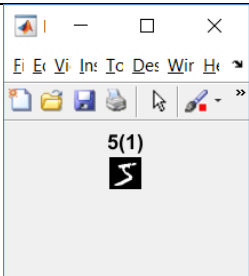
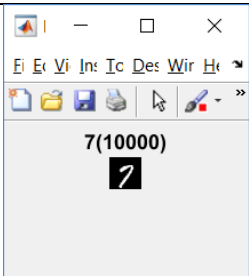
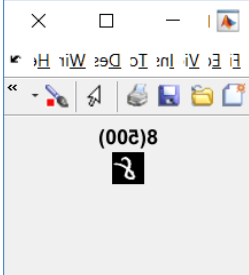
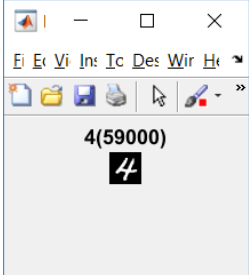
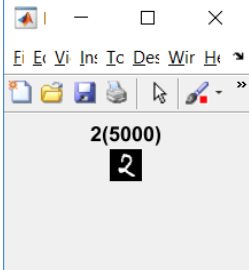


Q1 - Hãy viết function hiển thị ảnh có thứ tự là n (n là tham số) cùng label tương ứng trong tập huấn luyện (train) của tập dữ liệu MNIST. Paste code vào bài thực hành và lập bảng cho biết kết quả khi chạy với n=1, 500, 5000, 10000, 59000.

```
function Q1(nNumber)
    imgTrainAll = loadMNISTImages ('train-images.idx3-ubyte');
    lblTrainAll = loadMNISTLabels ('train-labels.idx1-ubyte');
    figure;
    img = imgTrainAll(:, nNumber);
    img2D = reshape (img, 28, 28);
    strLabelImage = num2str(lblTrainAll(nNumber)); strLabelImage =
    [strLabelImage, '(', num2str(nNumber), ')']; imshow(img2D);
    title(strLabelImage);
end
```

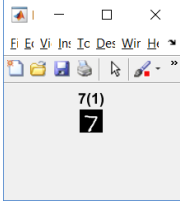
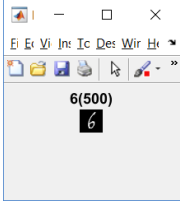
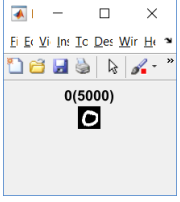
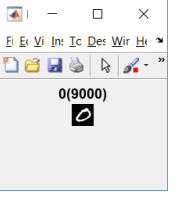
### Kết quả

1		10000	
500		59000	
5000			

Q2 - Hãy viết function hiển thị ảnh có thứ tự là n (n là tham số) cùng label tương ứng trong tập test của tập dữ liệu MNIST. Paste code vào bài thực hành và lập bảng cho biết kết quả khi chạy với n=1, 500, 5000, 9000.

```
function Q2(nNumber)
    imgTestAll = loadMNISTImages('t10k-images.idx3-ubyte');
    lblTestAll = loadMNISTLabels('t10k-labels.idx1-ubyte');
    figure;
    img = imgTestAll(:, nNumber);
    img2D = reshape(img, 28, 28);
    strLabelImage = num2str(lblTestAll(nNumber));

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

n	1	500	5000	9000
<b>Kết quả</b>				

Q 3 - Hãy viết function thống kê số lượng các ảnh tương ứng với các label trong tập huấn luyện (train) của tập dữ liệu MNIST. Paste code vào bài thực hành và lập bảng kết quả khi chạy (nên xuất dưới dạng csv để tiện import thành bảng).

