IT1007 Introduction to Programming with Python and C
Lab Exercise 03

Submission instructions:

1. There are two parts in this lab. For Part A, you have to submit within the same day of your lab session. For your Part B, you have **six days** to work on it. (E.g. if your TLab is on Monday, then your deadline is the coming Sunday midnight.) There are two separate skeleton files for each part of the lab. Please submit them separately. **Please do NOT merge the two parts into one single file**.
2. Submit your solution to the **CORRECT** group folder in the IVLE workbin.
3. Add your student number to your file in your submission, e.g. **`Lab_03_Part_A_A1234567X.py`**.
4. You must name your functions exactly as the questions state.
5. You should not use any package other than matplotlib.

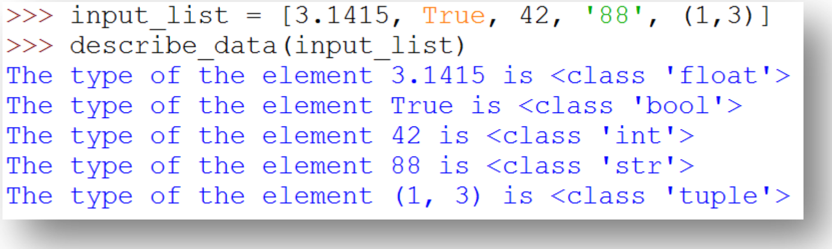
Failure to follow each of the instruction will result in 10% deduction of your marks.

Part A (Deadline: Same day of your TLab) (40%)

The functions to be submitted should work for both tuples and lists. And you should NOT change the value of the input. You do not have to go into a nested sequence.

Question 1 : Describe Data (20%)

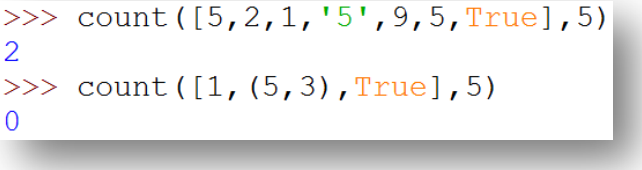
Write a function, **`describe_data`** , that takes in a sequence L and prints each element within it and its type.



```
>>> input_list = [3.1415, True, 42, '88', (1,3)]
>>> describe_data(input_list)
The type of the element 3.1415 is <class 'float'>
The type of the element True is <class 'bool'>
The type of the element 42 is <class 'int'>
The type of the element 88 is <class 'str'>
The type of the element (1, 3) is <class 'tuple'>
```

Question 2: Counting Data (20%)

Write a function, **`count`,** that takes in a sequence L and value x. The function should then search within the sequence and return how many times x appears in L. **NOTE: No credit will be given if the inbuilt function "count" or "len" is use.**

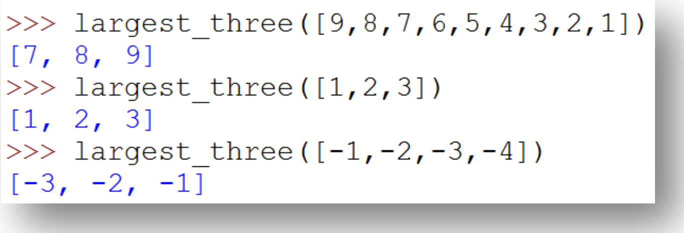


```
>>> count([5,2,1,'5',9,5,True],5)
2
>>> count([1,(5,3),True],5)
0
```

Question 3: Largest Three Number

Write a function, **largest_three,** which takes in a sequence L of numbers and returns the largest three numbers in the sequence. You can assume the input is a sequence of more than 2 numbers with no duplicates, but nothing else. Your output should be a list of three numbers from smallest to the largest.

NOTE: No credit will be given if the inbuilt function "sort" is used.

