợ, hợp tác của bạn bè đã giúp đỡ và truyền đạt những kinh nghiệm quý báu cho quá trình học tập của em.

Do kiến thức còn hạn chế nên báo cáo của em hoàn toàn không tránh khỏi những thiếu sót và sai sót. Vì vậy, em rất mong nhận được một số ý kiến đóng góp của thầy.

Trân trọng cảm ơn

Sinh viên báo cáo

Nguyễn Thị Linh Chi - 20216913

Mục lục

I. Tổng quan về chủ đề	3
1. Lý do chọn chủ đề	3
2. Giới thiệu chủ đề	3
II. Cấu trúc chương trình	4
1. Thiết kế và tổng quan cấu trúc chương trình	4
2. Phân tích thuật toán	5
III. Tổng hợp kỹ thuật lập trình (Chương trình tốt)	9
1. Yêu cầu	9
2. Thiết kế	9
3. Lập trình	9
4. Kiểm thử	10
IV. Kiểm thử và kết quả chạy chương trình	11
V. Tổng kết	26
1. Những điều đã làm được và học được	26
2. Khó khăn và hạn chế	26
3. Hướng phát triển chủ đề	26
VI. Tài liệu tham khảo	27

I Tổng quan về chủ đề

1. Lý do chọn chủ đề

Từ khi trở thành một sinh viên Bách Khoa, em đã có cơ hội học tập với nhiều môn học mới, đặc biệt là môn Đại số tuyến tính. Em nhận thấy, môn học có nhiều kiến thức bổ ích và tính ứng dụng cao. Bên cạnh đó, em cũng nhận ra rằng, khi thao tác với ma trận cỡ lớn nếu thực hiện thủ công như vậy thì sẽ rất mất thời gian, thậm chí là sai sót, và có thể không làm được. Và cũng thật may mắn, hiện tại em đang được học môn học Giải tích số, có cơ hội làm việc với ma trận rất nhiều và cũng biết thêm nhiều giải thuật để áp dụng cho ma trận.

Ngoài ra, em nhận thấy được, ma trận có ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau, từ khoa học máy tính đến thị giác máy tính và điều khiển tự động. Việc xây dựng một thư viện ma trận sẽ giúp cho việc tính toán và phân tích ma trận trở nên đơn giản và dễ dàng hơn, làm giảm đáng kể thời gian và công sức trong việc phân tích dữ liệu.

Vậy nên, em quyết định chọn chủ đề này vì tính ứng dụng và niềm yêu thích của bản thân. Em mong muốn có sự kết hợp kiến thức từ cả ba môn học: Đại số tuyến tính, Giải tích số và Kỹ thuật lập trình để xây dựng được một "thư viện/ tiện ích" về ma trận một cách tối ưu nhất.

2. Giới thiệu chủ đề

Chủ đề "Xây dựng thư viện/tiện ích về ma trận" là một chủ đề liên quan đến lý thuyết ma trận trong đại số tuyến tính. Trong đó, một số chủ đề cơ bản có thể bao gồm hệ thức ma trận, tính chất của ma trận, phép nhân ma trận, phép chuyển vị và toán tử đảo ngược ma trận. Các ứng dụng của ma trận rất rộng, từ lập trình máy tính đến kỹ thuật cơ khí và nhiều lĩnh vực khác. Việc xây dựng một thư viện hay tiện ích liên quan đến ma trận có thể giúp đơn giản hóa các tính toán và phân tích liên quan đến ma trận trong các dự án và nghiên cứu khác nhau.

Đối với những ai đã học đại số tuyến tính hoặc các môn liên quan đến toán học, ma trận là một khái niệm quen thuộc. Ma trận là một cách tiện lợi để biểu diễn, lưu trữ và tính toán các hệ thức tuyến tính và các quá trình trên các dữ liệu đa chiều.

Trong lý thuyết ma trận, các hệ thức ma trận cơ bản như ma trận đơn vị, ma trận nghịch đảo và ma trận chuyển vị có thể được sử dụng để giải các phương trình tuyến tính và các vấn đề liên quan đến phép toán trong các không gian đa chiều. Nhiều ứng dụng khác nhau của ma trận được sử dụng trong các lĩnh vực như nhận dạng hình ảnh, xử lý tín hiệu và máy học, mô hình hóa và mô phỏng, điều khiển tự động và nhiều lĩnh vực khác, bao gồm cả các mô hình học sâu phức tạp như mạng neural.

Cuối cùng, xây dựng một thư viện ma trận liên quan đến ma trận là một hoạt động thú vị và cũng là cách tuyệt vời để phát triển kỹ năng lập trình và toán học của bản thân.

II Cấu trúc chương trình

1. Thiết kế và tổng quan cấu trúc chương trình

• Chương trình thư viện ma trận được thiết kế thành 3 file mã nguồn.

- File mã nguồn matrix.py cài đặt ADT ma trận.
- File mã nguồn my_math.py bao gồm các thủ tục/ hàm con. Mỗi chương trình con tương ứng với một chức năng/ phép toán liên quan đến ma trận.
- File mã nguồn main.py là hàm main chính, nơi thực thi các yêu cầu mong muốn.

File mã nguồn matrix.py bao gồm các phần cài đặt chính như sau:

- ADT ma trận, kích thước ma trận.
- Hiển thị ma trận.
- Tạo ma trận đơn vị/ ma trận không.
- Ma trận chuyển vị.
- Phương thức gán giá trị cho một đối tượng.
- Phương thức truy xuất giá trị tại một vị trí.

File mã nguồn my_math.py bao gồm các thủ tục/ hàm con sau:

- Nhập ma trận từ bàn phím.
- Chuẩn hóa số
- Kiểm tra input từ file có là ma trận hay không?
- Ghi kết quả vào file.
- Đọc ma trận từ file.
- Nhân ma trận với 1 số.
- Phép cộng/ trừ/ nhân 2 ma trận.
- Tính định thức ma trận.
- Tạo ma trận bổ sung
- Chọn phần tử khử và khử ma trận.
- Tìm ma trận nghịch đảo.
- Tìm ma trận chuyển vị.

