

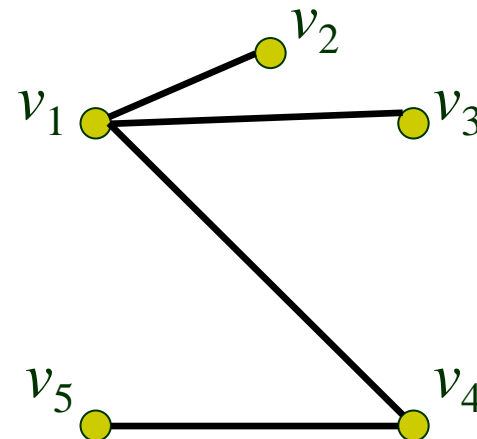


# Теория графов

## Деревья

# Деревья

- *Дерево* – это связный граф без циклов
- *Лес* – это граф без циклов
- Дерево, состоящее из 1 вершины, называется *тривиальным*

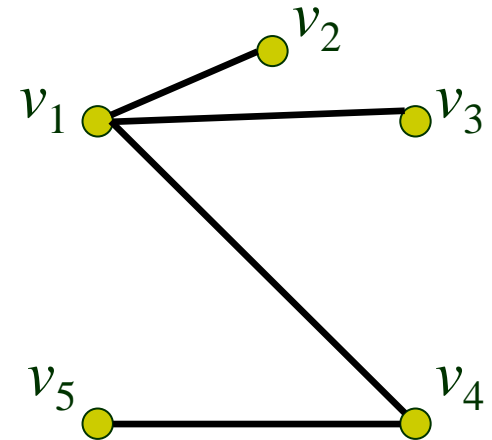


## Свойства

- Дерево является двудольным графом
- Если в дереве две любые несмежные вершины соединить ребром появится простой цикл

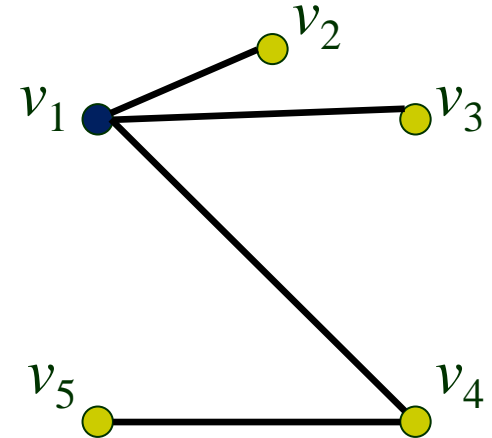
# Деревья. Свойства

- В дереве любое ребро является мостом
- В дереве  $m = n - 1$
- В любом нетривиальном дереве существует не менее 2 концевых вершин и не менее 1 концевого ребра
- Если в дереве больше 2 вершин, то в нем есть хотя бы 1 неконцевая вершина



# Корневое дерево

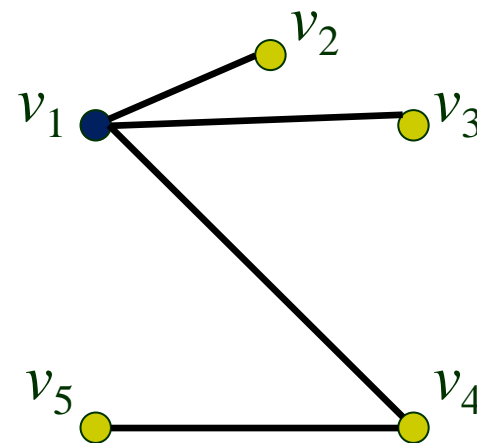
- Дерево с одной выделенной вершиной называется *корневым*
- Выделенная вершина называется *корнем*
- Корневое дерево можно задать *двоичным кодом (вектором)*
- Двоичный код – последовательность 0 и 1, которая строится на основе обхода дерева в глубину



