# Package Manager

## Objectives:

1. Parse an array of packages
2. Determine package hierarchy/dependencies
3. Throw out circular references
4. Determine install order

## Notes:

Directed graphs in C#: ( apparently this doesn’t exist in netcore?)

<https://msdn.microsoft.com/en-us/library/microsoft.visualstudio.graphmodel.graph.aspx>

topological sort:

* How to detect cycles??
  + Tarjan’s Strongly Connected Component Algorithm
  + <https://en.wikipedia.org/wiki/Tarjan%27s_strongly_connected_components_algorithm>
  + <https://www.youtube.com/watch?v=QnWDU1wcsPA>

<https://www.youtube.com/watch?v=eL-KzMXSXXI>

dijkstra’s algorithm:  
<https://www.youtube.com/watch?v=gdmfOwyQlcI>

topsort algorighm:

1. Pick an unvisited node (any)
2. DFS from node, exploring only unvisited nodes
3. On dfs recursion, add current node to ordering in reverse order (stack?)
4. Continue until all nodes visited, then go to 1