# QUIZ 5

**Due Date : 24.11.2023 - Friday (23:00)**
**Advisor :** R.A. Görkem AKYILDIZ
**Programming Language :** Python 3.6.8

## 1 Introduction

In this quiz, you are expected to gain practice on some basic sorting algorithms and their steps. So far you have used Python's own sorting algorithm without having any knowledge about what is under the hood. Normally, for real life scenarios it may seem beneficial and optimal to use built in sorting algorithm of Python but there are some benefits of learning how to implement and run your own sorting algorithm. First things first, when you learn the sorting algorithm's implementation, as you know what is under the hood you can be more careful about your decisions at the sorting step. Later on, you can also customize an existing sorting algorithm to solve your own sorting problem maybe even faster than the built in sorting algorithms. So, indeed, it is very beneficial to be curious and to learn what is under the hood.

## 2 Implementation

For this quiz, you will deal with two sorting algorithms, one of them is "Bubble Sort" while the other one is "Insertion Sort". You are requested to print out all of the steps of these sorting algorithms for given list to have a better understanding of what is going on under the hood.

## 3 Definition of Input

Initial status of the list will be given as a text file. Contents of the list will be integers and they will be separated by space to identify elements. Each input file will contain only and only one list. You must take the name of the input file as the first command line argument of your program. Note that an empty file means an empty list.

## 4 Definition of Output

Text representation of the list will remain the same as the input but you must printout all of the passes (steps) for the sorting. Note that you must cut out the solution if list is sorted before all passes have been done. Moreover, it is forbidden to use any kind of built in sorting or checking mechanisms for sort operations, you must check if list is sorted by your own, you must consider cut out cases and code your code so. If the given list is already sorted, you must just printout "Already sorted!". Note that empty lists or lists with only one element counts as sorted. Each pass will be separated from each other by a newline character and the structure of each line will be as follows:

- Pass <PN>: <CUR_LST>

Definitions:

- <PN>: Number of current pass.
- <CUR_LST>: Current status of the list at the end of the given pass.

## 5 Restrictions

- Your code must be able to execute on our department's developer server (dev.cs.hacettepe.edu.tr).
- You must obey given submit hierarchy and get score (1 point) from the submit system.
- **You must benefit from lists, loops and functions.**
- Your code must be clean, do not forget that main method is just a driver method that means it is just for making your code fragments run, not for using them as a main container, create functions in necessary situations but use them as required.
- You must use comments for this project and you must give brief information about the challenging parts of your code. Do not over comment as it is against clean code approach. Design your comments so that they make your code fully understandable and not excessive for others.
- You can benefit from Internet sources for inspiration but do not use any code that does not belong to you.
- You can discuss high-level (design) problems with your friends but do not share any code or implementation with anybody.
- Do not miss the submission deadline.

- Source code readability is a great of importance. Thus, write READABLE SOURCE CODE, comments, and clear MAIN function. This expectation will be graded as "clean code".
- Use UNDERSTANDABLE names to your variables, classes, and functions regardless of the length. The names of classes, attributes and methods should obey Python naming convention. This expectation will be graded as "coding standards".
- You can ask your questions through course's Piazza group, and you are supposed to be aware of everything discussed in the Piazza group. General discussion of the problem is allowed, but **DO NOT SHARE** answers, algorithms, source codes and reports.
- All quizzes must be original, individual work. Duplicate or very similar quizzes are both going to be considered as cheating.

## 6 Execution and Test

Your code must be executed under **Python 3.6.8** at **dev.cs.hacettepe.edu.tr**. If your code does not run at department's developer server during the testing stage, then you will be graded as 0 for code part even if it works on your own machine.
Sample run command is as follows:

- python3 sort.py input.txt output_bubble.txt output_insertion.txt

## 7 Grading

| Task | Point |
|---|---|
| **Bubble Sort** | 50 |
| **Insertion Sort** | 50 |
| **Total** | 100 |

**Note that you must score one at the submit system, otherwise 20% of your grade will be deducted, moreover, you must implement a main function otherwise 10% of your grade will be deducted! There may also be other point deductions if you do not obey the given rules, such as if you do not use lists, functions and/or loops as necessary.**

## 8 Submit Format

File hierarchy must be zipped before submitted (Not .rar, only not compressed .zip files because the system just supports .zip files).

```
- b<StudentID>.zip
    - sort.py
```