

ALGORYTMY I STRUKTURY DANYCH

ZBIORY ROZŁĄCZNE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. [*] Dany jest graf nieskierowany $G = (V, E)$ złożony z k spójnych składowych. Dla grafu tego wywołano następującą procedurę obliczającą liczbę składowych spójności:

```
function CONECTED-COMPONENTS (graph  $G$ )  $\mapsto$  int
{
    DisjointSets  $S$ ;
     $S.init(G.V)$ ;
    for-each  $(u, v) \in G.E$  do
    {
         $x \leftarrow S.find(u)$ ;
         $y \leftarrow S.find(v)$ ;
        if  $x \neq y$  then  $S.union(x, y)$ ;
    }
    return  $S.count-sets()$ ;
}
```

Ile razy nastąpi wywołanie *find* a ile razy *union*? Odpowiedź uzależnij od wielkości $|V|$, $|E|$ oraz k .

2. [**] Dany jest graf nieskierowany $G = (V, E)$ z ważonymi krawędziami. Jak za pomocą operacji na zbiorach rozłącznych wyznaczyć minimalne drzewo rozpinające grafu? Omów algorytm rozwiązujący to zadanie i przeanalizuj jego złożoność. Ile razy nastąpi wywołanie *find()* a ile razy *union()*? Odpowiedź uzależnij od wielkości $|V|$ i $|E|$.
3. [**] Zaproponuj efektywną strukturę danych do wykonywania ciągów podanych operacji na dynamicznie zmieniającym się podziale zbioru $\{1, 2, \dots, n\}$:

- **init()** — utworzenie podziału $\{\{1\}, \{2\}, \dots, \{n\}\}$;
- **union(A,B)** — połączenie zbiorów A i B bieżącego podziału w jeden zbiór;
- **find(x)** — wyznaczenie zbioru, do którego należy element x ;
- **find-smallest()** — wyznaczenie zbioru bieżącego podziału, który ma najmniej elementów.

Pesymistyczna złożoność tych operacji powinna być rzędu $O(\log n)$?

4. **[**]** Zaimplementuj operacje *union* i *find* dla drzewiastej reprezentacji zbiorów rozłącznych ze zbalansowanym łączeniem i kompresją ścieżek. Twoje procedury mają działać w miejscu (bez używania rekurencji).
5. **[*]** Dana jest rodzina n jednoelementowych zbiorów rozłącznych. Zbiory te będziemy pamiętali w postaci drzewiastej ze zbalansowanym łączeniem. Skonstruuj ciąg $n-1$ operacji *union*, by w ich wyniku powstał jeden zbiór reprezentowany przez n elementowe drzewo o wysokości $\lfloor \log n \rfloor$.
6. **[***]** Niech dany będzie ciąg m instrukcji *union* i *find*, w którym wszystkie instrukcje *union* występują przed instrukcjami *find*. Udowodnij, że koszt czasowy wykonania tych operacji na drzewiastych strukturach reprezentujących zbiory rozłączne ze zbalansowanym łączeniem i kompresją ścieżek wynosi $\Theta(m)$.
7. **[***]** Operacja *extract*(i) ma usuwać element i -ty ze zbioru, do którego on aktualnie należy i tworzyć nowy jednoelementowy zbiór $\{i\}$. Podaj implementację operacji *extract*, która pozwoli na wykonanie ciągu m instrukcji *union*, *find* i *extract* na n -elementowym zbiorze danych w czasie $O(m \log^* n)$.