File mã nguồn main.py bao gồm 3 phần chính:

- Bảng menu gồm 9 chức năng và 1 lựa chọn thoát chương trình.
- Sử dụng lời gọi từ điển trống matrices để lưu trữ ma trận đầu vào. Khi thực hiện các chức năng, sẽ lấy các ma trận từ từ điển.
- Input đầu vào để chọn chức năng.
- Các thao tác cần thực hiện đối với một chức năng sau khi đã chọn chức năng đó.

2. Phân tích thuật toán.

a) Chức năng nhập ma trận từ bàn phím

- Sử dụng 2 cấu trúc vòng lặp While để nhập m, n đến khi thỏa mãn m, n nguyên và $m, n \geq 1$ thì dừng.
- Dùng cấu trúc vòng lặp for để nhập vào ma trận. Nhập từng hàng của ma trận và trả về ma trận.

b) Kiểm tra input từ file có là ma trận hay không?

- Đầu vào là dữ liệu cần kiểm tra. Hàm `len(matrix[0])` tính chiều dài hàng 0.
- Dùng cấu trúc `list(map())`: Duyệt các phần tử của các hàng của ma trận. Điều kiện kiểm tra: chiều dài các hàng có bằng `dim1` hay không? Nếu đúng trả về `True`, sai trả về `False`.
- Kết thúc hàm kiểm tra trả về `mask = [True, False ...]` ứng với kết quả kiểm tra ở các hàng.

c) Đọc ma trận từ file

- Sử dụng cấu trúc with open(path) as f, để mở file và đọc dữ liệu từ file.
- Kiểm tra xem input vừa đọc có là ma trận hay không?

d) Nhân ma trận với một số

- Đầu vào là ma trận A và số thực k.
- Dùng hàm shape để tính kích thước A.
- Khởi tạo ma trận kết quả là ma trận 0 kích thước bằng kích thước của A.
- Sử dụng 2 vòng for, duyệt hết các phần tử của A. Tại mỗi phần tử được duyệt qua, lấy phần tử đó nhân với k và gán vào ma trận kết quả.
- Trả về ma trận kết quả.

e) Cộng/ Trừ 2 ma trận

- Đầu vào là 2 ma trận A,B.
- Kiểm tra A và B có là 2 ma trận cùng cỡ hay không?
- Nếu có thì khởi tạo ma trận kết quả là ma trận 0, cùng cỡ với 2 ma trận đầu vào.
- Dùng 2 vòng for và duyệt hết các phần tử của A, B. Tại mỗi lần duyệt, gán cho phần tử của ma trận kết quả là tổng (hiệu) 2 phần tử trong ma trận A, B. Trả về ma trận kết quả.
- Ngược lại, nếu không cùng kích thước, báo lỗi "Hai ma trận không cùng cỡ".

f) Nhân 2 ma trận

- Đầu vào là 2 ma trận A, B.
- Tính kích thước 2 ma trận A, B.
- Kiểm tra nếu số cột của A bằng số hàng của B thì phép nhân được thực hiện. Ngược lại, báo lỗi "Kích thước 2 ma trận không phù hợp".

g) Tính định thức ma trận

- Kiểm tra đầu vào, nếu là ma trận vuông thì thực hiện phép tính định thức

- Giải thuật áp dụng: Phương pháp Gauss. Đưa ma trận về ma trận tam giác trên. Định thức của ma trận là tích của các phần tử nằm ở đường chéo chính.
- Cách thực hiện Phương pháp Gauss: Tìm phần tử đầu tiên khác 0 ở mỗi cột, nếu nó không nằm ở trên đường chéo chính thì đổi 2 hàng i và j , và thay đổi dấu của định thức cần tính.
- Thực hiện phép khử để đưa về ma trận tam giác trên nếu tất cả các phần tử của đường chéo chính khác 0. Ngược lại, trả về $\det = 0$.
- Duyệt qua tất cả phần tử của đường chéo chính, trả về định thức là tích của các phần tử nằm trên đường chéo chính.

h) Nối 2 ma trận (Tạo ma trận bổ sung A/I)

- Đầu vào là 2 ma trận A, B. Tính kích thước 2 ma trận.
- Kiểm tra xem axis = 0 hoặc 1 hay không. Nếu không, ngoại lệ ValueError được thông báo "Axis must be 0 or 1".
- Nếu axis= 1, kiểm tra xem số hàng của A và B có bằng nhau hay không. Nếu không bằng nhau, ngoại lệ ValueError được thông báo "Dimension mismatch". Ngược lại, gán $m = m_1, n = n_1 + n_2$.
- Ngược lại, nếu axis = 0, kiểm tra xem số cột của A và B có bằng nhau hay không. Nếu không bằng nhau, ngoại lệ ValueError được thông báo "Dimension mismatch". Ngược lại, gán $m = m_1 + m_2, n = n_1$.
- Tạo ma trận result (mxn) là ma trận không.
- Sử dụng vòng lặp for để gán giá trị cho result bằng giá trị tương ứng trên A, B.

i) Chọn phần tử khử và khử ma trận.

- Đầu vào là ma trận A và 2 mảng để đánh dấu phần tử được chọn.
- Phép chọn phần tử khử: Ưu tiên phần tử có giá trị bằng 1 hoặc -1. Nếu không có thì tìm phần tử có trị tuyệt đối lớn nhất (max). Phần tử được chọn, không trùng hàng và cột với các phần tử được chọn trước đó. Điều kiện dừng: Không còn phần tử nào để chọn hoặc phần tử max = 0.
- Hàng, cột có phần tử được chọn rồi thì đánh dấu là True.

