Тема:
Код Прюфера

Сергей Витальевич Рыбин
svrybin@etu.ru

СПбГЭТУ «ЛЭТИ», кафедра «Алгоритмической математики»

24 июня 2023 г.

СПбГЭТУ «ЛЭТИ»
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

Основные понятия

Основные понятия

1. Связный граф, не содержащий циклов, называют **деревом**.

Основные понятия

1. Связный граф, не содержащий циклов, называют **деревом**.
2. У дерева с p вершинами $p - 1$ ребро.

Основные понятия

1. Связный граф, не содержащий циклов, называют **деревом**.
2. У дерева с p вершинами $p-1$ ребро.

Рассмотрим задачу хранения деревьев. Для кодирования структуры будем использовать **код Прюфера**. Код является оптимальным по объему.

Основные понятия

1. Связный граф, не содержащий циклов, называют **деревом**.
2. У дерева с p вершинами $p-1$ ребро.

Рассмотрим задачу хранения деревьев. Для кодирования структуры будем использовать **код Прюфера**. Код является оптимальным по объему.

3. Кодом Прюфера длины $n-2$ (для дерева с n вершинами) называется последовательность из чисел от 1 до n с повторениями.

Основные понятия

1. Связный граф, не содержащий циклов, называют **деревом**.
2. У дерева с p вершинами $p-1$ ребро.

ℹ Рассмотрим задачу хранения деревьев. Для кодирования структуры будем использовать **код Прюфера**. Код является оптимальным по объему.

3. Кодом Прюфера длины $n-2$ (для дерева с n вершинами) называется последовательность из чисел от 1 до n с повторениями.
4. Существует взаимно однозначное соответствие между стягивающими деревьями для графа из n вершин и кодами Прюфера длины $n-2$.

Основные понятия

1. Связный граф, не содержащий циклов, называют **деревом**.
2. У дерева с p вершинами $p - 1$ ребро.

ℹ Рассмотрим задачу хранения деревьев. Для кодирования структуры будем использовать **код Прюфера**. Код является оптимальным по объему.

3. Кодом Прюфера длины $n - 2$ (для дерева с n вершинами) называется последовательность из чисел от 1 до n с повторениями.
4. Существует взаимно однозначное соответствие между стягивающими деревьями для графа из n вершин и кодами Прюфера длины $n - 2$.
5. По каждому дереву с n вершинами можно построить код Прюфера длины $n - 2$ и наоборот.

Основные понятия

1. Связный граф, не содержащий циклов, называют **деревом**.
2. У дерева с p вершинами $p-1$ ребро.

- Рассмотрим задачу хранения деревьев. Для кодирования структуры будем использовать **код Прюфера**. Код является оптимальным по объему.

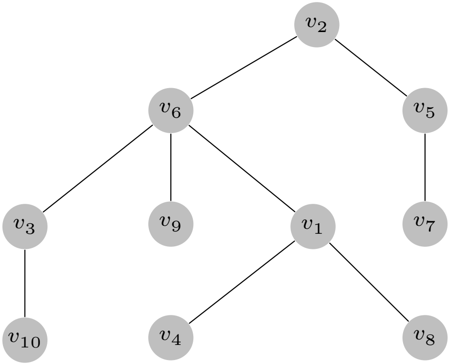
3. Кодом Прюфера длины $n-2$ (для дерева с n вершинами) называется последовательность из чисел от 1 до n с повторениями.
4. Существует взаимно однозначное соответствие между стягивающими деревьями для графа из n вершин и кодами Прюфера длины $n-2$.
5. По каждому дереву с n вершинами можно построить код Прюфера длины $n-2$ и наоборот.
6. **формула Кэли**. Количество пронумерованных деревьев из n вершин равно n^{n-2}.

Построение кода Прюфера

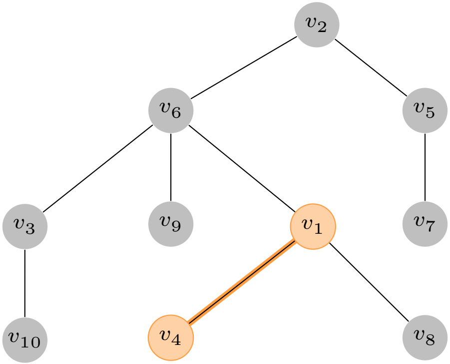
Алгоритм 2.0.

