**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**

**Logo

Description automatically generated**

**ĐỒ ÁN LẬP TRÌNH TÍNH TOÁN**

**TÊN ĐỀ TÀI:   
Tìm hiểu mô hình dự báo hồi qui**

**và xây dựng ứng dụng**

Người hướng dẫn**: PGS. TS. Nguyễn Tấn Khôi**

Sinh viên thực hiện**:**

**Hoàng Minh Trí Mã SV: 102230220 NHÓM: YYY**

**Đặng Hoài Đức Mã SV: 102230287 NHÓM: YYY**

**Đà Nẵng, 03/2024**

MỤC LỤC

[MỤC LỤC i](#_Toc37650459)

[DANH MỤC HÌNH VẼ ii](#_Toc37650460)

[MỞ ĐẦU {font: TimeNew Roman, bold, size: 14, căn lề: center} i](#_Toc37650461)

[1. TỔNG QUAN ĐỀ TÀI 1](#_Toc37650462)

[2. CƠ SỞ LÝ THUYẾT 2](#_Toc37650463)

[2.1. Ý tưởng **Error! Bookmark not defined.**](#_Toc37650464)

[2.2. Cơ sở lý thuyết 2](#_Toc37650465)

[3. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN 2](#_Toc37650466)

[3.1. Phát biểu bài toán 2](#_Toc37650467)

[3.2. Cấu trúc dữ liệu 2](#_Toc37650468)

[3.3. Thuật toán 2](#_Toc37650469)

[4. CHƯƠNG TRÌNH VÀ KẾT QUẢ 2](#_Toc37650470)

[4.1. Tổ chức chương trình 2](#_Toc37650471)

[4.2. Ngôn ngữ cài đặt **Error! Bookmark not defined.**](#_Toc37650472)

[4.3. Kết quả 3](#_Toc37650473)

[4.3.1. Giao diện chính của chương trình 3](#_Toc37650474)

[4.3.2. Kết quả thực thi của chương trình 3](#_Toc37650475)

[4.3.3. Nhận xét đánh giá 3](#_Toc37650476)

[5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 3](#_Toc37650477)

[5.1. Kết luận 3](#_Toc37650478)

[5.2. Hướng phát triển 3](#_Toc37650479)

[TÀI LIỆU THAM KHẢO 4](#_Toc37650480)

DANH MỤC HÌNH VẼ

No table of figures entries found.

DANH MỤC TỪ VIẾT TẮT

DANH MỤC KÝ HIỆU TOÁN HỌC

MỞ ĐẦU

Trong những năm gần đây, lĩnh vực Công nghệ Thông tin đã phát triển mạnh mẽ và trở thành một trong những ngành vững mạnh. Sự tiến bộ của xã hội đã làm cho Công nghệ Thông tin trở thành một phần không thể thiếu trong hầu hết các lĩnh vực, đóng vai trò ngày càng quan trọng hơn. Với chúng tôi, những sinh viên trong lĩnh vực này, việc đầu tư và không ngừng học hỏi là điều cần thiết để nâng cao kiến thức và kỹ năng. Vì vậy, để củng cố kiến thức, chúng tôi đã chọn đề tài "Mô hình dự báo hồi qui và xây dựng ứng dụng dự báo theo chuỗi dữ liệu thời gian " cho dự án của mình.

Mô hình Hồi quy Tuyến tính không còn xa lạ với nhiều người trong lĩnh vực nghiên cứu. Áp dụng nó vào Công nghệ Thông tin mang lại nhiều lợi ích, với việc đầu tư ít thời gian nhưng mang lại hiệu quả cao, chính xác và tiện lợi trong thời gian ngắn.

**1. Mục đích và mục tiêu thực hiện đề tài :**

**1.1 Mục đích :**

**-** Áp dụng được kiến thức các môn học trong việc xây dựng mô hình hồi qui tuyến tính và ứng dụng dự báo theo chuỗi dữ liệu thời gian.

