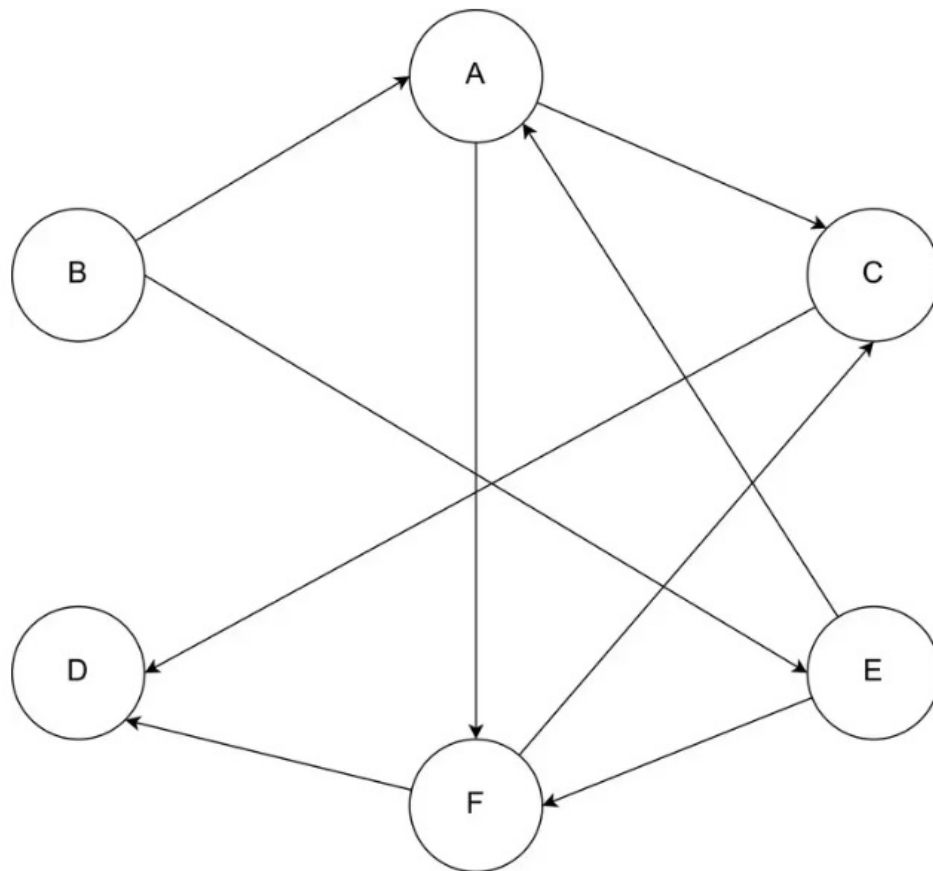


Описание

Топологическая сортировка может применяться для ориентированного ациклического графа (DAG). Ее суть состоит в упорядочивании вершин по следующему условию: Узел или вершина *a* должны посещать перед узлом или вершиной *b* для каждого направленного ребра *ab*. Соответственно если граф не DAG то применить сортировку невозможно.

Алгоритмическая сложность этого алгоритма составляет $O(V + E)$, где *V* это количество вершин, а *E* это количество ребер.

Например для следующего графа вывод топологической сортировки будет следующий:



Результат: [B, E, A, F, C, D]

Реализация на python

Реализация алгоритма на языке python представлена в файле под именем `python_lab_1`. В результате для полного графа DAG размером 1500 топологическая сортировка заняла 0.055 секунд времени или 0.43 если не использовать профилировщик

Реализация на c++

Реализация кода на с++ представлена в файле lab_1. В среднем на выполнение команды было затрачено 0.023 секунды что в 2 раза меньше чем на python. Для теста также применялся полный граф DAG с 1500 вершин.