- Phép khử: Với mỗi phần tử khử được chọn, thực hiện phép khử của cột đó, đưa các phần tử còn lại trong cột về bằng 0.

j) Tìm ma trận nghịch đảo

- Kiểm tra tính vuông của ma trận A. Nếu không vuông thì báo lỗi "Ma trận không vuông".
- Kiểm tra định thức của ma trận A. Nếu định thức ma trận bằng 0, báo lỗi "Ma trận không khả nghịch".
- Giải thuật áp dụng: Phương pháp Gauss - Jordan. Bao gồm 2 bước chính: Chọn phần tử khử và phép khử.
 - Dùng thuật toán chọn phần tử khử và thực hiện phép khử (mục i).
 - Phép chọn phần tử khử thực hiện trên ma trận A.
 - Phép khử thực hiện trên ma trận bổ sung của A $[A|I]$.
- Sau khi thực hiện xong phép chọn và khử, thực hiện đổi vị trí các hàng $[A|I]$ để đưa ma trận A về ma trận đường chéo. Chia mỗi hàng cho giá trị tại phần tử đường chéo. Ma trận A trở thành ma trận đơn vị.
- Ma trận I bây giờ trở thành ma trận nghịch đảo của A.
- Trả về ma trận nghịch đảo.

k) Tìm ma trận chuyển vị

- Đầu vào là ma trận A.
- Tính kích thước ma trận A($m \times n$).
- Tạo ma trận không là ma trận kết quả kích thước ($n \times m$).
- Sử dụng 2 vòng lặp for, gán cho phần tử vị trí $[j][i]$ ở ma trận kết quả bằng giá trị phần tử $A([i][j])$.
- Trả về ma trận chuyển vị.

III Tổng hợp kỹ thuật lập trình (Chương trình tốt)

1. Yêu cầu

- Xác định và đáp ứng được yêu cầu xây dựng "thư viện/ tiện ích" ma trận.
- Xác định và đáp ứng được các vấn đề cần giải quyết, các chức năng: Đọc file, ghi vào file, cộng trừ 2 ma trận, nhân ma trận với 1 số, tính định thức, tìm ma trận nghịch đảo...

2. Thiết kế

- Tiếp cận bài toán theo hướng Modun hóa - Modularity.
- Chia nhỏ bài toán thành nhiều bài toán nhỏ. Mỗi bài toán nhỏ ở đây có thể xem là một chức năng của ma trận. Gộp các bài toán nhỏ thành bài toán xây dựng "thư viện/ tiện ích" ma trận.

3. Lập trình

- **Giải thuật:** Lựa chọn giải thuật tối ưu: Phương pháp Gauss cho tính định thức, phương pháp Gauss - Jordan tìm ma trận nghịch đảo...
- **Cấu trúc mã nguồn:**
 - Chia mã nguồn thành 3 file gồm file cấu trúc ma trận, file chức năng/ thuật toán và file hàm main chính. Giúp chỉnh sửa mã nguồn dễ dàng, trình bày mã nguồn rõ ràng, dễ nâng cấp, người dùng dễ nắm bắt mã nguồn và tránh việc một file chứa mã nguồn quá dài.
 - File mã nguồn chức năng/ thuật toán: Mỗi hàm con có một nhiệm vụ riêng, để bảo trì mã nguồn. Ví dụ: `def sum_matrix(A,B)`: cộng 2 ma trận, `multiply_with_scalar(A,k)`: Nhân ma trận với 1 số, `det_matrix(A)`: tính định thức ma trận ...
 - Cách lề, sử dụng khoảng trắng phù hợp. Phân chia các phần khác nhau trong mã nguồn, như các hàm, lệnh điều kiện, vòng lặp, và các khối lệnh. Các điều đó làm cho mã nguồn trong sáng, dễ nhìn.

- Cách đoạn: Trong file chức năng/ thuật toán hoặc file matrix.py, hết mỗi chức năng sẽ cách đoạn. Trong file hàm main chính, hết mỗi case lựa chọn sẽ có cách đoạn để phân biệt.
- **Trình bày code:**
 - Đặt tên: Đặt tên các hàm bằng tiếng anh, tên đặc tả được chức năng của mỗi hàm (subtraction_matrix, multiply_matrix, det_matrix ...) và nhất quán trong việc đặt tên hàm. Đặt tên các biến đặc tả được ý nghĩa của biến, ví dụ: rows, cols, data, det...
 - Chú thích: Trước mỗi hàm chức năng, đều có chú thích tên của chức năng. Trong các hàm chức năng, cũng được chú thích một số chỗ có thể gây khó hiểu.
- **Ngoài ra, cũng áp dụng được một số quy tắc như sau:**
 - Số tham số đầu vào mỗi hàm là tối thiểu, thường chỉ có đầu vào là 1 - 2 ma trận, và thêm 1 - 2 tham số phụ.
 - Ngoài mỗi chức năng, mã nguồn ghi kết quả vào file được lặp đi lặp lại ở nhiều case lựa chọn. Vậy nên, khởi tạo 1 hàm con để ghi kết quả vào file.
 - Ở các phép toán nhân với 1 số, cộng/ trừ / nhân 2 ma trận, đều cần tạo một ma trận không lưu kết quả. Vậy nên có một hàm con để tạo dữ liệu.
 - Với mỗi hàm chức năng, thay vì khởi tạo trước mảng kết quả gây tốn bộ nhớ, thì kiểm tra trước điều kiện thực hiện chức năng. Sau khi thỏa mãn điều kiện thì mới khởi tạo mảng kết quả.
 - Trong mỗi case lựa chọn, với mỗi lỗi xảy ra, sẽ báo cho người dùng biết lỗi để người dùng thực hiện lại chương trình.

4. Kiểm thử

- Thực hiện kiểm thử với mỗi hàm con, đảm bảo tính đúng đắn giải thuật và không có lỗi logic. Kiểm thử các trường hợp đặc biệt, ví dụ: Dữ liệu đầu vào lỗi, không đúng dạng ma trận, tính định thức đối với ma trận không vuông, cộng/ trừ 2 ma trận không cùng kích thước... để kiểm tra việc thông báo lỗi của chương trình.
- Kiểm thử toàn bộ chương trình và áp dụng với đầu vào kích thước lớn, tốc độ và độ chính xác vẫn được đảm bảo.

IV Kiểm thử và kết quả chạy

1. Chọn chức năng

Trường hợp chọn chức năng:

Nhập số đầu vào không nằm trong Menu.

```
*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                 **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình            **
*****

Chọn chức năng: 10
No match found
```

Nhập 1 số nằm trong Menu:

```
*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                 **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình            **
*****

Chọn chức năng: 1
Nhập tên của ma trận: |
```

2. Thoát chương trình

- Nhập phím 0 để thoát chương trình.