**-** Dần hoàn thiện kỹ năng và tư duy lập trình tính toán.

**1.2 Mục tiêu :**

**-** Xây dựng ứng dụng dự đoán chính xác hoặc mô tả các biến phụ thuộc dựa trên các biến độc lập.

**2. Phạm vi và đối tượng nghiên cứu :**

**2.1 Phạm vi nghiên cứu :**

**-** Phân tích và xử lý dữ liệu thời gian: Bao gồm việc thu thập dữ liệu thời gian từ các nguồn khác nhau, kiểm tra tính đúng đắn và đầy đủ của dữ liệu, và xử lý các vấn đề như thiếu sót, nhiễu.

- Xây dựng mô hình hồi qui và chuỗi dữ liệu để dự báo giá trị của các biến số trong tương lai.

- Đánh giá và tinh chỉnh mô hình để cải thiện hiệu suất và độ chính xác.

**2.2 Đối tượng nghiên cứu :**

**-** Các chuỗi dữ liệu thời gian. Ví dụ như nhu cầu thị trường, giá cả hàng hoá, tỷ lệ thất nghiệp, … cùng nhiều biến số khác.

- Mô hình hồi qui và chuỗi dữ liệu thời gian được áp dụng để dự đoán giá trị các biến số trong tương lai.

- Các lĩnh vực và ứng dụng: Mô hình dự án có thể áp dụng vào nhiều lĩnh vực như y tế, kinh tế - tài chính, kinh doanh, dự báo thời tiết và nhiều lĩnh vực khác.

**3. Phương pháp nghiên cứu :**

**3.1 Phương pháp lý thuyết :**

**-** Nghiên cứu về phương pháp và cơ sở lý thuyết của mô hình dự báo hồi qui và chuỗi dữ liệu thời gian.

**3.2 Phương pháp thực hiện :**

**-** Lựa chọn ngôn ngữ lập trình C++ để xây dựng chương trình.

- Thực hiện chương trình trên các bộ dữ liệu thực tế.

- Đánh giá, kiểm tra kết quả.

**4. Cấu trúc đồ án môn học :**

Bố cục đồ án gồm 3 chương như sau:

**Chương 1: Cơ sở lý thuyết.**

**Chương 2: Xây dựng chương trình và kết quả thực hiện**

**Chương 3: Kết luận và hướng phát triển**

TỔNG QUAN ĐỀ TÀI

Trong môi trường kinh doanh ngày nay, sự cạnh tranh ngày càng trở nên khốc liệt hơn bao giờ hết. Doanh nghiệp phải đối mặt với áp lực từ cả những đối thủ truyền thống lâu năm và từ sự xuất hiện của các công ty khởi nghiệp đầy sáng tạo. Để tồn tại và phát triển trong cảnh cạnh tranh này, việc nắm bắt và dự đoán được xu hướng và biến động trong thị trường là không thể phủ nhận. Trong bối cảnh toàn cầu hóa và sự phát triển mạnh mẽ của công nghệ thông tin, thông tin và dữ liệu trở thành một phần quan trọng và không thể thiếu trong việc đưa ra quyết định kinh doanh chiến lược.

Với sự tiến bộ của khoa học và công nghệ, đặc biệt là trong lĩnh vực của công nghệ thông tin, khả năng thu thập, xử lý và phân tích dữ liệu đã trở nên dễ dàng và hiệu quả hơn. Công nghệ big data, machine learning và data analytics đã mở ra những cánh cửa mới cho việc dự đoán và đánh giá thông tin từ dữ liệu lớn, đồng thời cung cấp cái nhìn sâu sắc và đầy đủ hơn về thị trường và khách hàng.

Xuất phát từ nhu cầu thực tiễn của xã hội, đặc biệt là của các doanh nghiệp, mô hình hồi qui tuyến tính và ứng dụng dự báo theo chuỗi dữ liệu thời gian đã trở thành công cụ không thể thiếu trong việc dự đoán và quản lý rủi ro trong kinh doanh. Những mô hình này không chỉ giúp dự đoán xu hướng và biến động của thị trường mà còn hỗ trợ trong việc lập kế hoạch sản xuất, quản lý chuỗi cung ứng, và đưa ra các quyết định chiến lược.

Mục tiêu của các bài toán dự báo không chỉ là tạo ra các mô hình dự đoán chính xác, mà còn là tạo ra các công cụ dễ sử dụng và linh hoạt đối với tất cả mọi người trong tổ chức. Việc này đảm bảo rằng những dự báo và thông tin dựa trên dữ liệu được truy cập và áp dụng một cách hiệu quả nhất trong quá trình ra quyết định kinh doanh hàng ngày.

