

# Exercises - Home project 1

Include all your reasoning steps, but only the necessary ones. **Do not use built-in python functions that already solve the problems for you.**

Implement an adjusted version of the Havel-Hakimi algorithm in the provided Python notebook. This version should generate all possible non-isomorphic graphs corresponding to a given sequence of integers, or return an empty list if the sequence is not graphical.

**Input:** A sequence of integers representing a degree sequence.

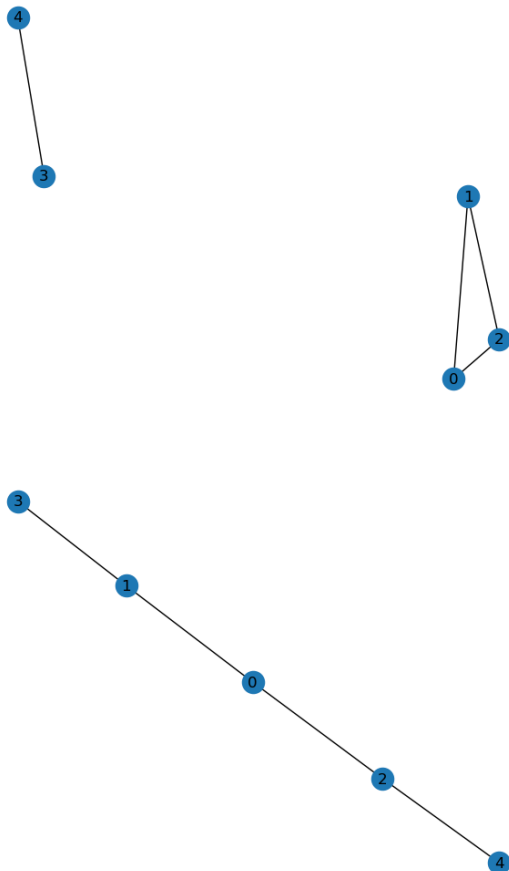
**Output:** A list of all possible graphs corresponding to the graphical sequence or an empty list if no valid graph exists.

- Tests run on these input sequences :
  - [2,2,2,1,1]
  - [3,3,3,3,3,3]
  - [3,3,3]

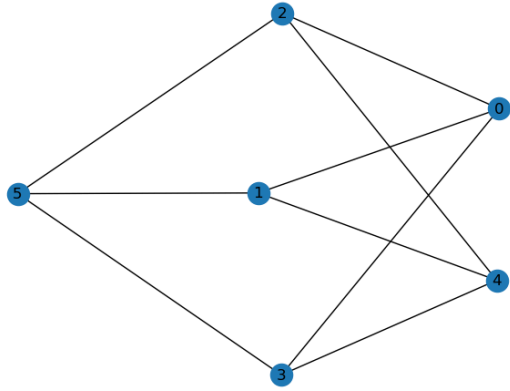
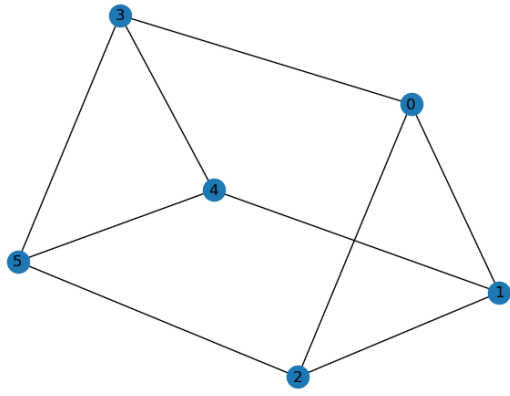
Make sure to include informal analysis on the worst time complexity of your algorithm in the python file. **(2 pts)**

## Tests:

Sequence [2,2,2,1,1] will output 2 graphs



Sequence [3,3,3,3,3,3] will output 2 graphs



Sequence [3,3,3] will output 0 graphs.