

Đồ án cấu trúc dữ liệu và giải thuật

Phạm Đình Sỹ, 1612572

1. Mức độ hoàn thành

STT	Chức năng	Chức năng cụ thể	Mức độ hoàn thành
1	Làm phép toán hàm Bool	Tìm tất cả các công thức đa thức tối thiểu của hàm Bool	100%
2	Làm phép toán trên vector	+ Phép toán cộng hai vector + Phép toán nhân vector với 1 số alpha	100%
3	Làm phép toán trên ma trận	+ Tìm định thức của ma trận + Nghịch đảo ma trận + Tích hai ma trận + Tìm hạng của ma trận + Hệ phương trình tuyến tính	70% - Chức năng hoàn thành: Tìm định thức, nghịch đảo ma trận, tính tích ma trận - Chức năng chưa hoàn thiện: tìm hạng ma trận - Chức năng chưa làm : hệ phương trình tuyến tính

2. Công việc thực hiện

1. Làm phép toán trên hàm Bool.

Tìm công thức đa thức tối thiểu của hàm Bool, hàm chính:

void findMinimalPolynomial(string str);

Danh sách chi tiết các hàm hỗ trợ cho chương trình:

int lenOfMonomial(string str);

- Xác định độ dài đơn thức
- Tham số truyền vào: chuỗi str đại diện đơn thức.
- Trả về độ dài đơn thức (số biến x/y/z/t của đơn thức)

void findMonomial(string str, string arrMonomial[MAX], int& len);

- Tìm danh sách đơn thức trong hàm Bool
- Tham số truyền vào: chuỗi str đại diện cho hàm Bool, mảng chuỗi arrMonomial lưu danh sách đơn thức, biến len kiểu int lưu độ dài của mảng
- Hàm thay đổi giá trị mảng arrMonomial và biến len tương ứng là danh sách đơn thức của hàm Bool và độ dài mảng

void tickWithMonomial(int Kar[4][4], string tmp);

- Đánh dấu các ô trong biểu đồ Karnaugh với một đơn thức nhập vào. Dùng để hỗ trợ cho hàm tickWithBool()
- Tham số truyền vào: Mảng int 2 chiều Kar thể hiện cho biểu đồ Karnaugh, chuỗi tmp đại diện đơn thức
- Hàm thay đổi giá trị mảng 2 chiều Kar thành biểu đồ Karnaugh của đơn thức tmp

void tickWithBool(int Kar[4][4], string str);

- Đánh dấu các ô trong biểu đồ Karnaugh với hàm Bool nhập vào
- Tham số truyền vào: Mảng int 2 chiều Kar của biểu đồ Karnaugh, chuỗi str của hàm Bool
- Hàm thay đổi giá trị mảng 2 chiều Kar thành biểu đồ Karnaugh của hàm Bool str

int isChild(string child, string parent);

- Xác định một đơn thức/ đa thức có thuộc đơn thức/ đa thức khác hay không
- Tham số truyền vào: chuỗi child và chuỗi parent
- Hàm trả về 1 nếu child thuộc parent, ngược lại trả về 0

void showKar(int Kar[4][4]);

- In ra biểu đồ Karnaugh
- Tham số truyền vào: mảng 2 chiều Kar
- Hàm in ra biểu đồ Karnaugh tương ứng

void printAllCell();

- Tìm tất cả các tế bào của biểu đồ Karnaugh
- Hàm không có tham số truyền vào
- Hàm in ra tất cả các tế bào có thể có của biểu đồ Karnaugh

void findMinimalPolynomial(string str);

- Tìm các công thức đa thức tối thiểu của hàm Bool
- Tham số truyền vào: chuỗi str đại diện hàm Bool
- Hàm in ra tất cả các đa thức tối thiểu của hàm Bool

2. Phép toán trên vector

Hàm cộng 2 vector:

Vector plusVector(Vector a, Vector b);

Hàm nhân vector với số thực

Vector mulVector(Vector a, float alpha);

Định nghĩa cấu trúc dữ liệu sử dụng: vector bậc N (degree)

struct Vector

{

int degree;

float value[MAX_DEGREE];

};