# CƠ SỞ LÝ THUYẾT

## Giới thiệu

- Hồi quy tuyến tính là một phương pháp thống kê được sử dụng để nghiên cứu mối quan hệ giữa các biến liên tục. Nó giả định rằng có một mối quan hệ tuyến tính giữa các biến độc lập (thường được ký hiệu là {"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"><msub><mi>X</mi><mi>i</mi></msub></mstyle></math>","origin":"MathType for Microsoft Add-in"} ) và biến phụ thuộc (thường được ký hiệu là Y).

A diagram of a triangle with points and lines

Description automatically generated

* Ưu điểm:

– Mô hình đơn giản, dễ hiểu.

– Dễ diễn giải hệ số hồi quy.

– Nhận được kết quả tốt khi dữ liệu quan sát nhỏ.

– Nhiều cải tiến/mở rộng.

* Nhược điểm:

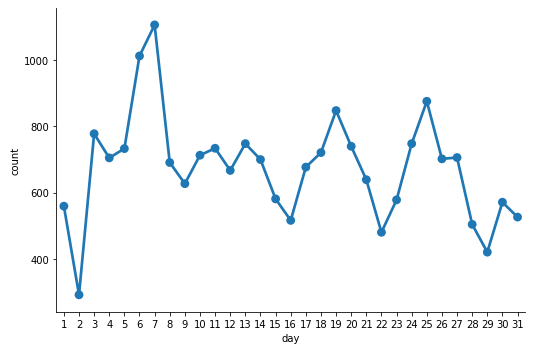
– Mô hình hơi đơn giản nên khó dự đoán chính xác với dữ liệu có miền giá trị rộng.

– Khả năng ngoại suy (extrapolation) kém.

– Nhạy cảm với dữ liệu ngoại lai (outliers).

– Do sử dụng phương pháp bình phương nhỏ nhất.

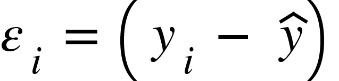
- Chuỗi thời gian là chuỗi các quan sát được thu thập trên cùng một đối tượng tại các mốc thời gian cách đều nhau. Số liệu chuỗi thời gian cung cấp thông tin về cùng một đối tượng tại các thời điểm khác nhau. Số liệu chuỗi thời gian được phân biệt theo tần suất xuất hiện: số liệu theo năm, số liệu theo quý, số liệu theo tháng, ...



- Chuỗi dữ liệu thời gian có thể biểu diễn thời gian theo các định dạng khác nhau như ngày/tháng/năm, giờ:phút:giây, hoặc dưới dạng số nguyên biểu diễn số giây từ một thời điểm thời gian cố định. Chuỗi dữ liệu thời gian thường được sử dụng để ghi log, xử lý dữ liệu thời gian trong các ứng dụng, và hiển thị thời gian cho người dùng.

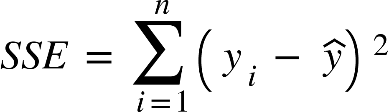
## Phần dư (Lỗi):

- Phần dư, còn được gọi là lỗi được tính:



Phần dư (Lỗi)

- Đường thẳng tìm thấy được đánh giá là phù hợp nhất khi tổng bình phương lỗi là nhỏ nhất, với tổng bình phương lỗi được tính như sau:

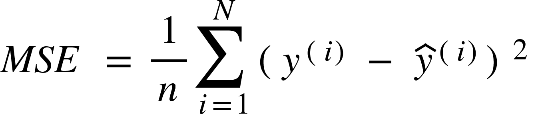


Tổng bình phương lỗi (SSE)

## Bình phương nhỏ nhất:

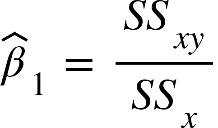
- Bình phương nhỏ nhất (Least Square): Để đánh giá độ phù hợp của mô hình từ dữ liệu quan sát ta sử dụng phương pháp bình phương nhỏ nhất.

- Lỗi bình phương trung bình (Mean squared error): Để đánh giá độ phù hợp của mô hình. Giá trị của lỗi càng nhỏ, độ phù hợp càng tang.

