

BÀI TẬP VÀ THỰC HÀNH 1

1. Tài khoản github: <https://github.com/thuynttk11/VRA>

2. Họ và tên HV: Nguyễn Thị Thu Thủy

MSHV: CH1601018

Email tài khoản github: thuyntt.11@grad.uit.edu.vn

3.

Vd1: Tạo ngẫu nhiên một số thực trong đoạn [0..1]

Tạo ngẫu nhiên một số nguyên trong đoạn [1 10]

Tạo ngẫu nhiên một ma trận kích thước 1x10 các số nguyên nằm trong đoạn [-10,10]

```
function Example001()
a = rand();
fprintf('\n a = %8.3f',a);
r = randi([1 10]);
fprintf('\n Random tu 1 den 10: %d',r);
matran = randi([-10,10],1,10);
fprintf('\n Size ma tran: %d',size(matran,2));
fprintf('\n Ma tran duoc tao: ');
fprintf('%2d',matran);
end
```

Vd2: Tạo ma trận

```
function Example001()
strMessage = '\n Nhap m: ';
m = input(strMessage);
strMessage = '\n Nhap n: ';
n= input(strMessage);
a = ones(m,n); fprintf('%d',a);
b = zeros(m,n); fprintf('%d',b);
c=eye(m,n);
d=randi([-10,10],m,n);
a(1,1) =5
e = size(a);
end
```

Vd3:Viết chương trình: nhập số nguyên n và tính tổng từ 1 đến số nguyên n với vòng lặp while

```
function Example003()
strMessage = '\n Nhap n: ';
n = input(strMessage);
s = 0;
i = 1;
while i<=n
```

```

        s = s+i;
        i = i+1;
end
fprintf('\n Tong S = 1+2+...+%d la %d.\n',n,s);
end

```

Vd4:Viết chương trình: nhập số nguyên n và tính tổng từ 1 đến số nguyên n với vòng lặp for

```

function Example004()
strMessage = '\n Nhap n: ';
n = input(strMessage);
s = 0;
for i =1:n
    s=s+i;
end
fprintf('\n Tong S = 1 + 2 + ... + %d la %d.\n',n,s);
end

```

Hàm isprime(n): kiểm tra số nguyên n có nguyên tố không (trả về 1 nếu nguyên tố, trả về 0 nếu không nguyên tố)

```

function m = isprime(n)
count = 0;
for i = 1:n
    if(mod(n,i)==0)
        count = count +1;
    end
end
m=(count==2);
end

```

Hàm findnprime(n): trả về một mảng gồm n số nguyên tố đầu tiên.

```

function primeArray = findnprime(n)

count = 0;
i = 2;
primeArray = [];
while (count<n)
    if(isprime(i)==1)
        count = count +1;
        primeArray = [primeArray,i];
    end
    i=i+1;
end

end

```

Hàm saveprime(n): tìm n số nguyên tố đầu tiên và lưu vào tập tin có tên là “prime<n>.mat”

```
function saveprime(n)
rArray = findnprime(n);
strFileName = ['prime',num2str(n),'.mat'];
save(strFileName,'rArray');
end
```

Hàm loadnprime(n): nạp n số nguyên tố từ tập tin có tên “prime<n>.mat”

```
function loadnprime(n)
strFileName = ['prime',num2str(n),'.mat'];
load(strFileName);
rArray
end
```

Hàm loadMNISTImages(filename): trả về ma trận chứa các ảnh của MNIST mỗi ảnh có kích thước 28x28

```
function images = loadMNISTImages(filename)
%loadMNISTImages returns a 28x28x[number of MNIST images] matrix containing
%the raw MNIST images

fp = fopen(filename, 'rb');
assert(fp ~= -1, ['Could not open ', filename, '']);
magic = fread(fp, 1, 'int32', 0, 'ieee-be');
assert(magic == 2051, ['Bad magic number in ', filename, '']);

numImages = fread(fp, 1, 'int32', 0, 'ieee-be');
numRows = fread(fp, 1, 'int32', 0, 'ieee-be');
numCols = fread(fp, 1, 'int32', 0, 'ieee-be');

images = fread(fp, inf, 'unsigned char');
images = reshape(images, numCols, numRows, numImages);
images = permute(images, [2 1 3]);

fclose(fp);

% Reshape to #pixels x #examples
images = reshape(images, size(images, 1) * size(images, 2), size(images, 3));
% Convert to double and rescale to [0,1]
images = double(images) / 255;
end
```