Danh sách chi tiết các hàm sử dụng trong chương trình

void setVector(Vector &A);

- Hàm tạo Vector 0
- Tham số truyền vào: Vector A
- Hàm set các giá trị 0 cho Vector A

int stringToInt(string str);

- Hàm chuyển đổi số nguyên kiểu string về kiểu int
- Tham số truyền vào: số nguyên dương kiểu string
- Trả về kết quả là số nguyên dương kiểu int

float stFloat(string str);

- Hàm chuyển đổi số thực dương kiểu string về kiểu float
- Tham số truyền vào: số thực dương kiểu string
- Trả về kết quả là số thực dương kiểu float

float stringToFloat(string str);

- Hàm chuyển đổi số thực kiểu string về kiểu float
- Tham số truyền vào: số thực kiểu string
- Trả về kết quả là số thực kiểu float

void findValue(string str, string arrValue[MAX_VALUE], int& len);

- Hàm lấy ra các giá trị của vector từ biểu diễn vector dạng chuỗi
- Tham số truyền vào: chuỗi str là vector dạng chuỗi, mảng chuỗi arrValue lưu các giá trị của vector, len là số phần tử mảng
- Hàm lưu các giá trị của vector str vào mảng arrValue cùng số phần tử vào biến len

void printVector(Vector a);

- Hàm in ra vector
- Tham số truyền vào: vector a
- Hàm in ra màn hình giá trị vector a

Vector plusVector(Vector a, Vector b);

- Hàm cộng 2 vector
- Tham số truyền vào: vector a, vector b
- Trả về kết quả là vector tổng của 2 vector a và b

Vector mulVector(Vector a, float alpha);

- Hàm nhân vector với một số alpha
- Tham số truyền vào: vector a, số thực alpha
- Trả về kết quả là vector tích của vector a với số thực alpha

3. Phép toán trên ma trận

Hàm tìm định thức ma trận

float detMatrix(Matrix pMatrix);

Hàm tìm ma trận nghịch đảo

Matrix inverseMatrix(Matrix pMatrix);

Hàm tìm tích 2 ma trận

Matrix mulMatrix(Matrix x, Matrix y);

Hàm tìm hạng ma trận vuông

int findRank(Matrix pMatrix);

Hàm tìm hạng ma trận (chưa hoàn thành)

int findRankMatrix(Matrix pMatrix);

Định nghĩa cấu trúc dữ liệu sử dụng trong chương trình: ma trận NxM

struct Matrix

```
{  
    int N;  
    int M;  
    float a[MAX_MATRIX][MAX_MATRIX]; // a[dòng][cột]  
};
```

Danh sách chi tiết các hàm sử dụng trong chương trình:

void printMatrix(Matrix pMatrix);

- Hàm in ra ma trận
- Tham số truyền vào: ma trận pMatrix
- Hàm in ra ma trận pMatrix

void setMatrix(Matrix & pMatrix);

- Hàm tạo Ma trận 0
- Tham số truyền vào: ma trận pMatrix
- Hàm set các giá trị 0 cho ma trận pMatrix

void refreshMatrix(Matrix& pMatrix);

- Hàm trình bày lại ma trận trong trường hợp hiển thị -0 thay vì 0
- Tham số truyền vào: ma trận pMatrix
Hàm sửa các giá trị -0 thay bằng 0

float detMatrix(Matrix pMatrix);

- Hàm tìm định thức ma trận
- Tham số truyền vào: ma trận pMatrix
- Trả về kết quả số thực là định thức của ma trận pMatrix

Matrix inverseMatrix(Matrix pMatrix);

- Hàm tìm ma trận nghịch đảo
- Tham số truyền vào: ma trận pMatrix
- Trả về kết quả ma trận là ma trận nghịch đảo của pMatrix

Matrix mulMatrix(Matrix x, Matrix y);

- Hàm tính tích 2 ma trận
- Tham số truyền vào: ma trận x và ma trận y
- Trả về kết quả ma trận là tích của 2 ma trận truyền vào

void createChildMatrix(Matrix t, Matrix listChild[100], int& len);