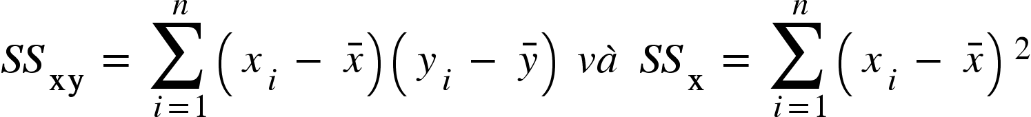


Lỗi bình phương trung bình (MSE)

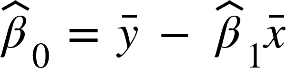
## Độ dốc của đường thẳng:



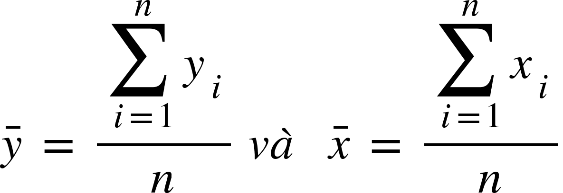
Trong đó:



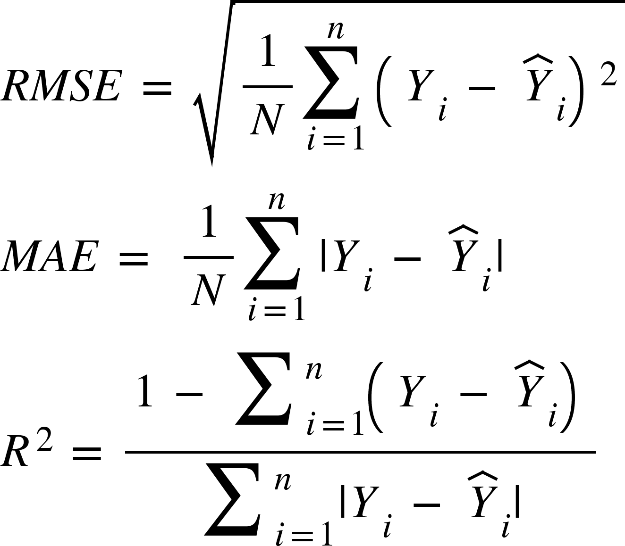
## Hệ số chặn của đường thẳng:



Trong đó:



## Phương pháp đánh giá:



Trong đó:

* RMSE (Root Mean Square Erro): Chỉ số đánh giá hiệu suất của mô hình dự báo, phản ánh mức độ chênh lệch giữa giá trị dự báo và thực tế.
* MAE: Sai số tuyệt đối trung bình.
* {"mathml":"<math xmlns=\"http://www.w3.org/1998/Math/MathML\" style=\"font-family:stix;font-size:16px;\"><msup><mi>R</mi><mn>2</mn></msup></math>","origin":"MathType for Microsoft Add-in"}: Phép đo để đánh giá độ phù hợp.

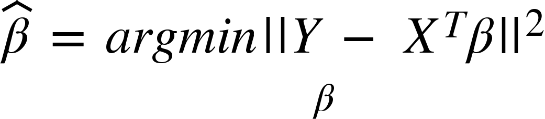
## Quá trình tính toán giải quyết bài toán

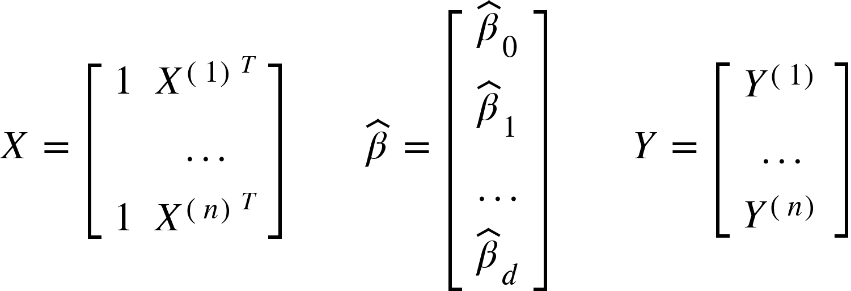
1. Giới thiệu

- Để giải quyết bài toán hồi quy tuyến tính đơn biến, sử dụng các cơ sở lý thuyết trên ta có thể dễ dàng giải quyết bài toán đơn biến. Kết hợp sử dụng ma trận và các tính chất của ma trận, ta có thể xử lý các bài toán hồi quy tuyến tính đa biến phức tạp.

b) Các bước thực hiện tính toán.

