1. (3 Punkte) Berechne die Faltung (ohne Padding, Stride 1) des Bildes B mit dem Filter K, wie sie ein CNN rechnet: Der Filter wird über das Bild geschoben, an jeder Stelle werden die übereinander liegenden Zahlen multipliziert und addiert.

$$B = \begin{pmatrix} 1 & 2 & 0 & 1 & 3 \\ 0 & 1 & 2 & 3 & 1 \\ 2 & 1 & 0 & 1 & 2 \\ 1 & 0 & 1 & 2 & 0 \\ 3 & 1 & 2 & 0 & 1 \end{pmatrix} \qquad K = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$$

   Welche Größe hat das Ergebnis? Berechne alle Einträge.

2. (3 Punkte) Gegeben sind das Bild B und die Filter K_1 und K_2.

$$B = \begin{pmatrix} 9 & 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 & 9 \\ 9 & 9 & 9 & 9 & 9 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \qquad K_1 = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix} \qquad K_2 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$$

   (a) Berechne die Faltungen von B mit K_1 und mit K_2.

   (b) Was erkennen die beiden Filter? Erkläre die Ergebnisse.

3. (3 Punkte) Ein $n \times n$-Bild wird mit einem $f \times f$-Filter, Padding p und Stride s gefaltet. Das Ergebnis hat die Größe $\frac{n+2p-f}{s} + 1$ (abgerundet). Berechne die Größe für
   (a) $n = 28$, $f = 5$, $p = 2$, $s = 1$
   (b) $n = 32$, $f = 3$, $p = 0$, $s = 1$
   (c) $n = 39$, $f = 5$, $p = 0$, $s = 2$
   (d) $n = 7$, $f = 3$, $p = 1$, $s = 2$
   (e) $n = 224$, $f = 7$, $p = 3$, $s = 2$

   Welches Padding braucht man bei $f = 3$ und $s = 1$, damit das Bild gleich groß bleibt?

4. (2 Punkte) Berechne das Ergebnis von Max-Pooling mit Größe 2 und Stride 2 für

$$\begin{pmatrix} 1 & 4 & 2 & 0 \\ 3 & 2 & 1 & 5 \\ 0 & 1 & 7 & 2 \\ 6 & 2 & 3 & 1 \end{pmatrix}$$

   Wie viele Parameter hat eine Pooling-Schicht?

5. (3 Punkte) Eine Faltungsschicht `nn.Conv2d(3, 10, kernel_size=3)` macht aus einem Farbbild mit 32×32 Pixeln 10 Feature Maps mit 30×30 Pixeln.
   (a) Wie viele Parameter hat die Schicht?
   (b) Eine vollständig verbundene Schicht `nn.Linear` soll aus denselben Inputs dieselbe Zahl an Outputs berechnen. Wie viele Parameter hätte sie?
   (c) Erkläre den Unterschied mit den Begriffen Parameter Sharing und lokale Verbindungen.

6. (4 Punkte) Ein Batch mit der Gestalt `(20, 3, 39, 39)` geht durch drei Faltungsschichten ohne Padding:

```
conv1 = nn.Conv2d(3, 10, kernel_size=3)
conv2 = nn.Conv2d(10, 20, kernel_size=5, stride=2)
conv3 = nn.Conv2d(20, 40, kernel_size=5, stride=2)
```

   (a) Welche Gestalt hat der Tensor nach jeder Schicht?

   (b) Wie viele Parameter hat jede Schicht?

7. (4 Punkte) Gegeben ist das folgende CNN für Fashion-MNIST.

```
class Netz(nn.Module):
    def __init__(self):
        super().__init__()
        self.conv1 = nn.Conv2d(1, 8, kernel_size=3, padding=1)
        self.conv2 = nn.Conv2d(8, 16, kernel_size=3, padding=1)
        self.fc = nn.Linear(16 * 7 * 7, 10)

    def forward(self, t):
        t = F.max_pool2d(torch.relu(self.conv1(t)), 2)
        t = F.max_pool2d(torch.relu(self.conv2(t)), 2)
        return self.fc(t.reshape(-1, 16 * 7 * 7))
```

   (a) Welche Gestalt hat ein Batch von 50 Fashion-MNIST-Bildern `(50, 1, 28, 28)` nach jeder Zeile von `forward`?

   (b) Wie viele Parameter hat das Netz?

8. (4 Punkte) (a) Warum eignen sich vollständig verbundene Schichten schlecht für große Bilder?
   (b) Was lernen die Filter in den ersten Schichten eines CNN typischerweise, was in späteren?
   (c) Welches Problem lösen die Skip Connections von ResNet?
   (d) Was ist Transfer Learning und wann ist es besonders nützlich?

9. (4 Punkte) Ändere in Colab das CNN aus dem Notebook so, dass es zwei Faltungsschichten mit 3×3-Filtern und Padding 1 hat (16 und 32 Filter), jeweils mit ReLU und Max-Pooling, und danach eine vollständig verbundene Schicht. Gib die Zahl der Parameter aus, prüfe die Ausgabegestalt mit `torch.randn(5, 1, 28, 28)` und trainiere es drei Epochen wie im Notebook. Vergleiche mit dem CNN aus dem Notebook.