Dawid Pawliczek

Lista 2, Zadanie 6

Treść

Dane jest n-wierzchołkowe drzewo $T = (V, E)$ oraz liczba całkowita $k \geq 1$. Trzeba **p**-pokolorować możliwie wiele wierzchołków, tak aby *na każdej prostej ścieżce* znajdowało się *co najwyżej* k p-pokolorowanych wierzchołków.

Intuicja

Oznaczmy przez *liść* wierzchołek stopnia 1. Każda prosta ścieżka w drzewie ma dwa końce, które możemy *rozszerzyć* do liści. Jeśli będziemy kolorować wyłącznie liście, a następnie usuwać je z drzewa i powtarzać tę operację, to w jednym ciągu usuwania *obydwa* końce dowolnej ścieżki otrzymają p-kolor najwyżej raz. Wykonując tę procedurę $\lfloor k/2 \rfloor$ razy zabezpieczamy się, że na każdej ścieżce znajdą się co najwyżej k p-wierzchołków. Gdy k jest nieparzyste, można dodatkowo pokolorować *dowolny* jeszcze niepokolorowany wierzchołek.

Algorytm

Algorithm 1 MAXPCOLOUR(T, k)

Require: drzewo $T = (V, E)$, liczba k
Ensure: zbiór p-pokolorowanych wierzchołków

for $i \leftarrow 1$ **to** $\lfloor k/2 \rfloor$ **do**
    $L \leftarrow$ wszystkie liście bieżącego T
    pokoloruj każdy $v \in L$
    usuń wierzchołki L z T
if k jest nieparzyste **then**
    pokoloruj dowolny niepokolorowany $v \in V$
end if
return zaznaczony zbiór

Poprawność

- **Górne ograniczenie.** Po każdej iteracji usuwania liści długość dowolnej ścieżki skraca się o co najmniej dwa wierzchołki (oba końce). Stąd ścieżka może zawierać co najwyżej $2 \cdot \lfloor k/2 \rfloor + (k \bmod 2) = k$ p-pokolorowanych wierzchołków.
- **Maksymalność.** Na drodze od każdego liścia w głąb drzewa pokryliśmy dokładnie $\lfloor k/2 \rfloor$ poziomów, więc dołożenie *jakiegokolwiek* kolejnego liścia przekroczyłoby limit k na ścieżce łączącej dwa liście. Dodatkowy wierzchołek (gdy k nieparzyste) te limit k dokładnie domyka.

Zatem algorytm koloruje optymalną liczbę wierzchołków. □

Złożoność

Każdy wierzchołek jest usuwany najwyżej raz. Łączny czas i pamięć wynoszą $O(n)$.