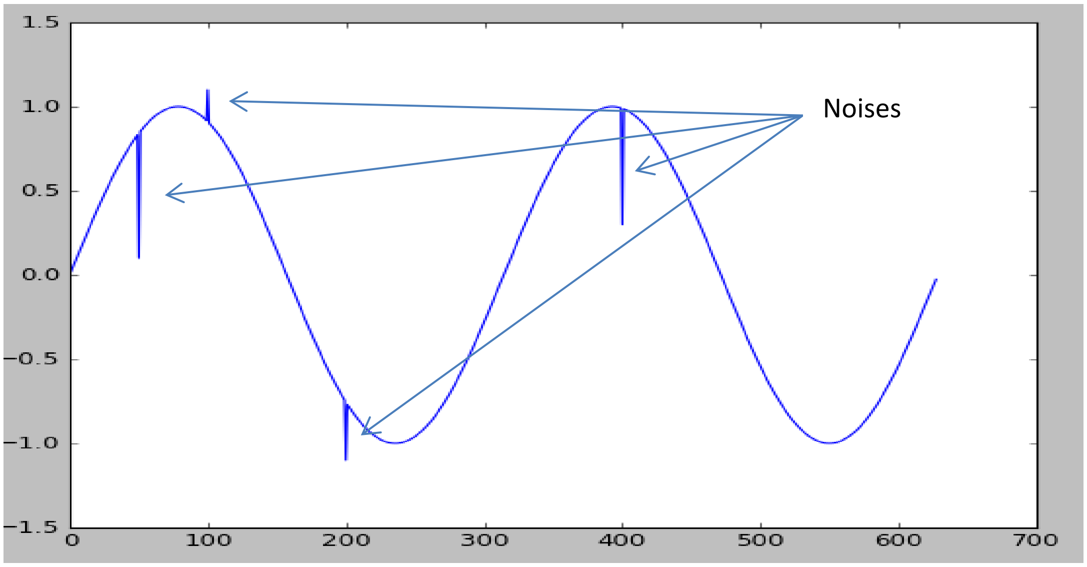


```
>>> largest_three([9,8,7,6,5,4,3,2,1])
[7, 8, 9]
>>> largest_three([1,2,3])
[1, 2, 3]
>>> largest_three([-1,-2,-3,-4])
[-3, -2, -1]
```

Part B (Deadline: Day of TLab + 6 days)

Noise Detection and Correction

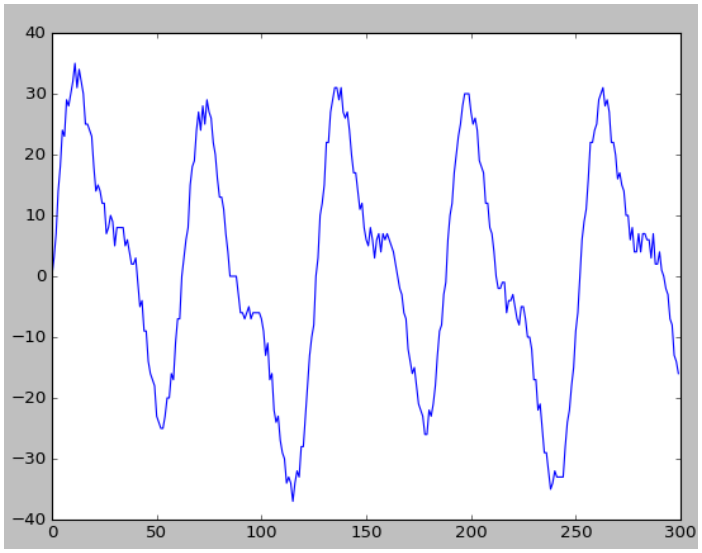
As an engineer, you will encounter signal processing problems. For example, if you record a wave sample, e.g. audio wave, you may receive your signal AND some unexpected jumps. Usually these unexpected jumps are NOT the original signal but are caused by some errors, e.g. faulty equipment or another external interference.