Hàm loadMNISTLabels(filename): trả về ma trận chứa các nhãn ảnh của MNIST

```
function labels = loadMNISTLabels(filename)
%loadMNISTLabels returns a [number of MNIST images]x1 matrix containing
%the labels for the MNIST images
fp = fopen(filename, 'rb');
assert(fp ~= -1, ['Could not open ', filename, '']);
```

```

magic = fread(fp, 1, 'int32', 0, 'ieee-be');
assert(magic == 2049, ['Bad magic number in ', filename, '']);

numLabels = fread(fp, 1, 'int32', 0, 'ieee-be');
labels = fread(fp, inf, 'unsigned char');
assert(size(labels,1) == numLabels, 'Mismatch in label count');

fclose(fp);
end

```

Vấn đề 01: Hàm Recognition001_Digits() nạp dữ liệu train và dữ liệu test của bài toán nhận dạng chữ viết tay

```

function Recognition001_Digits()
fprintf('\n Load du lieu train');
imgTrainAll= loadMNISTImages('./train-images.idx3-ubyte');
lblTrainAll= loadMNISTLabels('./train-labels.idx1-ubyte');
fprintf('\n Load du lieu Test');
imgTestAll=loadMNISTImages('./t10k-images.idx3-ubyte');
lblTestAll=loadMNISTLabels('./t10k-labels.idx1-ubyte');
fprintf('\n Ket thuc .\n');
end

```

Vấn đề 02: Hàm Recognition002_Digits() cho biết số lượng ảnh train và ảnh test sau khi nạp dữ liệu train và dữ liệu test của bài toán nhận dạng chữ viết tay

```

function Recognition002_Digits()
fprintf('\n Load du lieu train');
imgTrainAll= loadMNISTImages('./train-images.idx3-ubyte');
lblTrainAll= loadMNISTLabels('./train-labels.idx1-ubyte');
fprintf('\n Load du lieu Test');
imgTestAll=loadMNISTImages('./t10k-images.idx3-ubyte');
lblTestAll=loadMNISTLabels('./t10k-labels.idx1-ubyte');
nTrainImages=size(imgTrainAll,2);
nTrainLabels=size(lblTrainAll,1);
nTestImages=size(imgTestAll,2);
nTestLabels=size(lblTestAll,1);
nSizeofImage =size(imgTrainAll,1);

fprintf('\n So luong anh train: [%d].',nTrainImages);
fprintf('\n So luong nhan anh train: [%d].1',nTrainLabels);
fprintf('\n So luong anh test: [%d].', nTestImages);
fprintf('\n So luong nhan anh test: [%d].',nTestLabels);
fprintf('\n Kich thuc cua mot anh: [%d].',nSizeofImage);
end

```

Vấn đề 03: Hàm Recognition003_Digits() hiển thị ảnh train thứ nhất và cuối cùng sau khi nạp dữ liệu train và dữ liệu test của bài toán nhận dạng chữ viết tay

```

function Recognition003_Digits()
fprintf('\n Load du lieu train');
imgTrainAll= loadMNISTImages('./train-images.idx3-ubyte');
lblTrainAll= loadMNISTLabels('./train-labels.idx1-ubyte');
fprintf('\n Load du lieu Test');
imgTestAll=loadMNISTImages('./t10k-images.idx3-ubyte');
lblTestAll=loadMNISTLabels('./t10k-labels.idx1-ubyte');

nTrainImages=size(imgTrainAll,2);
figure;
img=imgTrainAll(:,1);
img2D=reshape(img,28,28);
strLabelImage=num2str(lblTrainAll(1));
imshow(img2D);
title(strLabelImage);

figure;
imgLast=imgTrainAll(:,nTrainImages);
img2DLast=reshape(imgLast,28,28);
strLabelImage=num2str(lblTrainAll(nTrainImages));%lblTrainLabelsAll
imshow(img2DLast);
title(strLabelImage);
end