Исходные данные: $\boldsymbol{T}$ — дерево с множеством вершин $\{v_1, \dots, v_n\}$
// *Считаем, что номер вершины v_k равен k*
Результат: последовательность кода Прюфера $P = \{p_1, \dots, p_{n-2}\}$
for $i \leftarrow 1, n-2$ **do**
    $v_k \leftarrow$ *висячая вершина с минимальным номером*
    $p_i \leftarrow s =$ *номер вершины* v_s *— единственного соседа вершины* v_k
    Удаляем из дерева висячую вершину v_k
end for

Пример работы алгоритма

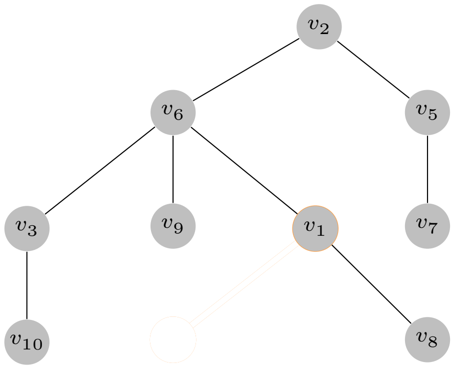


Пример работы алгоритма



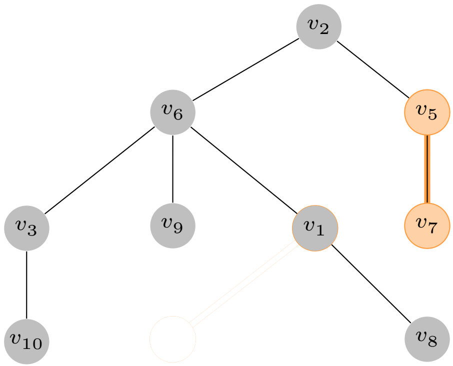
1. (v_4, v_1), $p_1 = 1$

Пример работы алгоритма



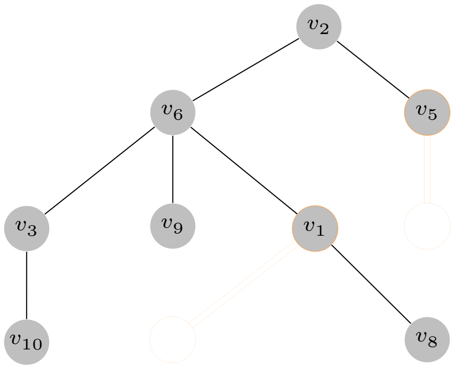
1. (v_4, v_1), $p_1 = 1$

Пример работы алгоритма



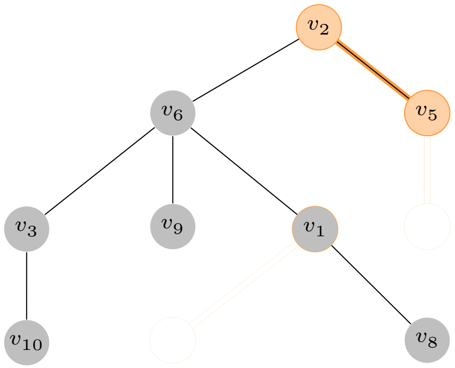
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$

Пример работы алгоритма



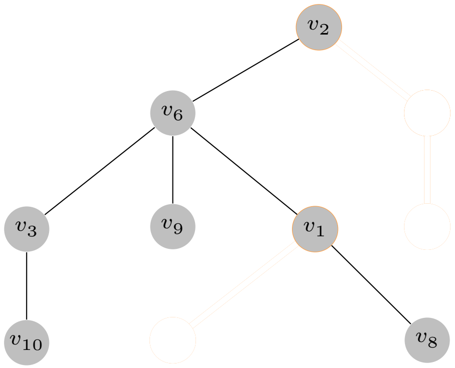
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$