# Корневое дерево. Двоичный код

## Алгоритм построения двоичного кода

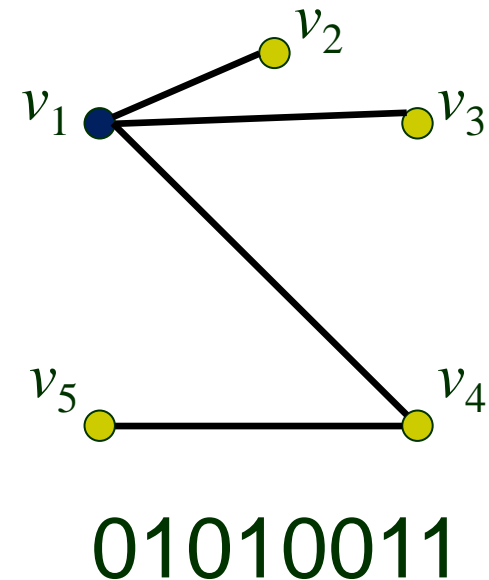
- Начиная с корня выполнить обход в глубину
- При прохождении ребра в первый раз (в направлении от корня) записать в код «0»
- При прохождении ребра во второй раз (в направлении обратно к корню) записать в код «1»



01010011

# Свойства двоичного кода

1. В двоичном коде количество «0» равно количеству «1» и равно количеству ребер
2. Для любого значения  $l$  среди первых  $l$  цифр двоичного кода количество «0» не меньше количества «1»

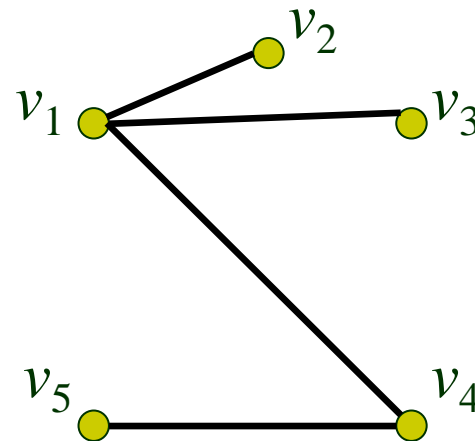


- Любая последовательность «0» и «1», удовлетворяющая свойствам 1 и 2, является кодом некоторого корневого дерева

# Центр, радиус и диаметр дерева

## Эффективный алгоритм

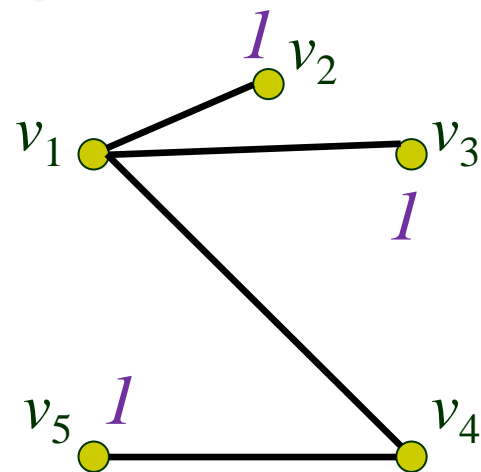
1. Тип  $k = 1$
2. Найти в графе все концевые вершины, пометить их типом  $k$  и удалить
3. Увеличить  $k$  на 1
4. Если в графе осталось больше 2 вершин – перейти на шаг 2, иначе отметить оставшиеся вершины типом  $k$
5. Вершины с максимальным типом  $k$  образуют центр
6. Если в центре 1 вершина, то  $r = k - 1, d = 2k - 2$
7. Если в центре 2 вершины, то  $r = k, d = 2k - 1$



# Центр, радиус и диаметр дерева

## Эффективный алгоритм

1. Тип  $k = 1$
2. Найти в графе все концевые вершины, пометить их типом  $k$  и удалить
3. Увеличить  $k$  на 1
4. Если в графе осталось больше 2 вершин – перейти на шаг 2, иначе отметить оставшиеся вершины типом  $k$
5. Вершины с максимальным типом  $k$  образуют центр
6. Если в центре 1 вершина, то  $r = k - 1, d = 2k - 2$
7. Если в центре 2 вершины, то  $r = k, d = 2k - 1$

