Problem 41. $x \to y, x \to z$, $\bar{x} = \bar{y} \cdot \frac{\partial y}{\partial x} + \bar{z} \cdot \frac{\partial z}{\partial x}$

$\bar{L} = \frac{\partial L}{\partial L} = 1$. $\bar{y}_1 = \bar{L} \cdot \frac{\partial L}{\partial y_1}$ $\bar{y}_2 = \bar{L} \cdot \frac{\partial L}{\partial y_2}$.

$\bar{z}_1 = \bar{y}_1 \cdot \frac{\partial y_1}{\partial z_1} + \bar{y}_2 \cdot \frac{\partial y_2}{\partial z_1}$ $\bar{z}_2 = \bar{y}_1 \cdot \frac{\partial y_1}{\partial z_2} + \bar{y}_2 \cdot \frac{\partial y_2}{\partial z_2}$

$\bar{w}_{11} = \bar{z}_1 \cdot \frac{\partial z_1}{\partial w_{11}}$ $\bar{w}_{12} = \bar{z}_1 \cdot \frac{\partial z_1}{\partial w_{12}}$ $\bar{b}_1 = \bar{z}_1 \cdot \frac{\partial z_1}{\partial b_1}$.

$\bar{w}_{11}$ → $\bar{w}_{21}$ $\bar{w}_{12}$ → $\bar{w}_{22}$ $\bar{b}_1$ → b_2

Problem 26. backprop provides gradients. that are used for GD.

α too high = too much change, unable to converge.

$w - \alpha \cdot \nabla = w - \alpha \cdot \frac{\partial y_i}{\partial z_i}$

$w_{11} - \alpha \cdot \nabla = w_{11} - \alpha \frac{\partial z_i}{\partial w_{11}}$

α too low = convergence takes a long time.

Problem 27:

$x_i = \cancel{d \times 1}$ $1 \times d$

input layer neurons = d. output layer after 1st layer: ✿

2nd layer neurons = ✿, output layer after 2nd layer = ★

3rd layer neurons = ★,

⋮

N layer

Output layer = input X, outplayer's final output = 10, dimensionality from N-1's output or the # of classes.

Problem 28. In essence, GD is used to train the MLP. (the explanation on)
so α's being high/low doesn't change if/when the architecture is MLP/CNN/others. Check Problem 26.

Problem 29. backprop v. sgd.
sgd = using 1 sample at a time for training
gd = using entire training set at a time for training.

Problem 30. Introduces non-linearity to the otherwise linear layers in MLP.

Problem 31. $b^{(\text{filter})} = [1, 1, 2]$, $a^{(\text{vector/input})} = [2, -1, 1]$.

$b * a = 1 \times [2, -1, 1, 0, 0]$.
$+ 1 \times [0, 2, -1, 1, 0]$
$+ 2 \times [0, 0, 2, -1, 1]$.
$= [\_, \_, \_, \_, \_]$ } $d = 5$.

Problem 32 ① $x = [-1, 0, 1, 2, 3]$ ② $x = [1, 1, 3, 4, 5]$.

ReLU(①) = [0, 0, 1, 2, 3] ReLU(②) = [1, 2, 3, 4, 5].

Problem 33: kernels = applying transformations onto input, so that we get different feature maps.

Multiple kernels = multiple representations.

Problem 34. Dropout: randomly deactivate certain neurons
Normalization/LayerNorm.

Problem 35. CNN lecture 21-23.

max pooling: taking the maximum of a given set of activations

avg pooling: " " average of a given region

Problem 36: self attention: $Q \cdot K^T$

Q: a version (a.k.a. linearly processed "Hi How Are You") → different.

K: a version (a.k.a. linearly processed " ") → same

Problem 37. Multi-Head Attention: multiple scaled dot product attention.

Attention slides 15.

Problem 38. ① the self-attention needs to be masked in the decoders.

② the decoder has two distinct "blocks" of multi-head attention.

Problem 39. Image Captioning: Look at (pay attention to) a subset region of the input picture to produce one word/phrase at a time.

problem 40. ① Get rid of the decoder

② add linear layers to the encoder

optional ③ do a softmax, assuming the dimensionality is correct

⇒ if not, address it with linear layer(s).

④. get the output