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                 **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình            **
*****

Chọn chức năng: 0
Đã thoát.
PS C:\Users\pc\Downloads\Project_KTLT>

```

3. Nhập đầu vào và in ra ma trận

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                 **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình            **
*****

Chọn chức năng: 1
Nhập tên của ma trận: A
Nhập số hàng: 4
Nhập số cột: 5
Nhập ma trận:
2 3 5 6 2
1 0 2 3 6
10 5 2 3 1
-5 3 -5 0 8

[ 2.0  3.0  5.0  6.0  2.0 ]
[ 1.0  0.0  2.0  3.0  6.0 ]
[ 10.0 5.0  2.0  3.0  1.0 ]
[ -5.0 3.0 -5.0  0.0  8.0 ]

```

```

Chọn chức năng: 1
Nhập tên của ma trận: K
Nhập số hàng: 3
Nhập số cột: 4
Nhập ma trận:
1 2 3 4
0 2 5 7
-10 5 -3 2


$$\begin{bmatrix} 1.0 & 2.0 & 3.0 & 4.0 \\ 0.0 & 2.0 & 5.0 & 7.0 \\ -10.0 & 5.0 & -3.0 & 2.0 \end{bmatrix}$$


```

- Nhập đầu vào không đúng, chương trình báo lỗi.

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình            **
*****

Chọn chức năng: 1
Nhập tên của ma trận: D
Nhập số hàng: 3
Nhập số cột: 2
Nhập ma trận:
1 2
3
4 5
Error: Incorrect input

```

- Nhập số hàng không đúng, yêu cầu nhập lại:

```

Chọn chức năng: 1
Nhập tên của ma trận: L
Nhập số hàng: -2
Nhập lại số hàng:
Nhập số hàng: -6
Nhập lại số hàng:
Nhập số hàng: 

```

4. Đọc ma trận từ file

- Đọc file F.txt(Không đúng dạng ma trận):

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím                                         **
*      2. Nhập ma trận từ file                                             **
*      3. Nhân ma trận với số thực                                         **
*      4. Cộng 2 ma trận                                                  **
*      5. Trừ 2 ma trận                                                    **
*      6. Nhân 2 ma trận                                                  **
*      7. Tính định thức                                                  **
*      8. Tìm ma trận nghịch đảo                                          **
*      9. Tìm ma trận chuyển vị                                           **
*      0. Thoát chương trình                                              **
*****

Chọn chức năng: 2
Nhập tên ma trận: F
Nhập tên file: F.txt
The file does not contain a matrix.
None

```

- Đọc file B.txt:

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím                                         **
*      2. Nhập ma trận từ file                                             **
*      3. Nhân ma trận với số thực                                         **
*      4. Cộng 2 ma trận                                                  **
*      5. Trừ 2 ma trận                                                    **
*      6. Nhân 2 ma trận                                                  **
*      7. Tính định thức                                                  **
*      8. Tìm ma trận nghịch đảo                                          **
*      9. Tìm ma trận chuyển vị                                           **
*      0. Thoát chương trình                                              **
*****

Chọn chức năng: 2
Nhập tên ma trận: B
Nhập tên file: B.txt

[ 153.0  -21.0   12.0    5.0   -7.0 ]
[ 12.0  -258.0  -31.0   15.0    9.0 ]
[  0.0   15.0  108.0   25.0   19.0 ]
[ 14.0   -1.0    0.0  155.0   32.0 ]
[ 21.0   13.0  -25.0    6.0  298.0 ]

```

- Đọc file C.txt:

```
Chọn chức năng: 2
Nhập tên ma trận: C
Nhập tên file: C.txt
```

$$\begin{bmatrix} 3.0 & 2.0 & 5.0 & -7.0 & 4.0 \\ 11.0 & 3.0 & 2.0 & -3.0 & 2.0 \\ -5.0 & 6.0 & 0.0 & 2.0 & 14.0 \\ -4.0 & -7.0 & 2.0 & 3.0 & 10.0 \\ 3.0 & 2.0 & 4.0 & 0.0 & -4.0 \end{bmatrix}$$

- Đọc file D.txt:

```
Chọn chức năng: 2
Nhập tên ma trận: D
Nhập tên file: D.txt
```

$$\begin{bmatrix} 1.0 & 2.0 & 3.0 & 4.0 \\ 5.0 & 6.0 & 0.0 & -2.0 \\ 1.0 & 2.0 & 5.0 & 10.0 \end{bmatrix}$$

- Đọc file E.txt:

```
Chọn chức năng: 2
Nhập tên ma trận: E
Nhập tên file: E.txt
```

$$\begin{bmatrix} 130.75 & 3.75 & 5.25 & 11.25 & 18.25 & 6.5 & 12.0 & 15.5 & 2.75 & 20.0 \\ -7.0 & 156.0 & 4.0 & 8.0 & 17.6667 & -3.6667 & -3.0 & 4.0 & 9.0 & 21.6667 \\ 2.8 & 4.2 & 89.0 & 5.4 & 10.8 & 12.6 & -7.8 & 8.0 & 14.8 & 13.0 \\ 16.5714 & 14.8571 & 25.4286 & 140.4286 & 25.7143 & 2.1429 & 5.5714 & 10.5714 & 5.4286 & 6.4286 \\ 6.1667 & -19.0 & 9.8333 & 12.1667 & 114.0 & -6.1667 & 14.5 & 3.6667 & -10.5 & -14.1667 \\ -0.875 & 11.125 & 7.125 & 6.375 & 13.375 & 81.75 & -6.0 & 1.875 & -15.75 & -8.75 \\ -7.5714 & -1.4286 & 5.2857 & 9.8571 & -18.0 & 7.1429 & 96.8571 & -10.1429 & -8.8571 & -12.8571 \\ 17.2 & 11.0 & 15.6 & 2.8 & 2.2 & 14.2 & 15.6 & 106.8 & 5.2 & 6.4 \\ 15.8333 & 4.8333 & -5.8333 & 5.5 & -9.6667 & 7.6667 & 5.6667 & 9.5 & 81.1667 & -3.0 \\ 0.9167 & -2.0 & -2.5833 & 2.9167 & 3.9167 & 10.6667 & 2.1667 & 9.5833 & -3.5833 & 52.0833 \end{bmatrix}$$

- Đọc file E1.txt:

