AiSD L2

Maurycy Borkowski

17.03.2021

Zadanie 2

```
I <- lista par odcinkow
S = set()
sorted(A, key = lambda x: x.fk)
S.add(A[0])

last = A[0].k
for i in range(2,n):
    if A[i].p > last:
        S.add(A[i])
        last = A[i].k
```

Sortujemy odcinki rosnąco po ich końcach.

Bierzemy zachłannie odcinki jeżeli możemy tzn. nie nachodzą na siebie. Możemy go wziąć jeżeli początek jest większy niż koniec ostanio wziętego (ostatnio wzięty ma największy koniec w S, na początku je sortowaliśmy).

Oznaczmy jako I_0 odcinek o najmniejszym końcu. ze wszystkich odcinków i I_k odcinek o najmniejszym końcu w dowolnym rozwiązaniu optymalnym B. Zauważmy, że rozwiązanie $A = B \setminus \{I_k\} \cup \{I_0\}$ też jest optymalne bo dodany odcinek nie nachodzi się na żaden inny (B było poprawnym rozwiązaniem i $k_0 < k_k$) oraz $|A| = |B|$.

Argumentacje możemy indukcyjnie powtórzyć dla podproblemu ze zbiorem odcinków $I' = \{j \in I : p_j \geqslant k_1\}$, tam ponownie pokazujemy, że zachłanne wzięcie najmniejszego (względem końca) odcinka nie zepsuje nam optymalności rozwiązania.

Zadanie 3

```
M = a*b + 1
while aPb > 0:
    k = min([i if 1/i <= a/b for i in range(1,M)])
    aPb = aPb - 1/k
```

Nie prowadzi to do rozwiązania optymalnego, kontrprzykład $\frac{5}{121}$:
Zachłannie weźmiemy:

$$\frac{5}{121} = \frac{1}{25} + \frac{1}{757} + \frac{1}{763309} + \frac{1}{873\,960\,180\,913} + \frac{1}{1\,527\,612\,795\,642\,093\,418\,846\,225}$$

Natomiast optymalne rozwiązanie wynosi:

$$\frac{5}{121} = \frac{1}{33} + \frac{1}{121} + \frac{1}{363}$$

Musimy pokazać dwie rzeczy do znajdowania rozwiązania:

- suma ułamków wybranych przez algorytm będzie się sumowała do $\frac{a}{b}$
- ułamki wybrane przez algorytm będą unikalne

Dowód.

Sumowanie

Pokażemy, że licznik $\frac{a}{b}$ będzie zbiegał do 1.
Oznaczmy, przez $\frac{1}{u}$ ułamek zachłannie brany przez algorytm tj:

$$\frac{1}{u-1} < \frac{a}{b} \leqslant \frac{1}{u} \tag{1}$$

Po odjęciu go zostanie nam reszta:

$$\frac{a}{b} - \frac{1}{u} = \frac{au-b}{bu}, \tag{2}$$

dalej z (1):

$$\frac{1}{u-1} < \frac{a}{b} \implies a > au - b \tag{3}$$

Wiemy, że $a, u, b \in \mathbb{N}$ stąd wiemy, że liczniki będą zbiegały do 1.

Unikalność

Zakładamy niewprost, że dwa razy weźmiemy jakiś ułamek $\frac{1}{u}$, ale

$$1 \geqslant \frac{1}{u-1} \iff 2 \geqslant 1 + \frac{1}{u-1} \iff 2 \geqslant \frac{u-1+1}{u-1}$$
$$\iff 2 \geqslant \frac{u}{u-1} \iff \frac{2}{u} = \frac{1}{u} + \frac{1}{u} \geqslant \frac{1}{u-1}$$

otrzymujemy sprzeczność bo to oznacza, że mogliśmy odjąć $\frac{1}{u-1}$ czyli większy ułamek niż $\frac{1}{u}$ gdy odejmowaliśmy $\frac{1}{u}$ po raz pierwszy. □

Zadanie 4

Lemat 1. *Dowolne optymalne kolorwanie możemy sprowadzić do optymalnego kolorwania, z pokolorwanymy liściami.*

