

Notice

- (NEW) submission procedure (Any inquiry about this should be addressed to Mr. WU Hanqing):
 - 1) Create a folder and name as <student_no>_<yourname> E.g., 12345678d_CHANTaiMan
 - 2) Name the .java file as A3.Q1_<student_no>_<yourname>_<class_name>.java
E.g., A3.Q1_12345678d_CHANTaiMan_Sorting.java (A for assignment, P for project)
But please don't change the class name in the Java file.
 - 2) Put all your .java files into this folder, including java files and screenshots. Please DO NOT submit the entire project.
 - 3) Compress this folder (all compressed format are acceptable, e.g., .jar, .rar, .zip, .7z) and submit the compressed file to the Blackboard.
 - 4) Any wrong file naming and submission will result in mark deduction.
 - All other requirements of assignment#2 also apply for this one.
1. (20 points) Given a sorted array, write an algorithm to transform it to a *balanced* binary search tree—for any node, the size of its left subtree and the size of its right subtree differ by at most 1. Try to make your algorithm as efficient as possible.
 2. (30 points) The square graph G^2 of a graph G has the same set of vertices, but two vertices are adjacent in G^2 when their distance in G is at most 2.¹
Write algorithms to find the square graph of a given graph, for both adjacency list and adjacency matrix representations. Try to make your algorithms as efficient as possible.
 3. (20 points (bonus)) A connected graph has a cycle that visits every edge exactly once if and only if every vertex has an even degree ([wikipedia](#)). Write algorithms to find such a cycle if it exists, for both adjacency list and adjacency matrix representations. Try to make your algorithms as efficient as possible.

¹This is very similar as the approach of finding triangles mentioned by ZHUANG Yufan.