Пример работы алгоритма



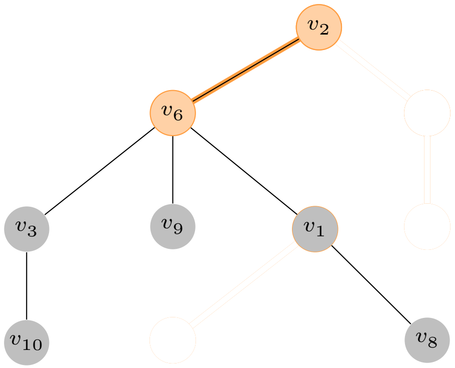
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$

Пример работы алгоритма



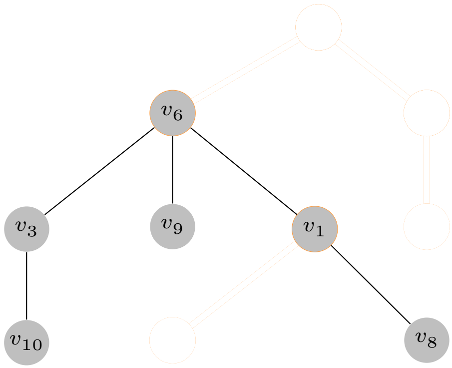
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$

Пример работы алгоритма



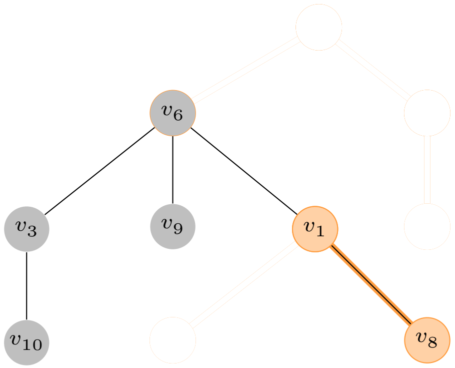
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$

Пример работы алгоритма



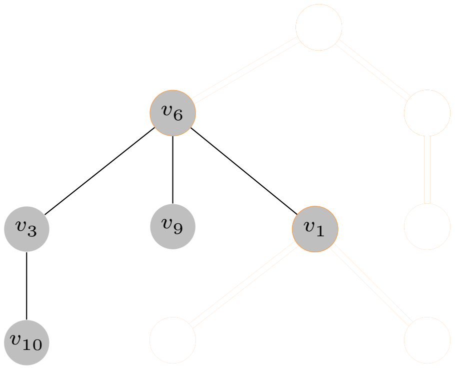
1. $(v_4\,, v_1)$, $p_1 = 1$
2. $(v_7\,, v_5)$, $p_2 = 5$
3. $(v_5\,, v_2)$, $p_3 = 2$
4. $(v_2\,, v_6)$, $p_4 = 6$

Пример работы алгоритма



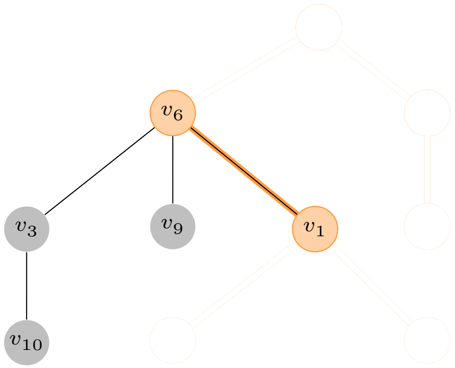
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$

Пример работы алгоритма



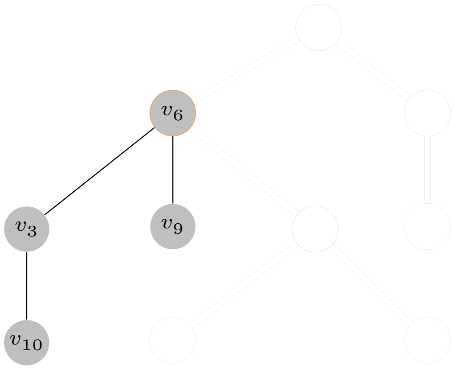
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$

Пример работы алгоритма



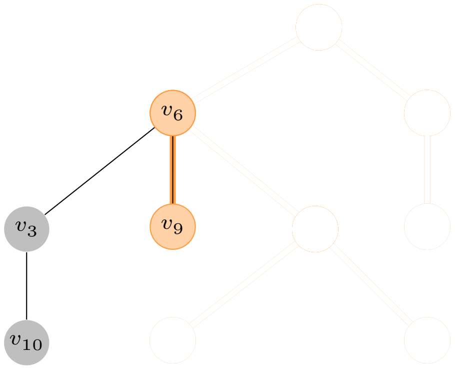
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$
6. (v_1, v_6), $p_6 = 6$