* Bước 1: Tìm các ước số bằng phương pháp bình phương nhỏ nhất.

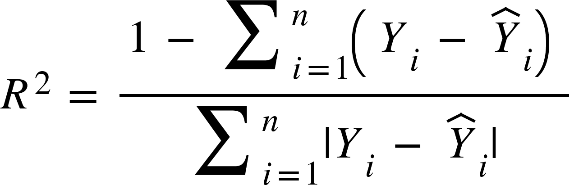




* Bước 2: Giải phương trình để tìm hệ số chặn và độ dốc.

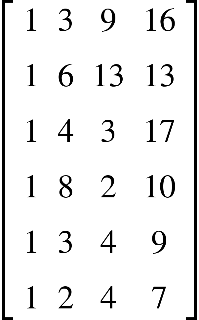
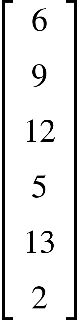
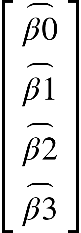
{"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"><mover><mi>&#x3B2;</mi><mo>^</mo></mover><mo>&#xA0;</mo><mo>=</mo><mo>&#xA0;</mo><msup><mfenced><mrow><msup><mi>X</mi><mi>T</mi></msup><mi>X</mi></mrow></mfenced><mrow><mo>-</mo><mn>1</mn></mrow></msup><mo>&#xA0;</mo><msup><mi>X</mi><mi>T</mi></msup><mi>Y</mi></mstyle></math>","origin":"MathType for Microsoft Add-in"}{"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"/></math>","origin":"MathType for Microsoft Add-in"}

* Bước 3: Đánh giá.



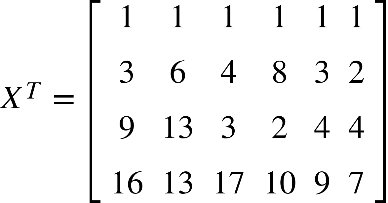
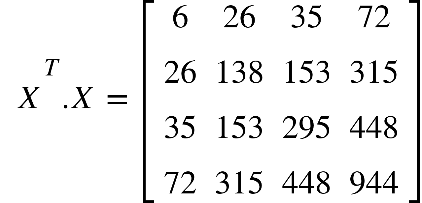
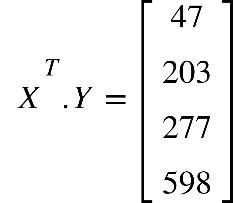
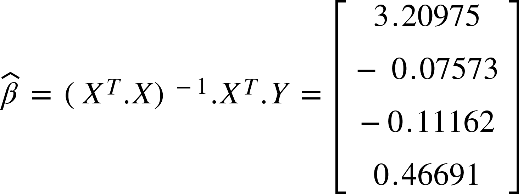
1. Ví dụ:

- Cho các tập sau:

X =  Y =  {"mathml":"<math xmlns=\"http://www.w3.org/1998/Math/MathML\" style=\"font-family:stix;font-size:16px;\"><mover><mi>&#x3B2;</mi><mo>&#x23DC;</mo></mover></math>","origin":"MathType for Microsoft Add-in"} = 

- Tìm phương trình phù hợp:

Ta có:

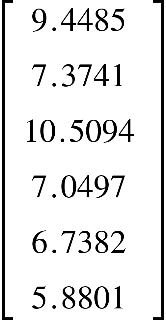
* 
* 
* 
* 
* {"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"><mi>Y</mi><mo>&#xA0;</mo><mo>=</mo><mo>&#xA0;</mo><mn>3</mn><mo>.</mo><mn>20975</mn><mo>-</mo><mn>0</mn><mo>.</mo><mn>07573</mn><mo>&#xD7;</mo><msub><mi>X</mi><mn>1</mn></msub><mo>-</mo><mn>0</mn><mo>.</mo><mn>11162</mn><mo>&#xD7;</mo><msub><mi>X</mi><mn>2</mn></msub><mo>+</mo><mn>0</mn><mo>.</mo><mn>46691</mn><mo>&#xD7;</mo><msub><mi>X</mi><mn>3</mn></msub></mstyle></math>","origin":"MathType for Microsoft Add-in"}