Dowód. K - dowolne optymalne kolorwanie, tż. liść u nie jest pokolorwany. Niech w oznacza najbliższy pokolorwany wierzchołek u. Zamieniamy je kolorowaniem pokażemy, że nowe kolorwanie K' dalej jest optymalne. Wystarczy, więc pokazać, że jest poprawne. Załóżmy niewprost, że nie jest:
Istenieje ścieżka S z u tż. ma ponad k kolorowych wierzchołków. Oznaczmy jeszcze przez P ścieżkę z u do w, tam jest tylko jeden pokolorwany wierzchołek (w najbliższy kolorowy u). Rozważmy ścieżkę $L = S + P - (S \cap P)$. Ta ścieżka ma dokładnie tyle samo kolorowych wierzchołków z kolorwaniem K jak i z K' (zarówno, u i w są w tej ścieżce). Zatem istnieje, ścieżka L po kolorwaniu K, tż. ma ponad k kolorowych wierzchołków. Sprzeczność. □

Lemat 2. *Dodanie dowolnych pokolorwanych liści do G nie psuje optymalności w $G' = G + liscie$ (G z dodanymi liśćmi) z $k' = k + 2$.*

Dowód. Oznaczmy przez K kolorwanie optymalne w G, pokażemy, że $K' = K + liscie$ jest optymalne w $G' = G + liscie$ z $k' = k + 2$.
Z Lematu 2. BSO można założyć, że optymalne kolorwanie K', będzie miało pokolorwane wszystkie liście).
Dodanie pokolorowanych liści może zwiększyć liczbę pokolorowanych wierzchołków w dowolnej ścieżce o co najwyżej 2. Jedyne nowo powstałe ścieżki to takie z conajmniej jednym nowym wierzchołkiem na końcu. Więc każda będzie miała co najwyżej $k + 2 = k'$ (nowe wierzchołki są kolorowane).

Nie możemy zwiększyć K' kolorując nowe wierzchołki (kolorujemy już wszystkie), więc jeżeli K' nie jest optymalne, oznacza to, że możemy jeszcze pokolorwać wierzchołek w $G' - liscie = G$, ale to przeczy optymalności kolorwania K w G z k. □

Czyli chcemy wziąć takie poddrzewo $\widetilde{G} \subset G$, że dodając $k \div 2$ razy (tyle możemy dodać startując od $k = 0$ lub $k = 1$) synów (w G) liści obecnego poddrzewa, otrzymamy G i kolorwać za każdym razem liście obecnego poddrzewa.

Zauważmy, że możemy to robić od końca (i tak pokolorujemy wszystkie potrzebne wierzchołki), czyli z G usunąć liście, pokolorwać je itd. Na końcu jeszcze jak zostaje nam $k = 1$ kolorujemy dowolny wierzchołek z pozostałego $\widetilde{G}$ po *strzyżeniu topologicznym* G.

```
while k > 1 and G:
    koloruj_liscie(G)
    usun_liscie(G)
    k-=2
if k == 1 and G:
    koloruj_dowolny(G)
```

$\mathcal{O}(n)$

Zadanie 5

Dowód podzielimy da dwie części:

1. W każdym momencie działania algorytmu nie będzie cyklu
2. Dla każdej składowej będzie MST

Dowód.

Brak cyklu

Załóżmy niewprost, że podczas działania algorytmu, w jakiejś spójnej składowej, powstał cykl C. Oznacza to, że powstał on na skutek połączenia n (super) wierzchołków $v_0, v_1, \ldots v_n$ będącymi kolejnymi wierzchołkami w C. Niech $e_0, e_1, \ldots e_n$ będą kolejnymi krawędziami takimi, że e_k łączy v_k i $v_{k+1 \ mod \ n+1}$. Z tego jak działa algorytm wynika, że

$$Cost(e_0) < Cost(e_1) < \ldots < Cost(e_n) < Cost(e_0)$$

sprzeczność

Minimalność

Załóżmy, nie wprost, że w algorytmie dodajemy krawędź e łącząc super wierzchołki V, W i e psuje minimalność.
Wtedy jeżeli do rozwiązania optymalnego dodamy e otrzymamy cykl C. Na tym cyklu leży conajmniej jedna krawędź e' (różna od e) łącząca pewne wierzchołki z V i W (inaczej OPT nie byłby spójny). Rozpatrzmy przypadki:

- $Cost(e) < Cost(e')$
  Sprzeczność. OPT nie jest OPT-em.
- $Cost(e) > Cost(e')$
  Sprzeczność. Zgodnie z działaniem algorytmu powinniśmy wziąć krawędź e' jako minimalną między V i W,

□