Пример работы алгоритма



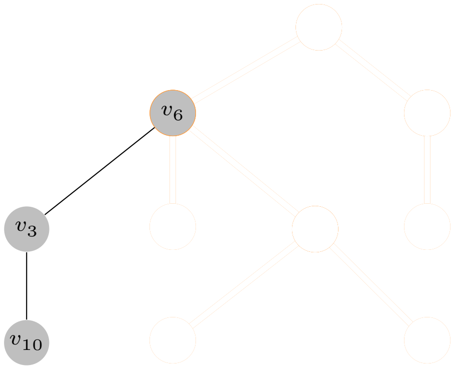
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$
6. (v_1, v_6), $p_6 = 6$

Пример работы алгоритма



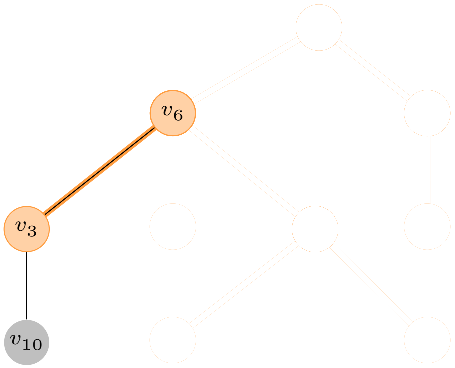
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$
6. (v_1, v_6), $p_6 = 6$
7. (v_9, v_6), $p_7 = 6$

Пример работы алгоритма



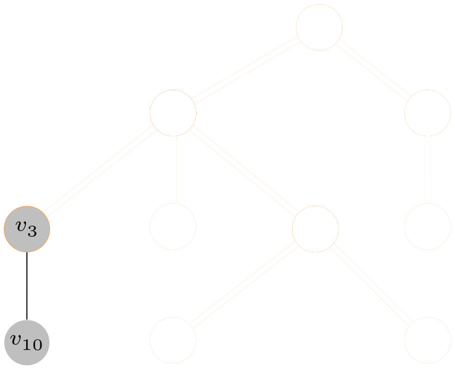
1. $(v_4, v_1),\ p_1 = 1$
2. $(v_7, v_5),\ p_2 = 5$
3. $(v_5, v_2),\ p_3 = 2$
4. $(v_2, v_6),\ p_4 = 6$
5. $(v_8, v_1),\ p_5 = 1$
6. $(v_1, v_6),\ p_6 = 6$
7. $(v_9, v_6),\ p_7 = 6$

Пример работы алгоритма



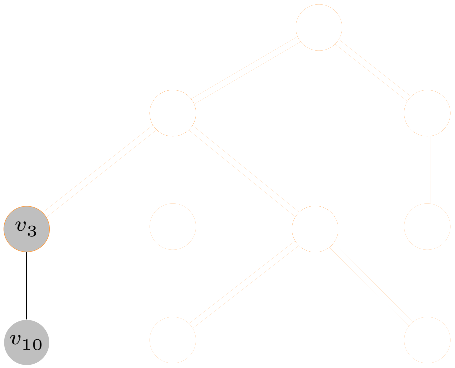
1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$
6. (v_1, v_6), $p_6 = 6$
7. (v_9, v_6), $p_7 = 6$
8. (v_6, v_3), $p_8 = 3$

Пример работы алгоритма



1. $(v_4, v_1),\ p_1 = 1$
2. $(v_7, v_5),\ p_2 = 5$
3. $(v_5, v_2),\ p_3 = 2$
4. $(v_2, v_6),\ p_4 = 6$
5. $(v_8, v_1),\ p_5 = 1$
6. $(v_1, v_6),\ p_6 = 6$
7. $(v_9, v_6),\ p_7 = 6$
8. $(v_6, v_3),\ p_8 = 3$

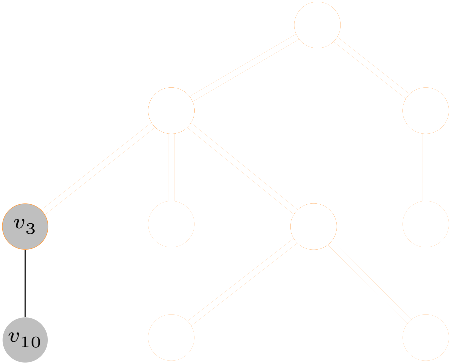
Пример работы алгоритма



1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$
6. (v_1, v_6), $p_6 = 6$
7. (v_9, v_6), $p_7 = 6$
8. (v_6, v_3), $p_8 = 3$

Код Прюфера: $P = \{1, 5, 2, 6, 1, 6, 6, 3\}$.