```
Chọn chức năng: 2
Nhập tên ma trận: E1
Nhập tên file: E1.txt
```

$$\begin{bmatrix} 1.0 & 3.0 & 5.0 & 1.0 & 10.0 & -2.0 & -9.0 & 5.0 & 1.0 & 15.0 \\ 3.0 & 25.5 & 30.5 & -10.0 & 123.0 & 2.0 & 8.0 & 16.0 & -21.0 & 12.0 \\ 4.0 & 28.5 & 35.5 & -9.0 & 133.0 & 0.0 & -1.0 & 21.0 & -20.0 & 27.0 \\ 2.0 & 22.5 & 25.5 & -11.0 & 113.0 & 4.0 & 17.0 & 11.0 & -22.0 & -3.0 \\ 5.0 & 48.0 & 56.0 & -21.0 & 236.0 & 6.0 & 25.0 & 27.0 & -43.0 & 9.0 \\ -4.0 & -28.5 & -35.5 & 9.0 & -133.0 & 0.0 & 1.0 & -21.0 & 20.0 & -27.0 \\ 2.0 & 45.0 & 53.0 & -24.0 & 233.0 & 3.0 & 22.0 & 24.0 & -46.0 & 6.0 \\ -2.0 & 16.5 & 17.5 & -15.0 & 100.0 & 3.0 & 23.0 & 3.0 & -26.0 & -21.0 \\ -1.0 & -3.0 & -5.0 & -1.0 & -10.0 & 2.0 & 9.0 & -5.0 & -1.0 & -15.0 \\ 1.0 & 42.0 & 48.0 & -25.0 & 223.0 & 5.0 & 31.0 & 19.0 & -47.0 & -9.0 \end{bmatrix}$$

5. Nhân ma trận với 1 số

- Trường hợp ma trận chưa được khởi tạo trước đó:

```

*****
**                                     **
**                                     MENU                                     **
**                                     **
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình           **
*****

Chọn chức năng: 3
Nhập tên của ma trận: I
Nhập số thực k: 10
Ma trận không tồn tại

```

- Thực hiện nhân ma trận B với 1 số:

```

*****
**                                     MENU                                     **
**                                     **
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình           **
*****

Chọn chức năng: 3
Nhập tên của ma trận: B
Nhập số thực k: 5

```

- Kết quả được ghi ra file matrix.out (Mở xem file bằng Visual Studio code):

```

15  Kết quả 5.0*B:
16  [
17  | 765.0  -105.0   60.0   25.0  -35.0 |
18  | 60.0  -1290.0 -155.0   75.0   45.0 |
19  |  0.0   75.0   540.0  125.0   95.0 |
20  | 70.0   -5.0    0.0  775.0  160.0 |
21  | 105.0   65.0 -125.0   30.0 1490.0 |
22  |

```

- Thực hiện nhân ma trận E với một số:

```

*****
**                                     **
**                                     MENU                                     **
**                                     **
**      1. Nhập ma trận từ bàn phím      **
**      2. Nhập ma trận từ file          **
**      3. Nhân ma trận với số thực      **
**      4. Cộng 2 ma trận                **
**      5. Trừ 2 ma trận                 **
**      6. Nhân 2 ma trận                **
**      7. Tính định thức                **
**      8. Tìm ma trận nghịch đảo        **
**      9. Tìm ma trận chuyển vị        **
**     0. Thoát chương trình            **
*****

Chọn chức năng: 3
Nhập tên của ma trận: E
Nhập số thực k: 12

```

- Kết quả được ghi ra file matrix.out (Mở xem file bằng Visual Studio code):

```

matrix.py  my_math.py  main.py  matrix.out X
C: > Users > pc > Downloads > Project_KTLT > matrix.out
1
2  Kết quả 12.0*E:
3
4  [
5    1569.0    45.0    63.0    135.0    219.0    78.0    144.0    186.0    33.0    240.0
6    -84.0    1872.0    48.0    96.0    212.0004    -44.0004    -36.0    48.0    108.0    260.0004
7    33.6    50.4    1068.0    64.8    129.6    151.2    -93.6    96.0    177.6    156.0
8    198.8568    178.2852    305.1432    1685.1432    308.5716    25.7148    66.8568    126.8568    65.1432    77.1432
9    74.0004    -228.0    117.9996    146.0004    1368.0    -74.0004    174.0    44.0004    -126.0    -170.0004
10   -10.5    133.5    85.5    76.5    160.5    981.0    -72.0    22.5    -189.0    -105.0
11   -90.8568    -17.1432    63.4284    118.2852    -216.0    85.7148    1162.2852    -121.7148    -106.2852    -154.2852
12    206.4    132.0    187.2    33.6    26.4    170.4    187.2    1281.6    62.4    76.8
13   189.9996    57.9996    -69.9996    66.0    -116.0004    92.0004    68.0004    114.0    974.0004    -36.0
14   11.0004    -24.0    -30.9996    35.0004    47.0004    128.0004    26.0004    114.9996    -42.9996    624.9996
15  ]

```

6. Cộng 2 ma trận

- Có ít nhất một đầu vào chưa được nhập từ file hoặc từ bàn phím trước đó:

```
Chọn chức năng: 4
Nhập tên của ma trận 1: B
Nhập tên của ma trận 2: J
Error: Input is invalid.
```

- Cộng 2 ma trận A, B không cùng cỡ:

```
*****
**                                     **
**                                     MENU                                     **
**                                     **
** 1. Nhập ma trận từ bàn phím          **
** 2. Nhập ma trận từ file              **
** 3. Nhân ma trận với số thực          **
** 4. Cộng 2 ma trận                    **
** 5. Trừ 2 ma trận                     **
** 6. Nhân 2 ma trận                    **
** 7. Tính định thức                    **
** 8. Tìm ma trận nghịch đảo            **
** 9. Tìm ma trận chuyển vị            **
** 0. Thoát chương trình                **
**                                     **
*****

