

Popis

1. Ze vzorků S_i si vytvoříme "automat" ve smyslu AC. Vybudujeme zároveň i zpětnou funkci+ ($zh+$), která tranzitivně bude obsahovat zpětné hrany, ale vrcholy, které nejsou koncem žádného vzorku, přeskočí. U každého stavu si též budeme pamatovat, jaké zpětné hrany+ na něj míří. Ve stavech, ze kterých žádná $zh+$ hrana nevede, přidáme takovou hranu do stavu 0. Pokud nalezneme nějaký výskyt, tak si v aktuálním stavu inkrementujeme akumulátor a takto projdeme celý text. Četnost stavu 0 ponecháme na 0.
2. Na konci zavoláme funkci *cclt* na kořen, pak ve stromě nalezneme vzorky a vyzvedneme si u nich jejich četnost. Funkce *cclt* má rekursivní charakter a je definována následovně:

```
def cclt(s):
    for state k in s.rev_zhp:
        freq[s] += cclt(k);
    return freq[state]
```

Korektnost

Každý běh *cclt* někdy skončí, neboť máme konečný počet vzorků (tedy i stavů, kde může vzorek končit) a k cyklení dojít nemůže, neboť stavy, mezi kterými vede $zh+$ hrana nemůžou mít stejnou velikost. Zároveň taky nevymecháme žádné výskyty, neboť pokud ze stavu s vedou opačně $zh+$ hrany k množině stavů A , pak neexistují jiné vzorky, než v A , které by mohly přispět k četnosti s .

Předpokládáme, že $zh+$ je intrazitivní, resp. tvoří strom. Jinak by vznikl problém s pořadím zpracování. To je však zřejmé, neboť pokud by tuhle vlastnost neměla, tak z nějakého stavu by musely vést dvě $zh+$, což je ve sporu s konstrukcí.

Složitost

1. fáze trvá $O(T + \sum S_i)$, neboť se jedná jen o lehce upraveného AC a konstrukci $zh+$ jsme řešili na cvičení.
2. fáze trvá jen dokonce $O(|S|)$, neboť děláme konstantní práci na každém vrcholu, kde končí vzorek. Celkem $O(T + \sum S_i)$, neboť $\sum S_i \geq \sum_{S_i} 1 = |S|$.