1. Consider a learning task where we want to fit a model to a set of 10,000 labeled training cases.

Assume that we have a database with data of the form

Table 1: example of data

x1	x2	y
4	1	2
2	8	-14
1	0	1
3	2	-1
1	4	7
6	7	-8

We want to use TensorFlow to create a classifier for this data. We will develop a model of the form:

$$\hat{y} = w_1 x_1 + w_2 x_2 + b.$$

Based on the data, our task is to compute the labels for the parameters $W = [w_1 \quad w_2]$ and b. Our objective is to train a neural network to minimise the loss function $L = (\hat{y} - y)^2$.

The update formula for any parameter p is given by $p' = p - \eta \, \partial L / \partial p$,

where

- p—current value
- p'—value after update
- η —learning rate, set to 0.05
- $\partial L / \partial p$—gradient, i.e. partial differential of L w.r.t. b

We use backpropagation to update the weights $W = [w_1 \quad w_2]$ and b. We initialize our weights randomly such that $W = [w_1 \quad w_2] = [-0.017 \quad -0.048 \quad]$ and b=0.

a) Draw the computation graph for this model.
b) If we assign a CPU to every operator in the computation graph, state the assignment of the computation graph entities to CPUs.
c) For the first data item (4, 1, 2) in Table 1, compute the loss.
d) Compute the partial derivatives necessary to update the weights.
e) Use the update formula to update the weights $W = [w_1 \quad w_2]$ and b.