- Đánh giá:

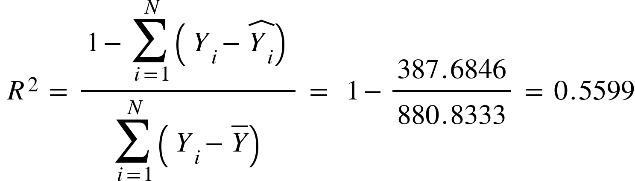
+) Bước 1: Tính các giá trị {"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"><mover><msub><mi>Y</mi><mi>i</mi></msub><mo>&#xAF;</mo></mover></mstyle></math>","origin":"MathType for Microsoft Add-in"}

{"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"><mi>Y</mi><mo>&#xA0;</mo><mo>=</mo><mo>&#xA0;</mo><mn>3</mn><mo>.</mo><mn>20975</mn><mo>-</mo><mn>0</mn><mo>.</mo><mn>07573</mn><mo>&#xD7;</mo><msub><mi>X</mi><mn>1</mn></msub><mo>-</mo><mn>0</mn><mo>.</mo><mn>11162</mn><mo>&#xD7;</mo><msub><mi>X</mi><mn>2</mn></msub><mo>+</mo><mn>0</mn><mo>.</mo><mn>46691</mn><mo>&#xD7;</mo><msub><mi>X</mi><mn>3</mn></msub></mstyle></math>","origin":"MathType for Microsoft Add-in"} Với:

Ta tính được giá trị các {"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"><mover><msub><mi>Y</mi><mi>i</mi></msub><mo>&#x23DC;</mo></mover></mstyle></math>","origin":"MathType for Microsoft Add-in"} lần lượt là:



+) Bước 2: Tính {"mathml":"<math style=\"font-family:stix;font-size:16px;\" xmlns=\"http://www.w3.org/1998/Math/MathML\"><mstyle mathsize=\"16px\"><msup><mi>R</mi><mn>2</mn></msup></mstyle></math>","origin":"MathType for Microsoft Add-in"}



# XÂY DỰNG CHƯƠNG TRÌNH VÀ KẾT QUẢ THỰC HIỆN

## Phát biểu bài toán

- Xây dựng công cụ tính toán, dự đoán số liệu theo mô hình dự báo hồi qui.

- Khai báo dữ liệu: Các biến đầu vào.

- Input: Nhận dữ liệu đầu vào từ file csv có sẵn (đã được xử lý)

- Output: Các số liệu cần tính như độ dốc, hệ số chặn, độ phù hợp của và hình vẽ biểu diễn (2 biến). A diagram of a person with text

Description automatically generated with medium confidence

## Cấu trúc dữ liệu xây dựng để giải quyết

## Thuật toán

- Chương trình bao gồm 6 thuật toán chính:

* Trans\_matrix (Chuyển vị ma trận)
* Determinant\_matrix (Định thức của ma trận)
* Inverse\_matrix (Nghịch đảo ma trận)
* Get\_submatrix (Xác định ma trận con)
* Multiple\_matrix (Nhân hai ma trận)
* CaclR2 (Tính độ phù hợp của phương trình)

### Thuật toán 01 (Trans\_matrix):

1. Giới thiệu

- Trans\_matrix là thuật toán chuyển vị ma trận, được sử dụng chuyển vị một ma trận cỡ m \* n thành ma trận cỡ n \* m.

b) Các bước thực hiện

1. Nhận đầu vào là ma trận “Matrix” dưới dạng là vector của vector double.
2. Xác định số hàng và số cột của ma trận bằng cách sử dụng ‘.size()’ của vector.
3. Khởi tạo một ma trận chuyển vị ‘transposed’ với số hàng bằng số cột và số cột bằng số hàng của ma trận ban đầu.
4. Duyệt qua từng phần tử của ma trận ban đầu, sao chép giá trị của phần tử tại hàng i và cột j của ma trận ban đầu vào hàng j và cột i của ma trận chuyển vị.
5. Trả về ma trận chuyển vị ‘transposed’.

c) Đánh giá độ phức tạp

- Độ phức tạp của thuật toán Trans\_matrix được đánh giá như sau:

1. - Xác định số hàng và số cột của ma trận: Độ phức tạp: O(1)

- Vì chúng ta chỉ cần gọi các hàm size() của vector để lấy số hàng và số cột của ma trận.