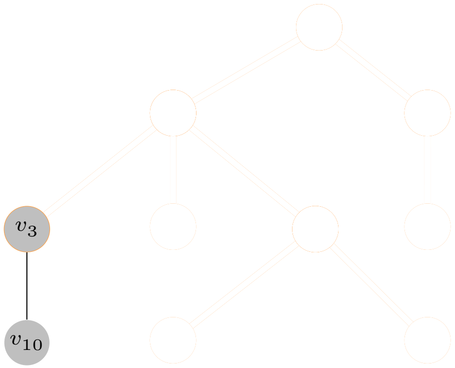
Пример работы алгоритма



1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$
6. (v_1, v_6), $p_6 = 6$
7. (v_9, v_6), $p_7 = 6$
8. (v_6, v_3), $p_8 = 3$

- Код Прюфера: $P = \{1, 5, 2, 6, 1, 6, 6, 3\}$.
- После завершения алгоритма в дереве останутся неудаленными две вершины. Одной из них будет вершина с максимальным номером.

Пример работы алгоритма



1. (v_4, v_1), $p_1 = 1$
2. (v_7, v_5), $p_2 = 5$
3. (v_5, v_2), $p_3 = 2$
4. (v_2, v_6), $p_4 = 6$
5. (v_8, v_1), $p_5 = 1$
6. (v_1, v_6), $p_6 = 6$
7. (v_9, v_6), $p_7 = 6$
8. (v_6, v_3), $p_8 = 3$

Код Прюфера: $P = \{1, 5, 2, 6, 1, 6, 6, 3\}$.

После завершения алгоритма в дереве останутся неудаленными две вершины. Одной из них будет вершина с максимальным номером.

Каждая вершина встречается в коде Прюфера определенное число раз, равное ее степени минус один.

Алгоритм 4.0.

Исходные данные: код Прюфера $P = \{p_1, \ldots, p_{n-2}\}$, $L = \{1, \ldots, n\}$

Результат: дерево $\boldsymbol{T}$ с n вершинами, пронумерованными от 1 до n

// *Как и в предыдущем алгоритме считаем, что номер вершины v_k равен k*

for $i \leftarrow 1, |P| = n - 2$ **do**

  $k \leftarrow \min\{j \mid j \in L,\ j \notin P\}$

  Соединяем вершины v_k и v_{p_i}

  Удаляем элемент k из L.

  Удаляем элемент p_i из P, $P = \{p_{i+1}, p_{i+2}, \ldots, p_{n-2}\}$.

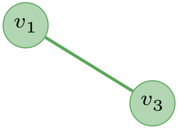
end for

Соединяем две оставшиеся вершины в $\boldsymbol{T}$.

Пример работы алгоритма

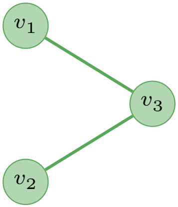
Пример работы алгоритма

1. $P = \{3, 3, 4, 5, 4, 6\}, L = \{1, 2, 3, 4, 5, 6, 7, 8\}, (v_3, v_1)$



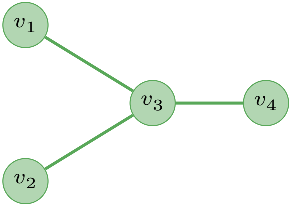
Пример работы алгоритма

1. $P = \{3, 3, 4, 5, 4, 6\}$, $L = \{1, 2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_1)
2. $P = \{3, 4, 5, 4, 6\}$, $L = \{2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_2)



