CERTLABZ CERTIFICATION BOARD

Certified Deep Learning Engineer (CDLE)

Ten Sample Questions

These ten questions are written at the difficulty of the live exam. Six carry a figure, a data table, a configuration or a code listing; four are plain scenario questions, as on the live paper. Every one is a scenario a working deep learning engineer would meet, and none of them appears in the live question bank.

The live exam is multistage adaptive: how you score on one stage selects the difficulty of the next, so no two candidates sit the same paper. It is a performance-based exam: each form can carry up to 5 to 12 performance-based questions alongside the multiple-choice questions, which means you build, sequence or complete something rather than only choosing from a list. Answers and full explanations follow each question here; the live exam does not disclose item-level answers.

Question 1 — Domain 3: Architectures — DATA

Emma Wilson matches data to a fitting architecture.

Data / task	Architecture
Images	CNN
Sequences with state	RNN / LSTM
Long-range dependencies in text	Transformer
Learn compressed representations	Autoencoder

Table 1.1 Data and architectures.

A task needs to understand long documents where distant words matter. Which architecture fits best today?

A. A shallow CNN, because convolutions over the text will always capture the relationships between distant words in a long document.

B. A transformer, because self-attention models long-range dependencies across the whole sequence and trains efficiently in parallel.

C. A single linear layer, because a linear model is sufficient to relate distant words in a document without any attention mechanism at all.

D. k-means clustering, because grouping the words by similarity is the most effective way to understand a long document with distant dependencies.

Correct answer: B
Transformers use self-attention to model relationships across the whole sequence, which suits long documents where distant words matter, and they parallelise well. A shallow CNN has a limited receptive field, a single linear layer cannot capture such structure, and k-means is a clustering method, not a document-understanding model.

Question 2 Domain 2: Training Deep Networks CONFIG

James Carter sees this training behaviour.

```
loss: oscillates wildly, sometimes rising, never settling
learning rate: 1.0 (very high)
optimiser: SGD
```

Listing 2.1 Training behaviour and settings.

What is the likely cause, and the fix?

A. The learning rate is too low, so raising it further will let the loss settle and the network finally converge to a good solution.

B. The learning rate is too high, so updates overshoot the minimum; lower it (or use a schedule) so training can converge.

C. The dataset is too large, so removing most of the training data will stabilise the loss and allow the model to converge properly.

D. The optimiser is broken, so the only fix is to train for far more epochs at the same high learning rate until the loss eventually settles.

Correct answer: B
A loss that oscillates and diverges with a learning rate of 1.0 is the classic sign of a learning rate that is too high, so updates overshoot the minimum. Lowering it, or using a schedule, lets training converge. Raising it further, cutting the data, or training longer at the same rate does not address the overshoot.

Question 3 Domain 4: Regularization, Tuning and Evaluation DATA

Olivia Davis observes the following after training.

Metric	Value
Training accuracy	99%
Validation accuracy	70%
Trend	Val accuracy falling as training continues
Dataset	Fairly small

Table 3.1 Training versus validation.

What is the diagnosis, and which fixes fit?

A. Underfitting; the fix is to add many more layers and parameters and to train for far more epochs on the same small dataset.

B. Overfitting; apply regularisation such as dropout and weight decay, use data augmentation or early stopping, or get more data.

C. The model is performing well, because a 99% training accuracy shows it has learned the task and the validation figure can be ignored.

D. A hardware fault; the gap between training and validation accuracy is caused by the GPU and will disappear on different hardware entirely.

Correct answer: B
A large gap between high training accuracy and falling validation accuracy on a small dataset is overfitting. The fixes are regularisation (dropout, weight decay), data augmentation, early stopping or more data. Adding capacity worsens it, the validation figure cannot be ignored, and the gap is not a hardware fault.

Question 4

Domain 4: Regularization, Tuning and Evaluation

DATA

Liam Anderson matches regularisation techniques to what they do.

Technique	Effect
Dropout	Randomly disables units in training
Weight decay	Penalises large weights
Early stopping	Stops when validation stops improving
Augmentation	Expands effective training data

Table 4.1 Regularisation techniques.

Which statement about using these techniques is correct?

A. These techniques increase overfitting, so they should be removed whenever a deep network fails to generalise to unseen validation data.

B. They all help reduce overfitting, and several can be combined to improve a deep network's generalisation to unseen data.

C. Only one may ever be used at a time, because combining any two regularisation techniques always cancels out their individual benefits.

D. They are only relevant during inference, so they are applied when the model serves predictions rather than while it is being trained.

Correct answer: B
Dropout, weight decay, early stopping and augmentation all reduce overfitting, and they can be combined to improve generalisation. They do not increase overfitting, they are not mutually exclusive, and they act during training (dropout, for example, is disabled at inference), not only at serving time.

Question 5

Domain 5: Frameworks, Hardware and Deployment

CONFIG

Ava Thompson must train a large model that will not fit on a single GPU.

```
error : out of GPU memory during training
model : large; batch size 512
options: reduce batch, gradient accumulation, mixed precision,
         distribute across GPUs
```

Listing 5.1 The memory problem.

Which approach is reasonable?

A. Ignore the out-of-memory error and rerun the same configuration repeatedly until the training happens to fit into GPU memory by chance.

B. Reduce the batch size, use gradient accumulation and mixed precision, or distribute training across multiple GPUs to fit the model.

