## Algorithms and data structures

Labwork 5: Searching and Sorting Algorithms

Dr. Doan Nhat Quang: doan-nhat.quang@usth.edu.vn

## 1. Searching and Sorting

After each labwork session:

- You will have one week (or 7 days) to complete the remaining exercises and upload your files to the "Labwork X Group Y" assignment in Google Classroom. (e.g. Labwork 1 Group 1) if you are from Group Y and want to submit labwork X
- Compress all code source files in a zip file and rename it as FULLNAME-ID-Lab#no.zip (e.g NguyenVanA-BI10-070-Lab1.zip). Save your files according to the exercise number i.e Ex1.cpp, Ex2.c, etc. Incorrect filenames will result in no score for the respective exercises.
- Only code source files (.c or .cpp) should be in the zip files. Other files (.exe, .o) MUST be removed from the zip file.
- - Copy/Paste from any source is not tolerated. Penalty will be applied for late submissions.

NOTE: You must follow the guide. Incorrect zip file names, zip files containing other files (.exe), and copy/paste lead to heavy penalties.



Exercise 1: Suppose that a given array is sorted and a new value v is added

into this array. We try to adjust the Binary Search to find the correct position for v in the array according to the ascending order.

## Exercise 2:

In this problem, we would like to implement a variation of the Insertion Sort algorithm The idea is to use the above Binary Search. Implement and test this algorithm in C/C++. Analyze and compute the complexity of this algorithm in the best, average, and worst scenarios.

## Exercise 3:

Re-implement Exercise 2 using a linear data structure: List, Stack, Queue. Justify your choice of data structure which one is the most suitable for sorting algorithms.