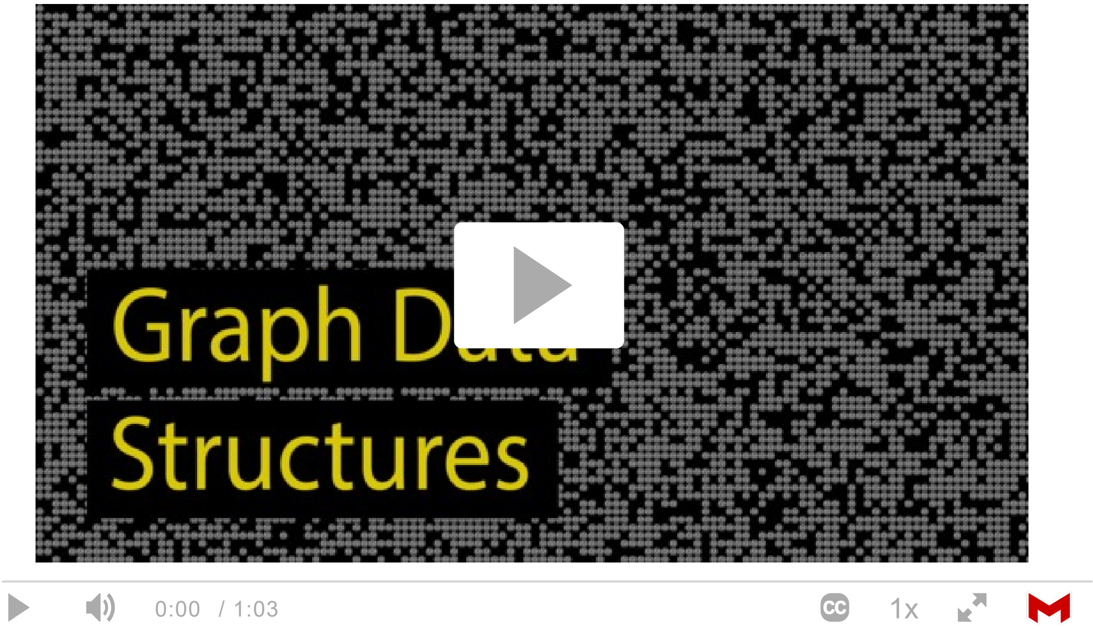
Week 7 - Overview



Graphs are a more general structure than the trees we studied last week; in fact, you can think of a tree as a special kind of graph. Graphs can be used to represent many interesting things about our world, including systems of roads, airline flights from city to city, how the Internet is connected, or even the sequence of classes you must take to complete a major in computer science. We will see in this module that once we have a good representation of a problem, we can use some standard graph algorithms to solve what otherwise might seem to be a very difficult problem.

While it is relatively easy for humans to look at a road map and understand the relationships between different places, a computer has no such knowledge. However, we can also think of a roadmap as a graph. When we do so we can have our computer do interesting things for us. If you have ever used one of the Internet map sites, you know that a computer can find the shortest, quickest, or easiest path from one place to another.

Objectives

During this week, you will:

- Differentiate the use of graph and map data structures from the application of tree data structures.
- Write a program which solves a problem using a complex data structure.

Readings

You will be responsible for reading the following chapters this week:

- *Data Structures and Algorithms in Python* by Michael Goodrich
  - 14. Graph Algorithms

Additional pages are available in this module reviewing:

- **An Introduction to Graphs (https://maryville.instructure.com/courses/43640/pages/an-introduction-to-graphs)**
- **The Graph Abstract Data Type (https://maryville.instructure.com/courses/43640/pages/the-graph-abstract-data-type)** *(including **Adjacency Matrix (https://maryville.instructure.com/courses/43640/pages/adjacency-matrix)** and **Adjacency List (https://maryville.instructure.com/courses/43640/pages/adjacency-list)**)*
- **The Word Ladder Problem (https://maryville.instructure.com/courses/43640/pages/the-word-ladder-problem)** *(including **Building a Word Ladder (https://maryville.instructure.com/courses/43640/pages/building-a-word-ladder)** and **Implement Breadth First Search (https://maryville.instructure.com/courses/43640/pages/implement-breadth-first-search)**)*
- **Do Together - Implementing Graph (https://maryville.instructure.com/courses/43640/pages/do-together-implementing-graph)**

Resources

This week highlights resources available through Runestone Interactive. This open education resource provides open source textbooks for computer science and programming content. The materials are licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International Licence. You may access the source material using the link below.

- Miller, B. & Ranum, D. (2011). *Problem solving with algorithms and data structures using python* (2nd ed.). Retrieved from **http://interactivepython.org/runestone/static/pythonds/index.html (http://interactivepython.org/runestone/static/pythonds/index.html)**