# VE477
# Introduction to Algorithms

*Lab 5*

Manuel — UM-JI (Fall 2021)

**Goals of the lab**

- Course application
- Data sctructures
- Python Object Oriented Programming

*Unless specified otherwise, all the programs are expected to be completed in Python or O'caml.*

## 1 Programming

1. Graph representations:
   (a) Implement a class for sparse graphs;
   (b) Implement a class for dense graphs;

   In each case implement at least the following methods:

   - `AddEdge`
   - `RemoveEdge`
   - `AddVertex`
   - `RemoveVertex`
   - `IsAdjacent`[1]
   - `GetEdgeWeight`
   - `SetEdgeWeight`
   - `GetVertexValue`
   - `SetVertexValue`

2. Implement Dijkstra algorithm (3.13|3.149) using Fibonacci heaps;
3. Bellman-Ford (algorithm (3.17|3.153));
4. Compare the efficiency of Bellman-Ford and Dijkstra in terms of (i) complexity and (ii) running time;

## 2 Interview Problems

- To homogenize the course code system over all departments, SJTU decides to implement a new strategy: a course code is an integer, it must be unique, and the sum of all course codes should be minimum. Given an array of $n$ course codes (with no letter, e.g. 477 instead of ve477), increment duplicated elements such that all new course codes are unique and their sum is minimum.

  Example: on input $[5, 1, 2, 3, 2]$, return $[5, 1, 2, 3, 4]$. All ids are unique and the sum is minimum.

- Due to the high increase in shared bikes on campus SJTU has decided to remove most bicycle shelters. However after a while they realise they should protect some bikes. For each line of bike park, they decide to build a removable roof that can cover $k$ out of $n$ bikes. We know that two bikes can be parked within one meter, and that not all slots are occupied. Given $n$, the location of the bikes, and $k$, determine the minimum length of the roof to cover $k$ bikes.

  Example: on input $n = 6$, $positions = [0, 1, 2.5, 10.5, 11, 12]$, and $k = 3$, return 1.5, corresponding to a roof covering bikes at locations 10.5 to 12.

[1] `v.IsAdjacent(u)` checks whether or not vertices $v$ and $u$ are adjacent.