Chọn chức năng: 4
Nhập tên của ma trận 1: A
Nhập tên của ma trận 2: B
Error: Matrix size mismatch.
```

- Cộng 2 ma trận B, C và kết quả:

```
Chọn chức năng: 4
Nhập tên của ma trận 1: B
Nhập tên của ma trận 2: C
```

```
23  Kết quả B + C:
24  [
25    156.0  -19.0   17.0   -2.0   -3.0
26    23.0  -255.0  -29.0   12.0   11.0
27    -5.0   21.0  108.0   27.0   33.0
28    10.0   -8.0    2.0  158.0   42.0
29    24.0   15.0  -21.0    6.0  294.0
30  ]
```

7. Trừ 2 ma trận

- Có ít nhất một đầu vào chưa được nhập từ file hoặc từ bàn phím trước đó:

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình            **
*****

Chọn chức năng: 5
Nhập tên của ma trận 1: A
Nhập tên của ma trận 2: G
Error: Input is invalid.

```

- Trường hợp 2 ma trận không cùng cỡ:

```

Chọn chức năng: 5
Nhập tên của ma trận 1: A
Nhập tên của ma trận 2: B
Error: Matrix size mismatch.

```

- Trừ 2 ma trận B, C và kết quả (Kết quả được ghi tiếp vào file matrix.out):

```

Chọn chức năng: 5
Nhập tên của ma trận 1: B
Nhập tên của ma trận 2: C

```

```

47  Kết quả B - C:
48  [
49      150.0  -23.0   7.0   12.0  -11.0
50      1.0  -261.0  -33.0   18.0   7.0
51      5.0   9.0  108.0   23.0   5.0
52      18.0   6.0   -2.0  152.0  22.0
53      18.0  11.0  -29.0   6.0  302.0
54  ]

```

- Trừ 2 ma trận E, E1 và kết quả được ghi tiếp vào file matrix.out:

```

Chọn chức năng: 5
Nhập tên của ma trận 1: E
Nhập tên của ma trận 2: E1

```

55	Kết quả E - E1:									
56										
57	129.75	0.75	0.25	10.25	8.25	8.5	21.0	10.5	1.75	5.0
58	-10.0	130.5	-26.5	18.0	-105.3333	-5.6667	-11.0	-12.0	30.0	9.6667
59	-1.2	-24.3	53.5	14.4	-122.2	12.6	-6.8	-13.0	34.8	-14.0
60	14.5714	-7.6429	-0.0714	151.4286	-87.2857	-1.8571	-11.4286	-0.4286	27.4286	9.4286
61	1.1667	-67.0	-46.1667	33.1667	-122.0	-12.1667	-10.5	-23.3333	32.5	-23.1667
62	3.125	39.625	42.625	-2.625	146.375	81.75	-7.0	22.875	-35.75	18.25
63	-9.5714	-46.4286	-47.7143	33.8571	-251.0	4.1429	74.8571	-34.1429	37.1429	-18.8571
64	19.2	-5.5	-1.9	17.8	-97.8	11.2	-7.4	103.8	31.2	27.4
65	16.8333	7.8333	-0.8333	6.5	0.3333	5.6667	-3.3333	14.5	82.1667	12.0
66	-0.0833	-44.0	-50.5833	27.9167	-219.0833	5.6667	-28.8333	-9.4167	43.4167	61.0833
67										

8. Nhân 2 ma trận

- Có ít nhất một đầu vào chưa được nhập từ file hoặc từ bàn phím trước đó:

```

*****
**                                     **
**                                     MENU                                     **
**                                     **
**      1. Nhập ma trận từ bàn phím      **
**      2. Nhập ma trận từ file          **
**      3. Nhân ma trận với số thực      **
**      4. Cộng 2 ma trận                **
**      5. Trừ 2 ma trận                 **
**      6. Nhân 2 ma trận                **
**      7. Tính định thức                **
**      8. Tìm ma trận nghịch đảo        **
**      9. Tìm ma trận chuyển vị         **
**      0. Thoát chương trình            **
*****
Chọn chức năng: 5
Nhập tên của ma trận 1: A
Nhập tên của ma trận 2: G
Error: Input is invalid.

```

- Nhân 2 ma trận B, D (Số cột của B khác số hàng của D):

```

*****
**                                     MENU                                     **
**                                     **
**      1. Nhập ma trận từ bàn phím      **
**      2. Nhập ma trận từ file          **
**      3. Nhân ma trận với số thực      **
**      4. Cộng 2 ma trận                **
**      5. Trừ 2 ma trận                 **
**      6. Nhân 2 ma trận                **
**      7. Tính định thức                **
**      8. Tìm ma trận nghịch đảo        **
**      9. Tìm ma trận chuyển vị         **
**      0. Thoát chương trình            **
*****
Chọn chức năng: 6
Nhập tên của ma trận 1: B
Nhập tên của ma trận 2: D
Error: Dimension mismatch. Two matrices have incompatible sizes.