```
function result = Q3(labels, filename)
    %Create matrix 10x2 with value = 0
    result = zeros([10 2]);
    % add label 0->9
    i = 0;
    while (i < 10)
        result(i+1,1) = i;
        i = i + 1;
    end
    % count label i = 1;
    while (i <= size(labels, 1))
        label = labels(i);
        result(label+1,2) = result(label+1,2) + 1;
        i = i + 1;
    end
    % export to csv file
    csvwrite(filename, result);
end
```

```
>> lblTrainAll = loadMNISTLabels ('train-labels.idx1-ubyte');
>> result = Q3(lblTrainAll, 'Q3.csv')

result =
```

0	5923
1	6742
2	5958
3	6131
4	5842
5	5421
6	5918
7	6265
8	5851
9	5949

Q 4 - Hãy viết function thống kê số lượng các ảnh tương ứng với các label trong tập test của tập dữ liệu MNIST. Paste code vào bài thực hành và lập bảng kết quả khi chạy (nên xuất dưới dạng csv để tiện import thành bảng). Code: Tương tự Q3.

```
>> lblTestAll = loadMNISTLabels ('t10k-labels.idx1-ubyte');
>> result = Q3(lblTrainAll, 'Q4.csv')

result =
```

0	5923
1	6742
2	5958
3	6131
4	5842
5	5421
6	5918
7	6265
8	5851
9	5949

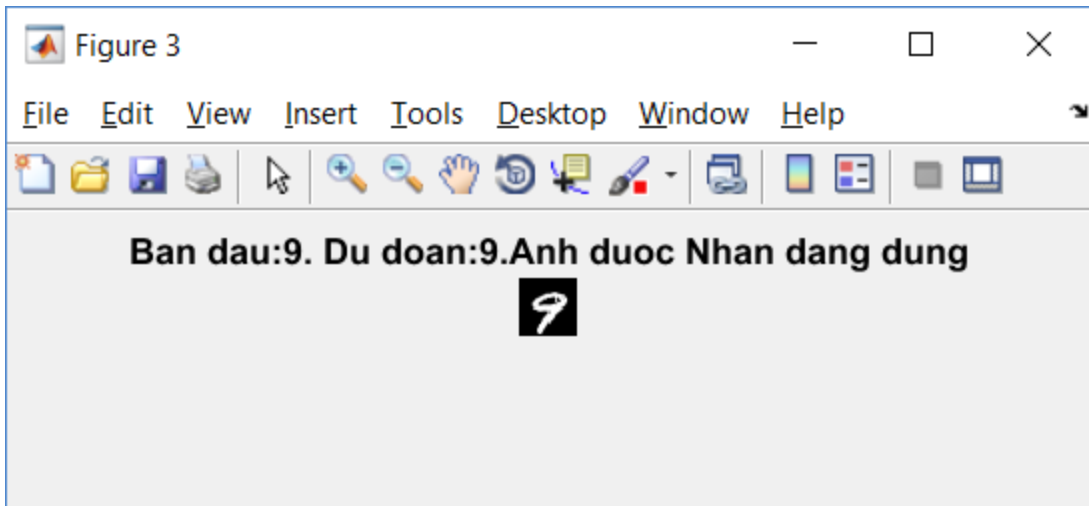
Q 5 - Hãy viết function trả về kết quả nhận dạng của ảnh trong tập test có thứ tự là n (n là tham số, nằm trong đoạn [1, 10000]). Paste code vào bài thực hành và lập bảng kết quả khi chạy với n = 5, 500, 900.

```
function Q5(n)
    imgTrainAll = loadMNISTImages('train-images.idx3-ubyte');
    lblTrainAll = loadMNISTLabels('train-labels.idx1-ubyte');
    Mdl = fitcknn(imgTrainAll, lblTrainAll);
    imgTestAll = loadMNISTImages('t10k-images.idx3-ubyte');
    imgTest = imgTestAll(:,n);
    lblPredictTest = predict(Mdl, imgTest);
    fprintf('Ket qua nhan dang anh thu %d trong tap test la: %d\n', n,
    lblPredictTest);
end
```

```
Command Window
>> Q5(5)
Ket qua nhan dang anh thu 5 trong tap test la: 4
>> Q5(500)
Ket qua nhan dang anh thu 500 trong tap test la: 6
>> Q5(900)
Ket qua nhan dang anh thu 900 trong tap test la: 8
fx >> |
```

Q6 - Hãy viết function với tham số đầu vào n là thứ tự của ảnh trong tập test- sau đó hiển thị ảnh tương ứng - rồi hiển thị kết quả nhận dạng - rồi cho biết kết quả nhận dạng là đúng hay sai khi so khớp với label của tập test.

