Question 1. Give an elevator pitch (no more than 100 words) for your final project idea.

Our smart parking detection system uses computer vision and deep learning to revolutionize parking management. By leveraging a Convolutional Neural Network (CNN), our solution automatically monitors parking spaces through video feeds, instantly identifying whether spaces are occupied or vacant. The system processes video frames in real-time, analyzing each designated parking spot to detect the presence of vehicles. This intelligent approach eliminates the need for expensive hardware installations like individual sensors, making it a cost-effective solution for parking lots of any size. The system provides real-time occupancy data, helping both lot operators and drivers save time and resources.

Question 2. Give the following for your data set,

1) **Collector(s):** Asad Ali Panwhar.
2) **Year:** 2024
3) **Title of Dataset:** Parking_Lot_Detection_Counter
4) **Publisher:** Kaggle
5) **DOI or URL:** https://www.kaggle.com/datasets/iasadpanwhar/parking-lot-detection-counter/data
6) **Study/Paper/Reason:** This dataset is designed for the task of detecting and counting empty and occupied parking spots in a parking area.

Question 3. List the language and the libraries you'll use. E. g., Python 3.13.1, Numpy, Matplotlib etc.

Python, OpenCV, Tensorflow/Keras, Numpy, Pickle.

Question 4. Describe what code will you be writing yourself?

- The code to train a CNN model that can detect if a parking space has a car or not
- A tool to mark and save parking space positions in video frames
- The main detection logic that:
  - Processes video frames
  - Checks each marked space for cars
  - Counts free/occupied spaces
  - Shows results on the video

Question 5. What do you think is the best choice of model(s) for your project? Why?

One of the models I chose is VGG16. It is basically an application of transfer learning.

1. **Advantages of VGG16**:

- Proven architecture with strong feature extraction capabilities
- Simple and straightforward architecture (just stacked convolution layers)
- Transfer learning benefits from ImageNet pre-training
- Good at learning spatial hierarchies of features
- Robust to various parking space angles and lighting conditions

2. **Trade-offs to Consider**:

- Larger model size compared to MobileNet
- More computationally intensive

- Slower inference time
- Higher memory requirements

Since parking space detection doesn't require real-time processing at extremely high speeds (cars park relatively slowly), VGG16's slightly slower inference time is an acceptable trade-off for its robust feature extraction and high accuracy.

Question 6. What are the hyper-parameters of this model and how will you pick the optimal ones?

We could optimize it by:

- Freezing early layers to use pre-trained weights
- Fine-tuning only the later layers for our specific task
- Using a smaller input size since parking spaces are simple scenes
- Potentially removing some layers if full depth isn't needed

Question 7. How will you evaluate the performance of the model?

Comparing the epochs with model loss and accuracy with test data. Also predicting the number of empty spaces in a parking space video.

Question 8. **https://github.com/Bhanupritam/vempali-b00116753-spring-2025**