1. **Explain convolutional neural network, and how does it work?**

A convolutional neural network (CNN) is a type of neural network designed for image recognition and processing. It is made up of multiple layers of interconnected nodes, each of which performs a specific mathematical operation on the input data. The layers of a CNN typically include an input layer, one or more convolutional layers, a pooling layer, and an output layer.

2. **How does refactoring parts of your neural network definition favor you?**

Overall, refactoring parts of a neural network definition can help to improve the performance, reduce the risk of overfitting, and make the network more modular and easier to maintain. These benefits can make the network more effective and useful for a wide range of applications.

3. **What does it mean to flatten? Is it necessary to include it in the MNIST CNN? What is the reason for this?**

To flatten in the context of a neural network means to convert a multi-dimensional array of data into a one-dimensional vector. This is often done as a preprocessing step before feeding the data into a fully connected layer of the network.

In the MNIST dataset, the images are 28x28 pixels, which means that each image is represented by a 28x28 array of pixel values. In order to feed this data into a fully connected layer of a neural network, it must first be flattened into a one-dimensional vector of 784 values (28x28=784). This is necessary because fully connected layers expect input data to be in the form of a one-dimensional vector.

Including a flattening step in the MNIST CNN is necessary because it allows the network to process the input images and make predictions based on the learned features. Without flattening the input images, the network would not be able to correctly interpret the data and make accurate predictions.

Overall, flattening is a necessary step in the MNIST CNN because it allows the network to correctly process the input images and make predictions based on the learned features. It is an important part of the network architecture and plays a crucial role in the performance of the model.

4. **What exactly does NCHW stand for?**

NCHW is an abbreviation that stands for "batch size, channels, height, width". It is a common convention used to specify the dimensions of a tensor, which is a multi-dimensional array of data used in deep learning.

The batch size refers to the number of samples in the tensor, the channels refer to the number of channels in the tensor (e.g. the number of color channels in an image), the height refers to the height of the tensor, and the width refers to the width of the tensor. For example, a tensor with dimensions NCHW=(32, 3, 28, 28) would have a batch size of 32, 3 channels, a height of 28, and a width of 28.

5. Why are there 7\*7\*(1168-16) multiplications in the MNIST CNN's third layer?

In the MNIST CNN, the third layer is a fully connected layer that has 1168 nodes. Each node in this layer is connected to all of the nodes in the previous layer, which has a shape of 7x7x16 (7 rows, 7 columns, and 16 channels). This means that there are 7x7x16 = 1168 connections between the nodes in the previous layer and the nodes in the third layer.

When calculating the number of multiplications in this layer, it is necessary to multiply the number of connections by the number of nodes in the third layer. This is because each connection between a node in the previous layer and a node in the third layer requires a multiplication operation. Therefore, there are 7x7x(1168-16) = 7x7x1152 = 70592 multiplications in the third layer of the MNIST CNN.

6.**Explain definition of receptive field?**

In the context of a convolutional neural network (CNN), the receptive field of a node in a layer refers to the region of the input data that the node can "see" or process. In other words, it is the spatial extent of the input data that is used to compute the output of the node.

7. **What is the scale of an activation's receptive field after two stride-2 convolutions? What is the reason for this?**

The scale of an activation's receptive field after two stride-2 convolutions is determined by the size of the filters used in the convolutional layers and the stride of the filters. If the filters have a size of 3x3 and a stride of 2, the receptive field of the activations in the output tensor will be 3x3 after the first convolution and 6x6 after the second convolution.

The reason for this is that stride-2 convolutions reduce the size of the output tensor by half in each dimension. For example, if the input tensor has a shape of 28x28, the output tensor from the first convolution will have a shape of 14x14, and the output tensor from the second convolution will have a shape of 7x7. Since the filters have a size of 3x3, the receptive field of the activations in the output tensor will be 3x3 after the first convolution and 6x6 after the second convolution.

8. **What is the tensor representation of a color image?**

The tensor representation of a color image is a multi-dimensional array of pixel values, where each element of the array represents the color of a single pixel in the image. In general, a color image can be represented by a tensor with three dimensions, where the dimensions correspond to the height, width, and color channels of the image.

For example, an RGB color image with dimensions 256x256 pixels would be represented by a tensor with dimensions 256x256x3, where the first two dimensions correspond to the height and width of the image, and the third dimension corresponds to the red, green, and blue color channels of the image. The value of each element in the tensor would represent the intensity of the corresponding color channel at that pixel location.

9. **How does a color input interact with a convolution?**

When a color input image is processed by a convolutional layer in a neural network, the color information is taken into account by the convolutional filters used in the layer. Each filter in the convolutional layer is a small matrix of weights that is applied to a region of the input image to extract features from it.

In the case of a color input image, the convolutional filters will typically have multiple channels, one for each color channel in the input image. For example, if the input image is an RGB image with red, green, and blue color channels, the convolutional filters will have three channels, one for each color channel.