

Задание №1

У вас есть таблица *StockQuotes(company TEXT, week INT, share_price INT)*. Строка в этой таблице говорит о том, что стоимость акции компании *company* в неделю номер *week* составляла *share_price*.

Назовем индексом в данную неделю среднее арифметическое роста стоимости одной акции по всем компаниям сравнительно с предыдущей неделей. То есть, если одна акция компании А подорожала на 100 единиц, а акция компании В подешевела на 50 единиц, то индекс равен 25.

Компании, чьи акции выросли выше индекса, назовем успешными в данную неделю. Под ростом понимается простое превышение показателя индекса, даже если оно отрицательное. Если компания была успешной три недели подряд то будем говорить, что она сделала успешную серию. Успешные серии могут пересекаться. Так, если компания была успешной 5 недель подряд, то у неё было 3 успешных серии.

Вам нужно посчитать для каждой компании количество успешных серий и вывести в результат два столбца - название компании и количество успешных серий. Компании, у которых не было успешных серий, выводить в результат не надо.

Задание №2

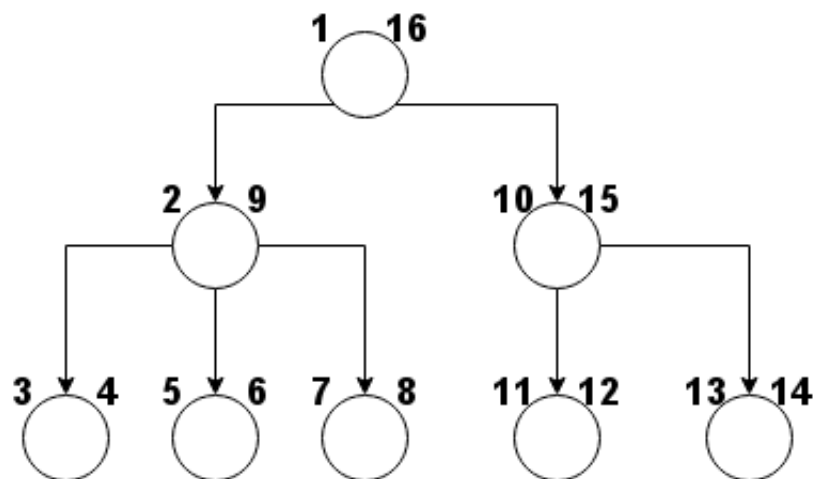
В СУБД PostgreSQL есть таблицы:

```
KeywordLtree(id INT PRIMARY KEY, value TEXT, path ltree);
```

```
KeywordSets(id INT PRIMARY KEY, value TEXT, lft INT, rgt INT);
```

В таблице *KeywordLtree* записано небольшое дерево, таблица *KeywordSets* пуста. Корневая вершина всегда имеет идентификатор *id=0*. Преобразуйте одним запросом дерево, представленное структурными метками [ltree](#) в дерево, представленное в виде вложенных множеств. Напишите один запрос, который сгенерирует содержимое таблицы *KeywordSets*. Можно использовать одно представление. Хранимые процедуры использовать запрещено.

Правила генерации левого и правого номеров: оба номера берутся из одной монотонно возрастающей последовательности, начинающейся с 1, левый номер назначается сразу после входа в вершину при обходе в глубину, правый непосредственно перед возвращением к родителю. Смотрите картинку в качестве примера:



В результате должны быть следующие столбцы, в порядке очередности:

1. id вершины, такой же как и в таблице KeywordLtree
2. value вершины, такое же, как и в таблице KeywordLtree
3. левый номер вершины
4. правый номер вершины