- Hàm tạo các ma trận con
- Tham số truyền vào: ma trận t, mảng danh sách ma trận con, độ dài mảng
- Hàm tăng thêm số lượng ma trận con trong mảng, đồng thời cập nhập độ dài mảng

int findRank(Matrix pMatrix);

- Hàm tìm hạng của ma trận vuông
- Tham số truyền vào: ma trận pMatrix
- Trả về kết quả nguyên là hạng của ma trận pMatrix

int findRankMatrix(Matrix pMatrix);

- Hàm tìm hạng của ma trận
- Tham số truyền vào: ma trận pMatrix
- Trả về kết quả nguyên là hạng của ma trận pMatrix

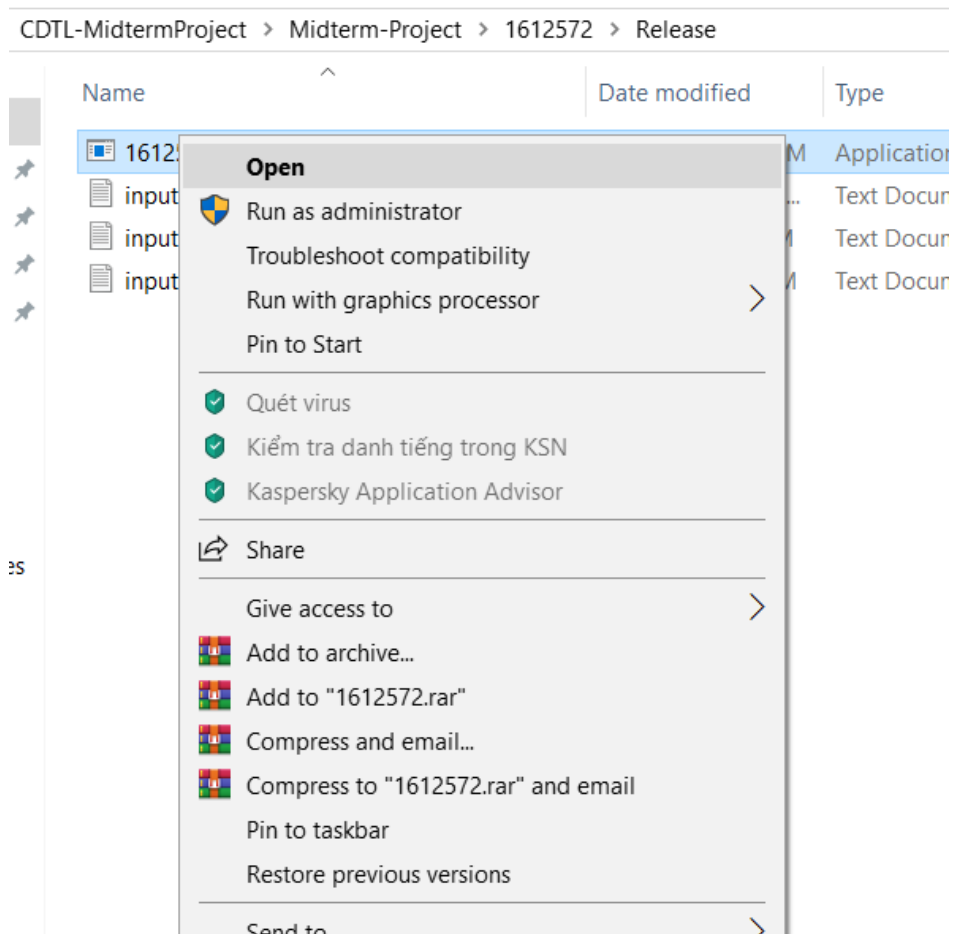
3. Sơ đồ lớp

4. Lưu đồ thuật toán

5. Hướng dẫn sử dụng

Chương trình nhập dữ liệu từ file input và xuất kết quả tính được ra màn hình

Để chạy chương trình, click chuột vào file .exe trong thư mục Release



Chương trình in ra kết quả theo dữ liệu từ các file input

C:\Users\Sy Pham\Desktop\CDTL-MidtermProject\Midterm-Proj

```
_____P1: HAM BOOL_____

-----
HAM BOOL: -yzt + -y-z-t + y-z-t + xyzt + -xz-t
BIEU DO KARNAUGH:
. . Q Q
Q Q . Q
. . . .
Q Q Q Q
CONG THUC DA THUC TOI TIEU:
-CT 1: -z-t + -x-t + xzt + -x-yz
-CT 2: -z-t + -x-t + xzt + -yzt
-----

-----
HAM BOOL: xyzt + x-y + x-z + yz + xy-z + xy-t