C. Move all training to the CPU, because a CPU has effectively unlimited memory and will train the large model just as fast as the GPU would.

D. Delete most of the training data, because a much smaller dataset is the only way to make a large model fit within a single GPU's memory.

Correct answer: B
Out-of-memory during training is handled by reducing the batch size, using gradient accumulation and mixed precision, or distributing across GPUs. Rerunning the same config will not help, CPUs are far slower for this and not unlimited, and cutting the data harms the model rather than solving the memory constraint properly.

Question 6

Domain 3: Architectures

DATA

A team compares two ways to model images.

Approach	Detail
Fully connected	A unique weight for every pixel-to-neuron pair
CNN	Shared convolutional filters across the image

Table 2.1 Two approaches to image modelling.

Why does the CNN generally need far fewer parameters and work better for images?

A. Because a CNN uses no weights at all, whereas a fully connected network must learn a separate unique weight for every single pixel in the whole image.

B. Because weight sharing reuses convolutional filters across the image instead of a unique weight per pixel pair, cutting parameters and capturing spatial structure.

C. Because a fully connected network cannot process images in any form, so a CNN is the only architecture capable of taking an image as input at all.

D. Because CNNs always have many more parameters than fully connected networks, and the extra parameters are what make them more accurate on images.

Correct answer: B
A CNN shares convolutional filters across the whole image rather than learning a unique weight for each pixel-to-neuron connection, which drastically reduces parameters and captures spatial structure. CNNs do use weights, fully connected networks can technically take images (just inefficiently), and CNNs have fewer, not more, parameters for this.

Question 7 Domain 1: Neural Network Foundations

Why does a deep network need non-linear activation functions to benefit from having many layers?

A. It does not; a network of stacked linear layers with no activations is just as powerful as one using non-linear activation functions.

B. Without non-linearity, any stack of linear layers is equivalent to a single linear layer, so depth adds no representational power; activations enable it.

C. Non-linear activations are only needed to speed up training, and a deep network of purely linear layers would learn exactly the same complex functions.

D. Activations exist only to store the network's data between layers, so their type has no effect on what functions a deep network can represent.

Correct answer: B
Composing linear layers yields another linear function, so without non-linear activations a deep stack collapses to a single linear layer and depth adds nothing. Non-linear activations are what let deep networks represent complex functions. They are not merely for speed or storage, and their absence genuinely limits the network.

Question 8 Domain 2: Training Deep Networks

A very deep network trains poorly: early layers barely update while later ones do. What is happening, and which advances address it?

A. The learning rate is simply too low everywhere, so raising it uniformly across the whole network will make the early layers train normally.

B. Vanishing gradients: gradients shrink through many layers; ReLU-type activations, good initialisation, normalisation and residual connections help.

C. The dataset is too large for a deep network, so the only remedy is to drastically reduce the amount of training data used for the model.

D. The GPU is failing under the load of a deep network, so the fix is to move training to a larger GPU without changing the network at all.

Correct answer: B
Early layers barely updating in a deep network is the vanishing-gradient problem, where gradients shrink as they propagate back. ReLU-type activations, good initialisation, normalisation and residual connections were the advances that made very deep networks trainable. It is not simply a uniform learning-rate issue, a data-size problem, or a hardware fault.

Question 9

Domain 4: Regularization, Tuning and Evaluation

A team repeatedly tunes their architecture against the test set until the test score looks excellent. Why is the reported score untrustworthy?

A. It is trustworthy, because repeatedly improving the score on the test set is the standard and recommended way to tune a deep learning model.

B. The test set has effectively been used for tuning, so it no longer estimates unseen performance; tune on validation and use the test set once at the end.

C. The only problem is that they did not tune for enough iterations, so continuing to tune against the test set for longer would make the score valid.

D. The score is untrustworthy solely because the model is too small, so switching to a larger model while keeping the same tuning process fixes it.

Correct answer: B

Repeatedly tuning against the test set leaks it into the selection process, so the score becomes optimistic and no longer reflects unseen data. The fix is to tune on a validation set (or cross-validation) and reserve the test set for a single final check. More iterations against the test set, or a bigger model, does not restore validity.

Question 10

Domain 6: Generative, Transfer Learning and Responsible DL

A team fine-tunes a large pretrained model for a hiring-related task and finds it carries bias from its training data. What is the responsible approach?

A. Assume the pretrained model is neutral, because a model trained on large, broad datasets cannot carry bias into a fine-tuned downstream task.

B. Assess and mitigate the bias for this specific use, testing outcomes across groups, and reconsider or add safeguards given the high-impact hiring context.

C. Ignore the bias and deploy, because the model's overall accuracy on the hiring task is high and that is the only factor that matters for deployment.

D. Delete the affected groups from the evaluation, because removing the lower-scoring groups makes the results look consistent and ready to ship.

Correct answer: B

A pretrained model can carry bias from its training data into a downstream task, so the responsible approach is to assess and mitigate that bias for the specific use, test outcomes across groups, and for a high-impact hiring context reconsider or add safeguards. Assuming neutrality, ignoring the bias for accuracy, or dropping groups from the evaluation would cause and conceal real harm.

Published by the CertLabz Certification Board, a division of CertLabz LLC. Companion to objectives document CDLE-001. https://certlabz.com/certifications/certified-deep-learning-engineer.html