Динамическая связность

Иванов Кирилл
371 группа

Задача 1 (a). Придумайте рекурсивную процедуру $fall(v)$, которая для вершины v, такой, что $N_1(v) = \varnothing$, "роняет"v на правильный уровень BFS-дерева, корректно обновляет уровни соседей v и "роняет" те вершины, чей уровень изменился при падении v.

Решение. Будем предполагать, что перед вызовом $fall(v)$ будет пройдена проверка на непустоту $N_2(v)$, иначе у нас образуется новая компонента связности.

Algorithm 1

```
1:  function FALL(v)
2:      l(v) ← l(v) + 1
3:      for u ∈ N_2(v) do
4:          N_2(u) ← N_2(u) \ {v}
5:          N_3(u) ← N_3(u) ∪ {v}
6:      for u ∈ N_3(v) do
7:          N_1(u) ← N_1(u) \ {v}
8:          N_2(u) ← N_2(u) ∪ {v}
9:      N_1(v) ← N_2(v)
10:     N_2(v) ← N_3(v)
11:     N_3(v)_old ← N_3(v)
12:     N_3(v) ← ∅                          ▷ Заполнится при вызове fall от детей
13:     for u ∈ {w | w ∈ N_3(v)_old and N_1(u) = ∅} do
14:         fall(u)
```

Задача 1 (b). Докажите, что если в графе n вершин и m рёбер изначально, на все обновления суммарно при удаления m рёбер уйдет время $O(mn)$.

Доказательство. Обработка одной вершины внутри функции $fall(n)$ занимает $O(deg(n))$, где $deg(n)$ – степень вершины v. Максимальное количество вызовов функции $fall$, инициированных данной вершиной v, сравнимо с n (в случае, когда BFS-tree «почти бамбук», его высота сравнима с n) $\Rightarrow$ время работы функции $fall$ для вершины v равно $O(n\ deg(v))$. Удаление ребра занимает $O(1)$. Итого: $O(m + \sum_{v \in V} n\ deg(v)) = O(m + n \sum_{v \in V} deg(v)) = O(m + nm) = O(mn)$ □

Задача 1 (c). Пусть вместо всего BFS-дерева нам разрешено хранить только BFS-дерево с d уровнями, т.е. структура будет поддерживать только расстояния до вершин v, такие, что $d(s, v) \leq d$. Докажите, что суммарное время на все апдейты в этом случае равно $O(md)$.

Доказательство. Обработка одной вершины внутри функции $fall(n)$ занимает $O(deg(n))$, где $deg(n)$ – степень вершины v. Максимальное количество вызовов функции $fall$, инициированных данной вершиной v, сравнимо с d (в случае, когда BFS-tree «почти бамбук», его высота сравнима с d) $\Rightarrow$ время работы функции $fall$ для вершины v равно $O(d\ deg(v))$. Удаление ребра занимает $O(1)$. Итого: $O(m + \sum_{v \in V} d\ deg(v)) = O(m + d \sum_{v \in V} deg(v)) = O(m + dm) = O(md)$ □