CS212 Topics in Computing 2
2019, Term 2 Second Half: Lab 1

Weights in your Mark
Details can be found in Lab 1. This assignment is 5% worth of your cs212 mark.

Assessment & Submission
The report for this lab assignment with your screenshots needs to be uploaded to myPlace before 12pm on Monday March 11th as a single PDF file. Get in touch with the lecturer ASAP if you are unable to meet those requirements due to your individual circumstances or technical glitches on our lab computers.

You can work on this assignment in the groups of your choice no larger than 3 students. But make sure everyone retains the copy of the code and uploads the report, which will be marked individually. You don't need to state the names of the students in the group.

Assessment Criteria
See lab 1.

Instructions
Task 1. (20%)
Using our Lecture 3 slides and exercise as guidance, fill in the table below with the steps of the perceptron training algorithm applied to boolean AND function. Use learning rate of 1 and initial weights (1,1,0). Hint: it should converge in no longer than 4 epochs.

X1	X2	Y
0	0	0
1	0	0
0	1	0
1	1	1

	X1	X2	X3	label	prediction	error	new W1	new W2	new W3
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									

Task 2. (30%)

Implement a Java code that executes the steps in Task 1. It should print all the variables used in the table approximately using the format as in the following of learning boolean OR function (starting from zero weights):

```
epoch 1 training on 0 0 1 label = 0 prediction = 1 error = -1 W1 = 0.0 W2 = 0.0 W3 = -1.0
epoch 1 training on 1 0 1 label = 1 prediction = 0 error = 1 W1 = 1.0 W2 = 0.0 W3 = 0.0
epoch 1 training on 0 1 1 label = 1 prediction = 1 error = 0 W1 = 1.0 W2 = 0.0 W3 = 0.0
epoch 1 training on 1 1 1 label = 1 prediction = 1 error = 0 W1 = 1.0 W2 = 0.0 W3 = 0.0
epoch 2 training on 0 0 1 label = 0 prediction = 1 error = -1 W1 = 1.0 W2 = 0.0 W3 = -1.0
epoch 2 training on 1 0 1 label = 1 prediction = 1 error = 0 W1 = 1.0 W2 = 0.0 W3 = -1.0
epoch 2 training on 0 1 1 label = 1 prediction = 0 error = 1 W1 = 1.0 W2 = 1.0 W3 = 0.0
epoch 2 training on 1 1 1 label = 1 prediction = 1 error = 0 W1 = 1.0 W2 = 1.0 W3 = 0.0
epoch 3 training on 0 0 1 label = 0 prediction = 1 error = -1 W1 = 1.0 W2 = 1.0 W3 = -1.0
epoch 3 training on 1 0 1 label = 1 prediction = 1 error = 0 W1 = 1.0 W2 = 1.0 W3 = -1.0
epoch 3 training on 0 1 1 label = 1 prediction = 1 error = 0 W1 = 1.0 W2 = 1.0 W3 = -1.0
epoch 3 training on 1 1 1 label = 1 prediction = 1 error = 0 W1 = 1.0 W2 = 1.0 W3 = -1.0
```

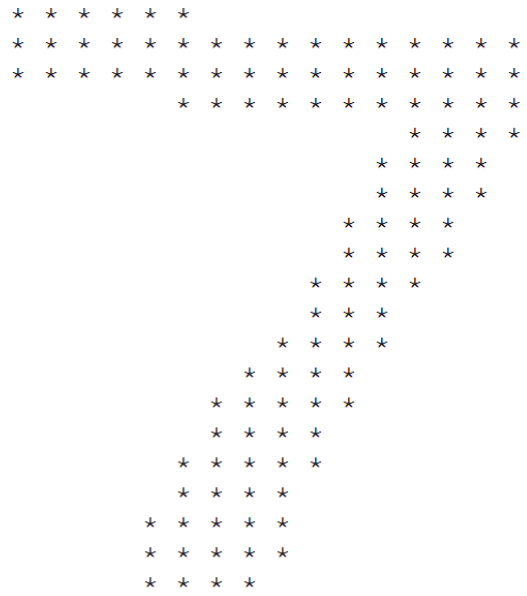
Make sure the numbers from Tasks 1 and 2 match. Take the screenshots covering your code and your output and add them to your report.

Task 3. (50%) Working with MNIST images

This task is important since it will give you the code to work with MNIST images, so you can later use them to train and test your models that can recognize hand-written digits.

Using the guidance in the slides on the format of MNIST dataset, write a Java program that reads the posted file test-10.txt and prints all the images, so the printout of each image looks similar to this:

7



You can see the complete printout posted as image-print-test-10.txt

Note: we print the label (“7” for the first image) prior to the image. Then, for each pixel that has a different value from 0, we print a star (‘*’), otherwise we print a single empty space (‘ ‘).
Hint: in order to read and understand each line in the image file, you may find it helpful to use *split()* function in Java. Specifically, to handle MNIST format, you will need to split by tabs or empty spaces, which can be done by using a regular expression *split("[\t]+").*
It is important that your program can work with any number of images in the file.
Hint: to find the number of images in the file, your program may count the total number of lines in the files first, then divide it by 28.
Take the screenshots covering your code and your output (at least one image) and add them to your report.