```

Vấn đề 04: Hàm Recognition004_Digits() hiển thị ảnh ngẫu nhiên trong bộ dữ liệu train và bộ dữ liệu test sau khi nạp dữ liệu train và dữ liệu test của bài toán nhận dạng chữ viết tay

```

function Recognition004_Digits()
fprintf('\n Load du lieu train');
imgTrainAll= loadMNISTImages('./train-images.idx3-ubyte');
lblTrainAll= loadMNISTLabels('./train-labels.idx1-ubyte');
fprintf('\n Load du lieu Test');
imgTestAll=loadMNISTImages('./t10k-images.idx3-ubyte');
lblTestAll=loadMNISTLabels('./t10k-labels.idx1-ubyte');

nTrainImages=size(imgTrainAll,2);
nNumber=randi([1 nTrainImages]);

figure;
img=imgTrainAll(:,nNumber);
img2D=reshape(img,28,28);
strLabelImage=num2str(lblTrainAll(nNumber));
strLabelImage=[strLabelImage, '(', num2str(nNumber), ')'];
imshow(img2D);
title(strLabelImage);

nTestImgs=size(imgTestAll,2);
nNumber=randi([1 nTestImgs]);

figure;

```

```

img=imgTestAll(:,nNumber);

img2D=reshape(img,28,28);
strLabelImage=num2str(lblTestAll(nNumber));

strLabelImage=[strLabelImage, '(' ,num2str(nNumber), ') '];
imshow(img2D);
title(strLabelImage);
end

```

Vấn đề 05: Hàm Recognition005_Digits()

- Nạp dữ liệu train của bài toán nhận dạng chữ viết tay
- Xây dựng một model từ tập dữ liệu train bằng cách sử dụng thuật toán kNN
- Nạp dữ liệu test của bài toán nhận dạng chữ viết tay. Lấy ngẫu nhiên một ảnh từ tập dữ liệu test và sử dụng model của câu trên để predict (phân tích).
- Kiểm tra kết quả tiên đoán có đúng không

```

function Recognition005_Digits()
fprintf('\n Load du lieu train');
imgTrainAll= loadMNISTImages('./train-images.idx3-ubyte');
lblTrainAll= loadMNISTLabels('./train-labels.idx1-ubyte');

Mdl=fitcknn(imgTrainAll',lblTrainAll);

fprintf('\n Load du lieu Test');
imgTestAll=loadMNISTImages('./t10k-images.idx3-ubyte');
lblTestAll=loadMNISTLabels('./t10k-labels.idx1-ubyte');

nTestImgs=size(imgTestAll,2);
nNumber=randi([1 nTestImgs]);
imgTest=imgTestAll(:,nNumber);

lblPredictTest=predict(Mdl,imgTest');
lblImageTest=lblTestAll(nNumber);
figure;
img2D=reshape(imgTest,28,28);
imshow(img2D);
strLabelImage='Ban dau';
strLabelImage=[strLabelImage,num2str(lblTestAll(nNumber)),'.']; %lblTestLabelAll
strLabelImage=[strLabelImage,'Du doan:'];
strLabelImage=[strLabelImage,num2str(lblPredictTest),'.'];
if(lblPredictTest==lblImageTest)
    strLabelImage=[strLabelImage,'Ket qua dung.'];
else
    strLabelImage=[strLabelImage,'Ket qua sai.'];
end
title(strLabelImage);
end

```

BÀI TẬP VÀ THỰC HÀNH 2

Q1 - Hãy cho biết câu lệnh tạo ngẫu nhiên một số trong đoạn [1 200].

`R=randi([1 200])`

Q2 - Hãy cho biết câu lệnh truy cập phần tử dòng 3, cột 5 của ma trận A.

`a(3,5);`

Q3 - Hãy cho biết lệnh tạo ma trận A kích thước 100x200 và các giá trị đều là zero (0).

`A=zeros(100 200);`

Q4 - Hãy cho biết lệnh lấy số dòng của ma trận A.

`Size(A,1)`

Q5 - Hãy cho biết lệnh lấy vector cột 10 của ma trận A.

`A(:,10)`

Q6 - Hãy cho biết lệnh lấy vector dòng 10 của ma trận A.

`A(10,:)`

Q7 - Hãy cho biết lệnh chuyển vector dòng thành ma trận 28x28.

`Reshape(A,28,28)`