**EM624 Midterm Exam - Fall 2020**

**Please submit your results for Section 1 in a doc/pdf file, for Section 2 and 3 in one single .py file**

**Section 1: General questions**

1. Why is important to define a testing strategy when coding? How would you perform testing for your code?
2. In Python, what are the differences between dictionaries and pandas structures? What they have in common?
3. Describe 3-4 most relevant things you want to do to your code to make it more readable
4. Why project management is important in software development projects and how can it be different from project management for other domains?
5. Why/when you may want to use functions in Python?

**Section 2: Code checking**

The following scripts are students’ solutions to exercises proposed a previous semester.

You will find a brief descriptions and the student’s solution.

For each of them:

* 1. Check if it’s doing what it was supposed to do
  2. Describe what is wrong, if any
  3. Fix it

If an input is required from the user, be sure the input testing is performed in the proper way.

1. Write a script that takes a string as input from the user and prints a string where for every character in the original, there are three characters (example: 'The' → 'TTThhheee').

**N = input("Enter your characters: ")**

**L = []**

**for letters in N:**

**letters.split()**

**L.append(letters)**

**print (L\*3)**

1. Write a script that calculates the 3 longest words of a text stored in a file and print them from the longest to the smaller of the 3. Please note:

* The name of the file is *word\_list.csv* and it doesn’t need to be asked to the user (meaning the name will be in the code)
* Assume that the file contains *n* records, each one composed by 1 word. Words can be present more than once, but only unique words need to be considered
* A sample word\_list.csv file is attached for testing.

**handle = open('word\_list.csv','r')**

**top3 = ["","",""]**

**for line in handle:**

**#For each line in the file, strip the input and put it into the word variable**

**word = line.strip()**

**#Compare the length of each incoming word to the length of each word in each position**

**for i in range(0,3):**

**top3.sort(key = len)**

**if (len(word) > len(top3[i])):**

**top3[i] = word**

**#Print the words**

**print ("\nThe 3 longest words are:"), top3**

1. Write a script that takes a character (i.e. a string of length 1) as input from the user and returns False if it is a consonant, True otherwise. A check on the length of the input string and its being alphabetical is required and if not, send a message to the user and ask again.

**while True:**

**#prompts and receives user input**

**char = input('Please enter an alphabetical character:')**

**if char.isdigit(): #checks if input is numerical**

**print ('Invalid input.')**

**else:**

**if len(char) > 1: #checks if input is more than one character**

**print ('Invalid input.')**

**else:**

**if char == 'a' or 'e' or 'i' or 'o' or 'u' or 'y': #checks if input is a vowel**

**print ('True')**

**else:**

**print ('False')**

**Section 3: Writing code**

1. Write a script that reads a file “ai\_trends.txt”, into a list of words, eliminates from the list of words the words in the file “stopwords\_en.txt” and then
   1. Calculates the average occurrence of the words. Occurrence is the number of times a word is appearing in the text
   2. Calculates the longest word
   3. Calculates the average word length. This is based on the unique words: each word counts as one
   4. Create a bar chart with the 10 most frequent words.
2. Write a script that reads a file “cars.csv”, into a pandas structure and then print
   1. the first 3 rows and the last 3 of the dataset
   2. the 3 cars with the lowest average-mileage
   3. the 3 cars with the highest average-mileage.