```

- Kết quả nhân 2 ma trận A,B:

```

Chọn chức năng: 6
Nhập tên của ma trận 1: A
Nhập tên của ma trận 2: B

```

```

68  Kết quả A * B:
69  [
70    468.0  -721.0  421.0  1122.0  896.0
71    321.0   84.0   78.0   556.0 1915.0
72    1653.0 -1460.0  156.0   646.0  407.0
73    -561.0  -640.0 -893.0   -57.0 2351.0
74  ]

```

- Kết quả nhân 2 ma trận E, E1:

```

Chọn chức năng: 6
Nhập tên của ma trận 1: E
Nhập tên của ma trận 2: E1

```

```

75  Kết quả E * E1:
76  [
77    261.0  3208.875  3886.125  -1425.75  15959.25  88.5  767.25  2004.75  -2853.0  1628.25
78    594.67  6017.5  7094.51  -2633.67  29653.6864  593.3337  2659.67  3493.67  -5365.33  1677.0009
79    352.4  3215.4  3873.4  -1285.6  15598.4  184.4  842.4  2046.4  -2671.6  1682.4
80    554.7142  6164.9991  7145.2845  -2884.714  30761.71  804.0002  3590.4292  3308.9992  -5794.9994  537.8556
81    625.5  5885.25  6918.75  -2514.0  28822.51  645.0  2702.99  3417.0  -5170.5  1581.0
82    -195.125  -1619.625  -2029.125  558.875  -7692.875  42.125  112.38  -1188.125  1196.875  -1553.625
83    120.4282  2923.4958  3479.0664  -1549.141  15142.26  121.8569  1195.4272  1616.8546  -2961.8534  670.284
84    -108.8  3258.0  3642.8  -2324.0  18236.8  421.6  3079.2  1172.0  -4198.4  -1845.6
85    -152.8336  -512.00115  -737.00145  -53.17  -1912.8382  134.8333  742.1669  -625.83  31.1672  -1528.5
86    8.167  2281.63  2567.3773  -1468.4172  12368.4253  317.8335  1986.917  920.418  -2717.8346  -944.7486
87  ]

```

9. Tính định thức ma trận

- Tính định thức ma trận I chưa được nhập từ trước:

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhập ma trận với số thực      **
*      4. Cộng 2 ma trận                 **
*      5. Trừ 2 ma trận                  **
*      6. Nhân 2 ma trận                 **
*      7. Tính định thức                 **
*      8. Tìm ma trận nghịch đảo         **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình             **
*****
Chọn chức năng: 7
Nhập tên của ma trận: I
Error: Matrix does not exist

```

- Tính định thức ma trận A(4x5):

```

*****
**                                     **
**                                     MENU                                     **
**                                     **
** 1. Nhập ma trận từ bàn phím          **
** 2. Nhập ma trận từ file              **
** 3. Nhân ma trận với số thực          **
** 4. Cộng 2 ma trận                   **
** 5. Trừ 2 ma trận                    **
** 6. Nhân 2 ma trận                    **
** 7. Tính định thức                   **
** 8. Tìm ma trận nghịch đảo           **
** 9. Tìm ma trận chuyển vị            **
** 0. Thoát chương trình                **
*****

Chọn chức năng: 7
Nhập tên của ma trận: A
Error: The matrix is not square.

```

- Tính định thức ma trận B:

```

*****
**                                     **
**                                     MENU                                     **
**                                     **
** 1. Nhập ma trận từ bàn phím          **
** 2. Nhập ma trận từ file              **
** 3. Nhân ma trận với số thực          **
** 4. Cộng 2 ma trận                   **
** 5. Trừ 2 ma trận                    **
** 6. Nhân 2 ma trận                    **
** 7. Tính định thức                   **
** 8. Tìm ma trận nghịch đảo           **
** 9. Tìm ma trận chuyển vị            **
** 0. Thoát chương trình                **
*****

Chọn chức năng: 7
Nhập tên của ma trận: B

```

```

2 Định thức của ma trận B là:
3 -194733570432.0

```

- Tính định thức ma trận E:

```

Chọn chức năng: 7
Nhập tên của ma trận: E

4 Định thức của ma trận E là:
5 1.0725407560941522e+20

```

10. Tìm ma trận nghịch đảo

- Tìm ma trận nghịch đảo của G chưa được khai báo trước đó:

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình           **
*****
Chosen chức năng: 8
Nhập tên của ma trận: G
Error: Matrix does not exist

```

- Tìm ma trận nghịch đảo của A (4x5):

```

*****
**                                     MENU                                     **
*****
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình           **
*****
Chosen chức năng: 8
Nhập tên của ma trận: A
Error: The matrix is not square.

```

- Tìm ma trận nghịch đảo của ma trận E:

```

Chosen chức năng: 8
Nhập tên của ma trận: E

122 Ma trận nghịch đảo của E là :
123
124 [ 0.0078867151 -0.0002617064 -0.0001769635 -0.0003681719 -0.0011268525 -3.56806e-05 -0.0005871916 -0.0007688327 -0.0004910265 -0.0032212773
125 0.0005398347 0.0002569169 -0.0003508046 -0.0002748041 -0.0009395163 0.0006087638 0.0003794413 0.0001142982 -0.0007228472 -0.0028039903
126 0.0002038731 -0.0002290193 0.0112124714 -0.0002336761 -0.000768059 -0.001261247 0.0011577569 -0.0002014399 -0.0023475964 -0.0029982785
127 -0.0008186672 -0.0007793073 -0.0018457298 0.0074425184 -0.001306564 0.0001483917 -0.0002157972 -0.000396677 -0.0001927366 -0.0001654542
128 -0.0003931016 0.0011290511 -0.000483539 -0.0007939203 0.0086660634 0.0006313135 -0.0011947941 -0.0006051687 0.001270204 0.002216289
129 -8.03036e-05 -0.0008688531 -0.00034745 -0.0005848356 -0.000807756 0.0116366161 0.0006927611 -0.0004506871 0.0025770224 0.0026612618
130 0.0003785213 0.0004385114 -0.0004061068 -0.0009388225 0.0018030505 -0.0011930753 0.0097436547 0.0005965796 0.0012161267 0.0025814889
131 -0.0013086942 -0.0005463259 -0.0015365775 0.0001960705 5.63767e-05 -0.0010745058 -0.0015668161 0.0005656883 -0.000649197 -0.000686854
132 -0.0014088371 -1.86461e-05 0.0012118516 -0.0004400035 0.0012678093 -0.0010299425 -0.0005189069 -0.0011090567 0.0122094267 0.0011838451
133 0.0001120091 0.0004511893 0.0011396566 -0.0002802637 -0.0004655927 -0.0023010439 -0.000110364 -0.0017100287 0.0001601274 0.0183983304
134 ]