These unexpected jumps in the recording are called **noises**. Before we further process the signal, we have to detect and repair these noises.

Question 4: Noise Detection (30%)

You are given a wave sample in the form of a list in your skeleton file. When you run the file, you can see the wave while the data is stored in "`original_wave_sample`".

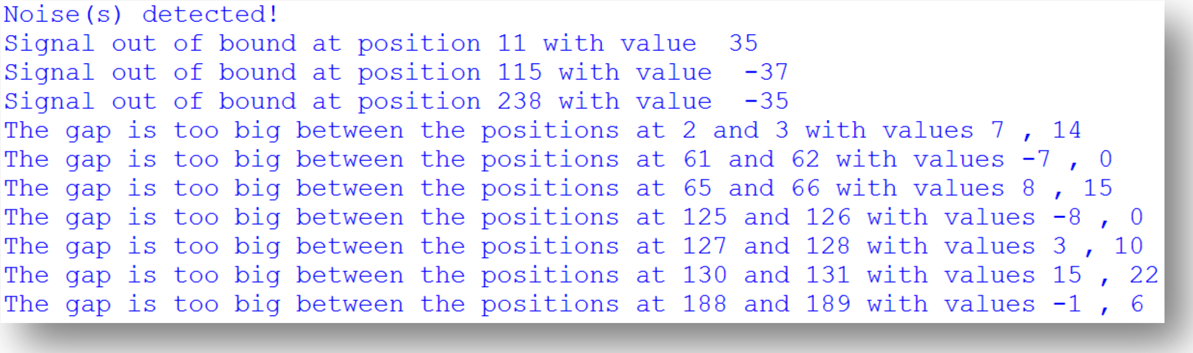


However, it is hard to detect where the noise in the graph is. We only know that

1. Every pair of consecutive data samples will have a gap less than or equal to 6 units
2. Overall, the sample amplitude x will be within the range -34 ≤ x ≤ 34.

Your Task

Write a function **`detect_noise`** to find out all the positions of the samples that violate the two criteria above. If there is no noise, you should print 'No `noise detected`'. Here is a sample output for you. Your answer should be exactly the same.



```
Noise(s) detected!
Signal out of bound at position 11 with value  35
Signal out of bound at position 115 with value  -37
Signal out of bound at position 238 with value  -35
The gap is too big between the positions at 2 and 3 with values 7 , 14
The gap is too big between the positions at 61 and 62 with values -7 , 0
The gap is too big between the positions at 65 and 66 with values 8 , 15
The gap is too big between the positions at 125 and 126 with values -8 , 0
The gap is too big between the positions at 127 and 128 with values 3 , 10
The gap is too big between the positions at 130 and 131 with values 15 , 22
The gap is too big between the positions at 188 and 189 with values -1 , 6
```

Note:

1. You can assume that each noise is "isolated". Namely, if the sample at position x has noise, the samples at positions x-1 and x+1 are NOT noises. Also, you can assume that the first and last samples in the list are not part of the noises.
2. You should print all "Signal out of bound" report before the "large gap".
3. You can assume if a sample is "out of bound", then it will not have the problem or "large gap", and vice versa.
4. You cannot use any ready-made signal processing packages for both questions 4 and 5.

Question 5: Noise Correction (30%)

If one sample value in the data is noise, it means that we have lost the data. However, we can try to "recover" the lost value by averaging the neighbors.

Your Task

Write a function **`repair_signal`** to fix the signal according to the rules below.

1. If a sample is out of bound, you can replace the value with the average of its two neighbors.
   - For example, if the signal with index 20 is a noise, replace signal[20] with (signal[19]+signal[21])/2
2. If two samples have a gap that is too big, you can replace the first value of the two samples with the average of its two neighbors.

Note:

1. You cannot change the values of the “original_wave_sample”. Your function should output another list that is the “corrected” signal.
2. You can assume that the data will not violate the two criteria in Question 4 after the averaging.