

Домашнее задание № 2 (10 баллов)

Последний срок для присылки решения - 5 декабря

Задача 1 (6 баллов)

Неориентированный связный граф задан списком ребер (исходные данные представляют собой последовательность пар номеров соседних вершин). Предполагается, что исходные данные заданы корректно, проверять, что в результате чтения данных действительно получится связный граф, не надо.

Эксцентриситетом вершины называется максимальное из расстояний от этой вершины до всех остальных вершин графа. Расстояния определяются просто количеством соединяющих вершины ребер, то есть граф - не нагруженный.

Диаметром графа называется максимальный из эксцентриситетов его вершин.

Написать программу, которая по заданному графу и номеру одной из его вершин вычисляет диаметр графа и эксцентриситет заданной вершины.

Замечание: если для каждой вершины графа вычислить ее эксцентриситет, а затем выбрать из полученных значений наибольшее, то в результате получится величина диаметра графа. Однако, этот алгоритм не очень эффективный. Поищите более эффективный алгоритм для нахождения диаметра графа. Также один из секретов успеха - правильный выбор представления графа.

Еще одно замечание. Похоже, что для произвольного графа алгоритма, который работает асимптотически быстрее, чем $O(n^3)$, не существует. Максимум из эксцентриситетов дает именно такую оценку, хотя мне кажется, что коэффициент все же не будет в этом случае оптимальным.

Задача 2 (4 балла)

Нагруженный неориентированный граф задан списком ребер. Величина нагрузки - целое положительное число. Исходными данными для данной задачи, таким образом, является список троек целых чисел, в каждой тройке два первых числа представляют номера концов дуги, и третье число представляет нагрузку на дугу. Предполагается, что граф связный, проверять это отдельно не надо.

Требуется написать функцию, которая выдает список ребер некоторого остовного дерева этого графа, имеющего минимальное значение "самого тяжелого" ребра, то есть ребра с максимальной нагрузкой среди всех ребер построенного дерева.

В комментариях к функции оцените сложность вашего алгоритма.