You must complete this programming exam yourself using a computer with Internet access and computer with Internet connection. You are allowed to search and use all available material. You are not allowed to discuss in any form (chat, email, etc.) during the exam (the course instructor is the only exception - use the course Slack). All code and text must be written by yourself. Return the requested files and ***only*** the requested files and ***nothing else*** (make sure you name them correctly and use the correct format)!

1. Basics (5pts) **[DL 2pm15]**
   In your Python prompt type:

   ```
   >>> import socket
   >>> hostname = socket.gethostname()
   >>> ipaddr = socket.gethostbyname(hostname)
   >>> address = '4 Pivet Drive, Little, UK'
   >>> print(f"Address: {address} - Hostname: {hostname} - ipaddr: {ipaddr}")
   ```

   Replace the "address" by the street address of where you do your exam. Run the code and take a high quality screenshot (high resolution so that everything is clearly visible) of your whole desktop that includes the code and its results. Store as a PNG file and return before the deadline.

   Return:

   - `<FIRST>_<SURNAME>__basics.png`

2. CIFAR-10 1-NN (10p) **[DL 2pm45]**
   The starting point is your solution to Question 4 in Exercise 2, i.e. 1-NN classifier for CIFAR-10 data. Re-run your code and output the test accuracy using the following Python code:

   ```
   >>> import socket
   >>> hostname = socket.gethostname()
   >>> ipaddr = socket.gethostbyname(hostname)
   >>> accuracy = <FROM YOUR CODE>
   >>> print(f"1-NN Accuracy: {accuracy} - Hostname: {hostname} - ipaddr: {ipaddr}")
   ```

   Take a screenshot where full desktop is visible including the code and results windows.

   Return:

   - `<FIRST>_<SURNAME>__1nn.py`
   - `<FIRST>_<SURNAME>_1nn_screen.png`

3. CIFAR-10 k-NN (15p) **[DL 4pm00]**
   This question you must solve by editing the above 1-NN solution (the submitted Python code). In the new version the code must calculate distance to k best matches and then classify the test sample to the most frequent class (mode) in the k best.

   You are not allowed to use any external Python libraries for k-NN, but the starting point is your 1-NN code.

   Test your code for $k = 1, 2, 3, 4, 5$ and make a plot where CIFAR-10 test set accuracy is plotted for all these numbers (x-axis is k and y-axis is the corresponding classification accuracy).

   Return:

   - `<FIRST>_<SURNAME>__knn.py`
   - `<FIRST>_<SURNAME>_knn_graph.png`