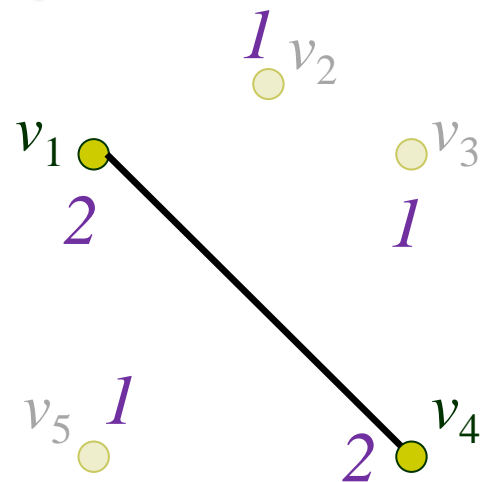




# Центр, радиус и диаметр дерева

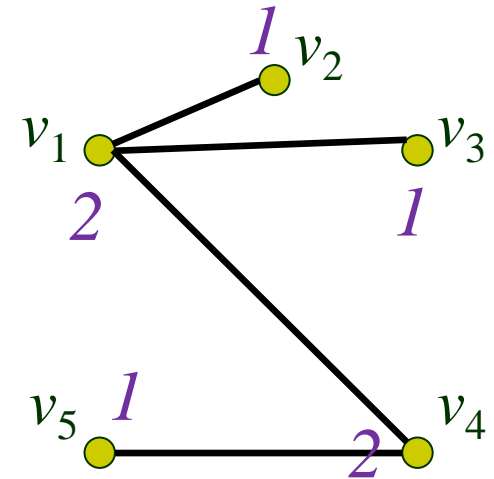
## Эффективный алгоритм

1. Тип  $k = 1$
2. Найти в графе все концевые вершины, пометить их типом  $k$  и удалить
3. Увеличить  $k$  на 1
4. Если в графе осталось больше 2 вершин – перейти на шаг 2, иначе отметить оставшиеся вершины типом  $k$
5. Вершины с максимальным типом  $k$  образуют центр
6. Если в центре 1 вершина, то  $r = k - 1, d = 2k - 2$
7. Если в центре 2 вершины, то  $r = k, d = 2k - 1$



# Центр, радиус и диаметр дерева

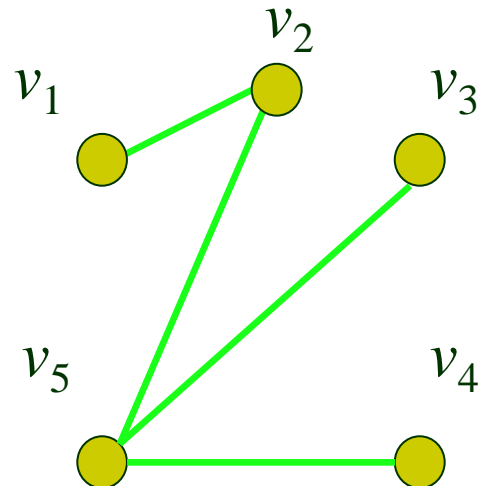
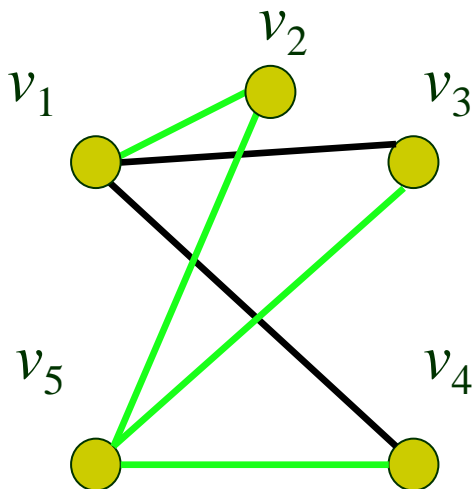
- Центр –  $\{v_1, v_4\}$
- $r = 2$
- $d = 3$



4. Если в графе осталось больше 2 вершин – перейти на шаг 2, иначе отметить оставшиеся вершины типом  $k$
5. Вершины с максимальным типом  $k$  образуют центр
6. Если в центре 1 вершина, то  $r = k - 1, d = 2k - 2$
7. Если в центре 2 вершины, то  $r = k, d = 2k - 1$

# Остовное дерево

- *Остовное дерево* связного графа  $G$  – это дерево, полученное из графа  $G$  удалением некоторых ребер (иными словами, составленное из всех вершин и некоторых ребер графа  $G$ )





Следующая тема:

# Теория графов

Экстремальные задачи