1. - Khởi tạo ma trận chuyển vị: Độ phức tạp: O(n \* m)

- Với n, m là số cột và hàng của ma trận ban đầu. Việc khởi tạo ma trận chuyển vị đòi hỏi hai vòng lặp lòng ghép nhau, mỗi lần khởi tạo vector cột ta cần m lần khởi tạo vector hàng.

1. Duyệt qua từng phần tử của ma trận ban đầu và gán vào ma trận chuyển vị: Độ phức tạp: O(n \* m)

- Vậy tổng độ phức tạp của thuật toán Trans\_matrix là O(n \* m)

d) Mã nguồn

### Thuật toán 02 (Determinant\_matrix)

1. Giới thiệu

- Determinant\_matrix là thuật toán dùng để tính định thức của ma trận.

b) Các bước thực hiện

c) Đánh giá độ phức tạp

d) Mã nguồn

### Thuật toán 03 (Inverse\_matrix)

1. Giới thiệu

- Inverse\_matrix là thuật toán sử dụng để tìm ma trận nghịch đảo của ma trận ban đầu đã cho.

1. Các bước thực hiện
2. Tính định thức của ma trận đầu vào bằng cách sử dụng hàm determinant.
3. Khởi tạo một ma trận nghịch đảo với cùng kích thước như ma trận đầu vào.
4. Duyệt qua từng phần tử của ma trận nghịch đảo và tính giá trị của nó:

* Đối với mỗi phần tử [i][j], ta lấy ra ma trận con bằng cách loại bỏ hàng i và cột j từ ma trận đầu vào, sử dụng hàm getSubMatrix.
* Tính định thức của ma trận con đã lấy được, nhân với (-1)(i+j) , và chia cho định thức của ma trận đầu vào để thu được giá trị của phần tử [i][j] trong ma trận nghịch đảo.

1. Chuyển vị của ma trận vừa tìm được bằng cách sử dụng hàm transposeMatrix.
2. Trả về ma trận nghịch đảo.
3. Đánh giá độ phức tạp

- Độ phức tạp của thuật toán Inverse\_matrix được đánh giá như sau:

1. Mã nguồn

### Thuật toán 04 (Get\_submatrix)

1. Giới thiệu
2. Các bước thực hiện
3. Đánh giá độ phức tạp
4. Mã nguồn

### Thuật toán 05 (Multiple\_matrix)

1. Giới thiệu
2. Các bước thực hiện
3. Đánh giá độ phức tạp
4. Mã nguồn

### Thuật toán 06 (CaclR2)

1. Giới thiệu
2. Các bước thực hiện
3. Đánh giá độ phức tạp
4. Mã nguồn

## Tổ chức chương trình

## Cài đặt chương trình

a) Môi trường cài đặt: Hệ điều hành, cấu hình máy, tốc độ mạng, ...

b) Các bước triển khai cài đặt

## Mô tả bộ dữ liệu

## Kịch bản thực hiện 1

### Giao diện chính của chương trình

### Các kết quả thực thi của chương trình

Mô tả kết quả 01 thực hiện chương trình, hình vẽ, giải thích.

Mô tả kết quả 02 thực hiện chương trình, hình vẽ, giải thích.

….

### Nhận xét đánh giá kết quả

## Kịch bản thực hiện 2

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

**1. Kết quả đạt được**

**2. Hướng phát triển**

TÀI LIỆU THAM KHẢO

[Giới thiệu về dữ liệu chuỗi thời gian (Time-series data) | Bài viết - Science for Economics](https://bai-viet.scienceforeconomics.com/posts/gioi-thieu-ve-du-lieu-chuoi-thoi-gian-time-series-data)

PHỤ LỤC

Sinh viên liệt kê mã nguồn chính từng phần vào đây.{Font: Courier; thường; cỡ chữ: 10}