Графы

Методы хранения

**Граф -** это топологичекая модель, которая состоит из множества вершин и множества соединяющих их рёбер. При этом значение имеет только сам факт, какая вершина с какой соединена.

(**Топологические** **модели** – это **модели**, которые отражают взаимные связи между объектами, не зависящие от геометрических свойств и содержат **топологическую** информацию в явном виде.)

**Вершина** - точка в графе, отдельный объект, для топологической модели графа не имеет значения координата вершины, её расположение, цвет, вкус, размер; однако при решении некоторых задачах вершины могут раскрашиваться в разные цвета или сохранять числовые значения.

**Ребро** - неупорядоченная пара двух вершин, которые связаны друг с другом. Эти вершины называются концевыми точками или концами ребра. При этом важен сам факт наличия связи, каким именно образом осуществляется эта связь и по какой дороге - не имеет значения; однако рёбра может быть присвоен “вес”, что позволит говорить о “нагруженном графе” и решать задачи оптимизации.

**Смежность** - две вершины смежные, если они соединены ребром, два ребра смежные - если они соединены вершиной.

Основные структуры данных для хранения графов:

* **Матрица смежности.** Матрица смежности представляет из себя матрицу, где заголовки строк и столбцов соответствуют номерам вершин графа, а само значение каждого ее элемента a(i,j) определяется наличием или отсутствием ребер между вершинами i и j (ясно-понятно, что для неориентированного графа такая матрица будет симметрична, ну или же можно договориться, что все значения мы храним лишь выше главной диагонали). Для невзвешенных графов a(i,j) можно задать количеством ребер из i в j (если нет такого ребра, то a(i,j)= 0), а для взвешенных также – весом (суммарным весом) упомянутых ребер.
* **Список смежности.** Ну тут, вроде бы, все просто. Каждой вершине графа можно, в общем случае, поставить в соответствие любую перечислительную структуру (список, вектор, массив, …), в которой будут храниться номера всех вершин, смежной данной. Для ориентированных графов будем заносить в такой список лишь те вершины, в которые есть «направленное» ребро из вершины-признака. Для взвешенных графов реализация будет более сложной.
* **Список ребер.** Довольно популярная структура данных. Список ребер и представляет собой собственно список ребер графа, каждое из которых задается начальной вершиной, конечной вершиной (для неориентированных графов здесь порядок следования не важен, хотя для унификации можно использовать различные правила, например, указывать вершины в порядке возрастания) и весом (только для взвешенных графов).

(Нарисовать детям граф, показать структуры наглядно)

**Дерево** — это связный ациклический граф.[1] Связность означает наличие маршрута между любой парой вершин, ацикличность — отсутствие циклов. Отсюда, в частности, следует, что число рёбер в дереве на единицу меньше числа вершин, а между любыми парами вершин имеется один и только один путь.

Далее решаем задачки: Из списка ребер составить список смежности и матрицу смежности,  [HYPERLINK "https://informatics.msk.ru/mod/statements/view.php?id=359&chapterid=464"От матрицы смежности к списку ребер](https://informatics.msk.ru/mod/statements/view.php?id=359&chapterid=464) , [Проверка на неориентированность](https://informatics.msk.ru/mod/statements/view.php?id=359) ,