```
function Q6(n)
    imgTrainAll = loadMNISTImages('train-images.idx3-ubyte');
    lblTrainAll = loadMNISTLabels('train-labels.idx1-ubyte');
    Mdl = fitcknn(imgTrainAll, lblTrainAll);
    imgTestAll = loadMNISTImages('t10k-images.idx3-ubyte');
    lblTestAll = loadMNISTLabels('t10k-labels.idx1-ubyte');
    imgTest = imgTestAll(:, n);
    lblPredictTest = predict(Mdl, imgTest);
    lblImageTest = lblTestAll(n);
    figure;
    img2D = reshape(imgTest, 28, 28);
    imshow(img2D);
    strLabelImage = 'Anh Test: ';
    strLabelImage = [strLabelImage, num2str(lblImageTest), '.'];
    strLabelImage = [strLabelImage, 'Du doan:'];
    strLabelImage = [strLabelImage, num2str(lblPredictTest), '.'];
    if(lblPredictTest == lblImageTest)
        strLabelImage = [strLabelImage, 'Anh duoc Nhan dang dung'];
    else
        strLabelImage = [strLabelImage, 'Anh bi nhan dang sai.'];
    end
    title(strLabelImage);
end
```



Q 7 - Hãy viết function đếm số lượng các ảnh có label là n (n là tham số) bị nhận dạng sai theo thuật toán knn. Paste code của function đã chạy được vào bài thực hành và lập bảng kết quả khi chạy với n= 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

```
function Q7(n)
    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');

    numLabel = size(lblTestAll, 1); result = 0;

    for i=1:numLabel
        if(lblTestAll(i) == n)
            imgTest = imgTestAll(:, i);
            if(predict(Mdl, imgTest') ~= lblTestAll(i))
                result = result + 1;
            end
        end
    end
    fprintf('\nSo luong anh co label %d bi nhan dang sai la: %d\n', n, result);
end
```

```
>> Q7(3)

Load du lieu train
Load du lieu test
So luong anh co label 3 bi nhan dang sai la: 40
```

Q 7\* - Tương tự bài tập Q7 - nhưng lập bảng confusion matrix.

```
function Q7n(n)
    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');

    numLabel = size(lblTestAll, 1); result = 0;

    for i=1:numLabel
        if(lblTestAll(i) == n)
            imgTest = imgTestAll(:, i);
            if(predict(Mdl, imgTest') ~= lblTestAll(i))
                result = result + 1;
            end
        end
    end
    fprintf('\nSo luong anh co label %d bi nhan dang sai la: %d\n',
n,result);
end
```

Q8\*\* (Tương đương 50% đồ án môn học nếu kết hợp với Q7\*) - Hãy viết function tính độ chính xác của thuật toán knn với các tham số khác nhau của hàm knn (ví dụ số lượng nearest neighbors, độ đo distance). Paste code vào bài thực hành và lập bảng kết quả.

```
function Q8_Advance(k)
    imgTrainAll = loadMNISTImages('./train-images.idx3-ubyte');
    lblTrainAll = loadMNISTLabels('./train-labels.idx1-ubyte');

    Mdl = fitcknn(imgTrainAll, lblTrainAll, 'NumNeighbors', k);

    imgTestAll = loadMNISTImages('./t10k-images.idx3-ubyte');
    lblTestAll = loadMNISTLabels('./t10k-labels.idx1-ubyte');

    lblResult = predict(Mdl, imgTestAll');
    result = (lblResult == lblTestAll);
    nCount = sum(result);

    accuracy = nCount*100/size(imgTestAll, 2);
    fprintf('\nDo chinh xac cua thuat toan knn voi %d nearest
neighbors la %d\n', k, accuracy);
end
```