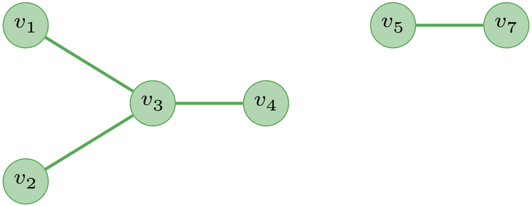
Пример работы алгоритма

1. $P = \{3, 3, 4, 5, 4, 6\}, L = \{1, 2, 3, 4, 5, 6, 7, 8\}, (v_3, v_1)$
2. $P = \{3, 4, 5, 4, 6\}, L = \{2, 3, 4, 5, 6, 7, 8\}, (v_3, v_2)$
3. $P = \{4, 5, 4, 6\}, L = \{3, 4, 5, 6, 7, 8\}, (v_4, v_3)$



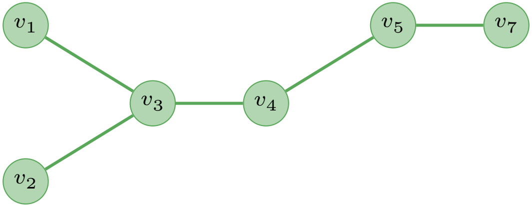
Пример работы алгоритма

1. $P = \{3, 3, 4, 5, 4, 6\}, L = \{1, 2, 3, 4, 5, 6, 7, 8\}, (v_3, v_1)$
2. $P = \{3, 4, 5, 4, 6\}, L = \{2, 3, 4, 5, 6, 7, 8\}, (v_3, v_2)$
3. $P = \{4, 5, 4, 6\}, L = \{3, 4, 5, 6, 7, 8\}, (v_4, v_3)$
4. $P = \{5, 4, 6\}, L = \{4, 5, 6, 7, 8\}, (v_5, v_7)$



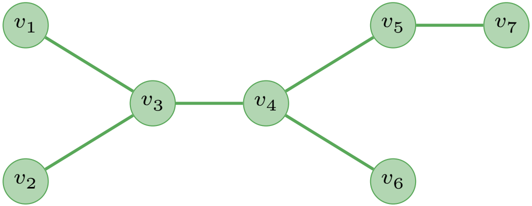
Пример работы алгоритма

1. $P = \{3, 3, 4, 5, 4, 6\}$, $L = \{1, 2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_1)
2. $P = \{3, 4, 5, 4, 6\}$, $L = \{2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_2)
3. $P = \{4, 5, 4, 6\}$, $L = \{3, 4, 5, 6, 7, 8\}$, (v_4, v_3)
4. $P = \{5, 4, 6\}$, $L = \{4, 5, 6, 7, 8\}$, (v_5, v_7)
5. $P = \{4, 6\}$, $L = \{4, 5, 6, 8\}$, (v_4, v_5)



Пример работы алгоритма

1. $P = \{3, 3, 4, 5, 4, 6\}$, $L = \{1, 2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_1)
2. $P = \{3, 4, 5, 4, 6\}$, $L = \{2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_2)
3. $P = \{4, 5, 4, 6\}$, $L = \{3, 4, 5, 6, 7, 8\}$, (v_4, v_3)
4. $P = \{5, 4, 6\}$, $L = \{4, 5, 6, 7, 8\}$, (v_5, v_7)
5. $P = \{4, 6\}$, $L = \{4, 5, 6, 8\}$, (v_4, v_5)
6. $P = \{6\}$, $L = \{4, 6, 8\}$, (v_6, v_4)



Пример работы алгоритма

1. $P = \{3, 3, 4, 5, 4, 6\}$, $L = \{1, 2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_1)
2. $P = \{3, 4, 5, 4, 6\}$, $L = \{2, 3, 4, 5, 6, 7, 8\}$, (v_3, v_2)
3. $P = \{4, 5, 4, 6\}$, $L = \{3, 4, 5, 6, 7, 8\}$, (v_4, v_3)
4. $P = \{5, 4, 6\}$, $L = \{4, 5, 6, 7, 8\}$, (v_5, v_7)
5. $P = \{4, 6\}$, $L = \{4, 5, 6, 8\}$, (v_4, v_5)
6. $P = \{6\}$, $L = \{4, 6, 8\}$, (v_6, v_4)
7. $P = \{\}$, $L = \{6, 8\}$, (v_6, v_8)