I present the tasks to be performed, for credit for the subject. I divided them into two programming languages. If necessary, I can add some more.

To pass the course, 50% of the simple tasks and one of the open tasks are required. The higher grade is, of course, all open and as many simple ones as possible. It is possible to propose individually your own idea. All of this will also be in MS Teams, of course.

There will also be some lab assignments. If anyone completes less than 50%, they will have their final grade reduced by one degree. If less than 30% - he will not be able to get more than a 3. If there is no lab work at all....

Python simple tasks:

1. Build and train a simple Perceptron for the classification of a dataset.

2. Use a neural network to predict the price of a used car based on its properties.

3. Build a neural network for binary classification of spam and non-spam emails.

4. Use neural networks to classify images of hand-written digits.

5. Train a neural network to predict the survival rate of passengers on the Titanic based on their features.

6. Use a neural network to classify the sentiment of movie reviews.

7. Build a neural network for the prediction of housing prices based on its location, size, neighborhood, etc.

8. Use a neural network for predicting the type of a flower based on its features (such as petal length, petal width, etc.).

9. Train a neural network to detect human faces in images.

10. Use a neural network to classify TV series episodes as comedy, drama or thriller.

11. Build and train a Convolutional Neural Network (CNN) for the classification of images in a medical imaging dataset.

12. Use a neural network for the prediction of stock prices based on historical data and market indicators.

13. Train a Recurrent Neural Network (RNN) for the generation of text in a given style or tone.

14. Build and train a feedforward neural network for the prediction of the amount of rain in a particular region in the coming days.

15. Use a neural network for recommendation system based on users viewing behavior in a video-related app/website.

16. Train a Generative Adversarial Network (GAN) for the generation of realistic images.

17. Build and train a Convolutional Neural Network (CNN) for object detection and localization in images.

18. Use a neural network for the prediction of a baseball team's outcome based on their player statistics.

19. Train an Autoencoder neural network for image de-noising, reconstruction and compression.

20. Use a neural network for the identification of a disease based on patient symptoms and test results.

And open tasks:

1. Implement a neural network-based model to predict the stock prices of cryptocurrency markets. The model should be capable of doing real-time predictions and should use historical data from multiple cryptocurrency exchanges.

2. Build a web application that uses a convolutional neural network to identify plant leaf diseases from images uploaded by users. The application should provide information on the type of disease and recommend methods of treatment.

3. Use deep learning techniques to build a language translation application. The application should take text from one language and output it in another language, using a trained neural network model.

C++ simple tasks:

1. Implement a simple Perceptron for the classification of a dataset.

2. Use a neural network to predict the price of a used car based on its properties.

3. Build a neural network for binary classification of spam and non-spam emails.

4. Use neural networks to classify images of hand-written digits.

5. Train a neural network to predict the survival rate of passengers on the Titanic based on their features.

6. Use a neural network to classify the sentiment of movie reviews.

7. Build a neural network for the prediction of housing prices based on its location, size, neighborhood, etc.

8. Use a neural network for predicting the type of a flower based on its features (such as petal length, petal width, etc.).

9. Train a neural network to detect human faces in images.

10. Use a neural network to classify TV series episodes as comedy, drama or thriller.

11. Implement Convolutional Neural Networks (CNNs) for the classification of images in a medical imaging dataset.

12. Use a neural network for the prediction of stock prices based on historical data and market indicators.

13. Train a Recurrent Neural Network (RNN) for the generation of text in a given style or tone.

14. Implement Feedforward neural network for the prediction of the amount of rain in a particular region in the coming days.

15. Use a neural network for recommendation system based on users viewing behavior in a video-related app/website.

16. Train a Generative Adversarial Network (GAN) for the generation of realistic images.

17. Implement Convolutional Neural Networks (CNNs) for object detection and localization in images.

18. Use a neural network for the prediction of a baseball team's outcome based on their player statistics.

19. Implement an autoencoder neural network for image de-noising, reconstruction, and compression.

20. Use a neural network for the identification of a disease based on patient symptoms and test results.

And open tasks:

1. Build a platform for analyzing electroencephalogram (EEG) data using neural networks. The platform should be capable of processing large datasets of EEG signals, computing frequency and time-domain features, and using a deep learning model to predict diagnoses such as epilepsy or sleep disorders.

2. Implement a Convolutional Neural Network (CNN) to predict the likelihood of a crime occurring in a particular geographic area. The neural network should use data from a geographical information system (GIS) and be capable of making real-time predictions.

3. Build a weather forecasting application that uses deep learning techniques. The application should be capable of predicting local weather patterns by analyzing satellite images, weather reports, and other meteorological data.