BIEU DO KARNAUGH:
Q Q Q .
Q Q Q .
Q Q . .
Q Q . .
CONG THUC DA THUC TOI TIEU:
-CT 1: x + yz
-----

_____P2: VECTOR_____
```

1. Tổ chức file input chức năng 1: hàm Bool

Tên file: inputBool.txt

```
inputBool - Notepad
File Edit Format View Help
|-yzt + -y-z-t + y-z-t + xyzt + -xz-t
xyzt + x-y + x-z + yz + xy-z + xy-t
Windows ( Ln 1, Col 1 100%
```

File lưu 2 hàm Bool, mỗi dòng 1 hàm Bool với quy ước

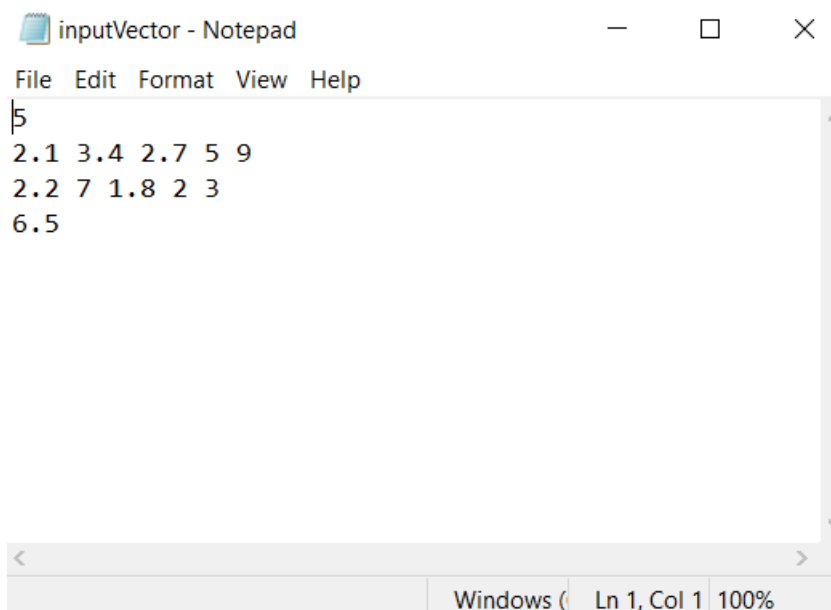
- Phép hợp kí hiệu bằng dấu cộng (+)
- Phủ định của biến x là -x
- Các từ đơn cách nhau bởi các dấu cách và dấu cộng

Chương trình sẽ nạp dữ liệu từ file và thực hiện in ra màn hình hàm Bool, biểu đồ Karnaugh cùng các công thức đa thức tối thiểu của nó.

```
P1: HAM BOOL
-----
HAM BOOL: -yzt + -y-z-t + y-z-t + xyt + -xz-t
BIEU DO KARNAUGH:
. . Q Q
Q Q . Q
. . .
Q Q Q Q
CONG THUC DA THUC TOI TIEU:
-CT 1: -z-t + -x-t + xzt + -x-yz
-CT 2: -z-t + -x-t + xzt + -yzt
-----
```

2. Tổ chức file input chức năng 2: tính toán trên vector

Tên file: inputVector.txt



```
inputVector - Notepad
File Edit Format View Help
5
2.1 3.4 2.7 5 9
2.2 7 1.8 2 3
6.5
Windows Ln 1, Col 1 100%
```

