

# Mini-project Topics – HEDSPI

## Object-Oriented Language and Theory

Nguyen Thi Thu Trang, trangntt@soict.hust.edu.vn

### Guidelines:

- All the mini-projects must be designed and implemented in Java and by students themselves. If the teacher find out that students don't write the source code (even a part of it), the score will be 0.
- Milestones:
  - o 8th week: Choosing mini-project topic, starting to study the mini-project
  - o 9th week: *Midterm day off*
    - Monitor should submit the list of groups and corresponding mini-project topic before Friday, 4th of April 2019
    - Working in group for the mini-project
  - o 10th-11th week: Normal labs
  - o 12th-13th week: Doing and discussion about mini-projects at lab
  - o **Deadline for final submission: 8th of May**
  - o 14th-15th-16th week: Presenting your results at lab: 15 minutes for each group

Please choose one topic below, but you should keep the balance for the number of groups for each topic. Visual demonstration here means using visual objects/shapes to present laws/algorithm (this is one way to present: <https://www.youtube.com/watch?v=es2T6KY45cA>)

1. An application to (visually) demonstrate Newtons laws of motion
2. An application to (visually) demonstrate following data structures and algorithms:
  - a. Stack
  - b. LinkedList
  - c. BalancedTree
3. An application to (visually) demonstrate following data structures and algorithms:
  - a. Queue
  - b. ArrayList
  - c. HashTable
4. An application to (visually) demonstrate following algorithms:
  - a. Insertion sort
  - b. Shell sort
  - c. Tower of Hanoi Problem
5. An application to (visually) demonstrate following sorting algorithms
  - a. Bubble sort
  - b. Quick sort
  - c. Heap sort
  - d. Radix sort
6. An application to (visually) demonstrate sorting algorithms
  - a. Bubble sort

- b. Selection sort
  - c. Merge sort
  - d. Bucket sort
- 7. An application to (visually) demonstrate graph algorithms
  - a. Breadth First Search (BFS)
  - b. Depth First Search (DFS)
  - c. Shortest Path from source to all vertices **\*\*Dijkstra\*\***
- 8. An application to (visually) demonstrate searching algorithms
  - a. Sequential searching
  - b. Binary searching
  - c. A\* algorithm