```


- Tìm ma trận nghịch đảo của B:

```
Chọn chức năng: 8
Nhập tên của ma trận: B
```

```
114 Ma trận nghịch đảo của B là :
115 [
116     0.006552136 -0.0005705004 -0.0008390343 -2.96401e-05 0.0002278174
117     0.0002408597 -0.0039564896 -0.0011304907 0.000552114 0.0001379399
118     0.000160679 0.0005106667 0.009286399 -0.0015354208 -0.0004388569
119     -0.0004976096 -2.68863e-05 -0.0001152152 0.0065160721 -0.0007032433
120     -0.0004487362 0.0002561841 0.0008898234 -0.0002820033 0.0033109753
121 ]
```

- Tìm ma trận nghịch đảo của E1 (định thức bằng 0):

```
*****
**                                     **
**                                     MENU                                     **
**                                     **
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình           **
*****
Chọn chức năng: 8
Nhập tên của ma trận: E1
The matrix is not invertible.
```

11. Tìm ma trận chuyển vị

- Tìm ma trận của ma trận I(Chưa được nhập từ trước):

```
*****
**                                     MENU                                     **
**                                     **
*      1. Nhập ma trận từ bàn phím      **
*      2. Nhập ma trận từ file          **
*      3. Nhân ma trận với số thực      **
*      4. Cộng 2 ma trận                **
*      5. Trừ 2 ma trận                 **
*      6. Nhân 2 ma trận                **
*      7. Tính định thức                **
*      8. Tìm ma trận nghịch đảo        **
*      9. Tìm ma trận chuyển vị         **
*      0. Thoát chương trình           **
*****
Chọn chức năng: 9
Nhập tên của ma trận: I
Error: Matrix does not exist
```

- Tìm ma trận chuyển vị của E:

Chọn chức năng: 9
Nhập tên của ma trận: E

```

135 Ma trận chuyển vị của E là :
136
137 [ 130.75 -7.0 2.8 16.5714 6.1667 -0.875 -7.5714 17.2 15.8333 0.9167 ]
138 [ 3.75 156.0 4.2 14.8571 -19.0 11.125 -1.4286 11.0 4.8333 -2.0 ]
139 [ 5.25 4.0 89.0 25.4286 9.8333 7.125 5.2857 15.6 -5.8333 -2.5833 ]
140 [ 11.25 8.0 5.4 140.4286 12.1667 6.375 9.8571 2.8 5.5 2.9167 ]
141 [ 18.25 17.6667 10.8 25.7143 114.0 13.375 -18.0 2.2 -9.6667 3.9167 ]
142 [ 6.5 -3.6667 12.6 2.1429 -6.1667 81.75 7.1429 14.2 7.6667 10.6667 ]
143 [ 12.0 -3.0 -7.8 5.5714 14.5 -6.0 96.8571 15.6 5.6667 2.1667 ]
144 [ 15.5 4.0 8.0 10.5714 3.6667 1.875 -10.1429 106.8 9.5 9.5833 ]
145 [ 2.75 9.0 14.8 5.4286 -10.5 -15.75 -8.8571 5.2 81.1667 -3.5833 ]
146 [ 20.0 21.6667 13.0 6.4286 -14.1667 -8.75 -12.8571 6.4 -3.0 52.0833 ]
147 ]

```

- Tìm ma trận chuyển vị của A:

Chọn chức năng: 9
Nhập tên của ma trận: A

```

156 Ma trận chuyển vị của A là :
157
158 [ 2.0 1.0 10.0 -5.0 ]
159 [ 3.0 0.0 5.0 3.0 ]
160 [ 5.0 2.0 2.0 -5.0 ]
161 [ 6.0 3.0 3.0 0.0 ]
162 [ 2.0 6.0 1.0 8.0 ]
163 ]

```

- Tìm ma trận chuyển vị của B:

Chọn chức năng: 9
Nhập tên của ma trận: B

```

148 Ma trận chuyển vị của B là :
149
150 [ 153.0 12.0 0.0 14.0 21.0 ]
151 [ -21.0 -258.0 15.0 -1.0 13.0 ]
152 [ 12.0 -31.0 108.0 0.0 -25.0 ]
153 [ 5.0 15.0 25.0 155.0 6.0 ]
154 [ -7.0 9.0 19.0 32.0 298.0 ]
155 ]

```

V Tổng kết

1. Những điều đã làm được và học được

- Hiểu hơn về các phép toán của ma trận.
- Biết cách áp dụng lí thuyết vào trình bày code, và giải thuật để có chương trình tốt.
- Học được các kỹ thuật phân tích yêu cầu, thiết kế và triển khai đối với một chương trình cụ thể.
- Thành thạo hơn trong việc sử dụng ngôn ngữ lập trình Python.
- Phát triển khả năng tự học, tự tìm hiểu đối với lập trình.

2. Khó khăn và hạn chế

Khó khăn:

- Do chưa có nhiều kinh nghiệm trong việc làm một project và viết báo cáo liên quan đến lập trình, nên em còn gặp một số khó khăn trong việc tìm hiểu và học hỏi, cũng như vận dụng lí thuyết vào thực hành.
- Việc mới bắt đầu học tập và sử dụng ngôn ngữ Python từ đầu kỳ học để phục vụ cho bài báo cáo này, nên kiến thức về ngôn ngữ còn hạn chế, nên trong quá trình làm vẫn phải tự tìm hiểu thêm.

Hạn chế:

- Sản phẩm code vẫn còn nhiều thiếu sót, chưa tối ưu được code và trình bày code trong sáng tối đa nhất có thể.
- Các chức năng trong chương trình vẫn chưa quá đa dạng.

3. Hướng phát triển chủ đề

- Phát triển thêm nhiều chức năng hơn về ma trận.
- Cải tiến giải thuật, tối ưu mã nguồn.

VI Tài liệu tham khảo

1. Slide bài giảng môn Kỹ thuật lập trình - TS. Vũ Thành Nam.
2. Giáo trình giải tích số - Lê Trọng Vinh.
3. Giáo trình đại số tuyến tính - TS. Bùi Xuân Diệu.
4. Slide giải tích số - TS. Hà Thị Ngọc Yến.
5. Ngôn ngữ Python - Viblo.asia, codelearn.io ...
6. Một số tài liệu tham khảo khác.