Dòng thứ nhất lưu bậc của vector

Dòng thứ hai lưu vector a: các giá trị vector ngăn cách bởi dấu cách

Dòng thứ ba lưu vector b: các giá trị vector ngăn cách bởi dấu cách

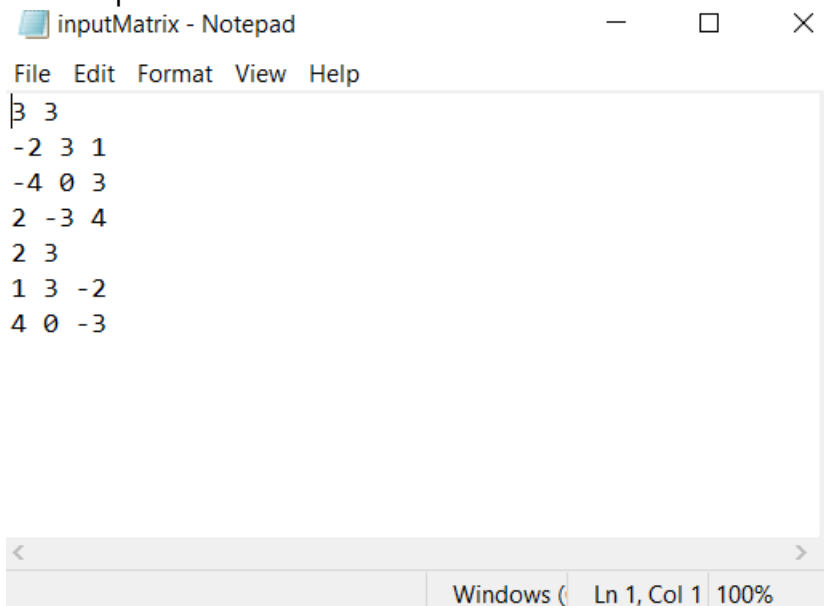
Dòng thứ tư lưu số thực alpha để thực hiện phép nhân vector

Chương trình sẽ nạp dữ liệu từ file và thực hiện in ra màn hình số thực alpha, vector a, vector b, tổng vector a+b, tích vector a với số thực alpha

```
_____P2: VECTOR_____
-----
ALPHA:          6.5
VECTOR A:       (2.1, 3.4, 2.7, 5, 9)
VECTOR B:       (2.2, 7, 1.8, 2, 3)
TONG 2 VECTOR:  (4.3, 10.4, 4.5, 7, 12)
TICH A*ALPHA:   (13.65, 22.1, 17.55, 32.5, 58.5)
-----
```

3. Tổ chức file input chức năng 3: tính toán trên ma trận

Tên file: inputMatrix.txt



```
File Edit Format View Help
3 3
-2 3 1
-4 0 3
2 -3 4
2 3
1 3 -2
4 0 -3

Windows | Ln 1, Col 1 | 100%
```

Dòng thứ nhất lưu kích thước ma trận p ($N_p \times M_p$)

N_p dòng tiếp theo lưu các hàng, mỗi hàng lưu các giá trị tương ứng của ma trận p

Dòng tiếp theo lưu kích thước ma trận q ($N_q \times M_q$)

N_q dòng tiếp theo lưu các hàng, mỗi hàng lưu các giá trị tương ứng của ma trận q

- Ma trận p nhập vào phải là ma trận vuông
- Ma trận q nhập vào phải có kích thước $M_b = N_a$ để thực hiện phép nhân ma trận

Chương trình in ra màn hình ma trận p, định thức, hạng cùng nghịch đảo của nó.

Chương trình in ra màn hình ma trận q, tích của ma trận qxp

```

P3: MATRIX
-----
MA TRAN P:
|   -2   3   1   |
|   -4   0   3   |
|    2  -3   4   |
* DINH THUC MA TRAN: 60
* HANG CUA MA TRAN: 3
* MA TRAN NGHICH DAO:
|   0.15  -0.25  0.15  |
|   0.36  -0.16  0.03  |
|    0.2    0    0.2   |

MA TRAN Q:
|    1    3   -2   |
|    4    0   -3   |

MA TRAN QxP:
|   -18    9    2   |
|   -14   21   -8   |
-----
```

6. Video Demo

<https://drive.google.com/file/d/1qLFYsHRmcl7LcLze0kHi7yVXYylxNtzd/view>

Link đến github: <https://github.com/sypham206/Midterm-Project>