Lista zadań. Nr 7. 15 maja 2024

Algorytmy i Struktury Danych

IIUWr. II rok informatyki.

1. (1pkt) Niech σ będzie ciągiem instrukcji $Union$ i $Find$, w którym wszystkie instrukcje $Union$ występują przed instrukcjami $Find$. Udowodnij, że algorytm oparty na strukturach drzewiastych wykonuje σ w czasie proporcjonalnym do długości σ.

2. (2pkt) Rozważamy ciągi operacji $Insert(i)$, $DeleteMin$ oraz $Min(i)$ wykonywanych na S - podzbiorze zbioru $\{1, \ldots, n\}$. Obliczenia rozpoczynamy z $S = \emptyset$. Instrukcja $Insert(i)$ wstawia liczbę i do S. Instrukcja $DeleteMin$ wyznacza najmniejszy element w S i usuwa go z S. Natomiast wykonanie $Min(i)$ polega na usunięciu z S wszystkich liczb mniejszych od i.

   Niech σ będzie ciągiem instrukcji $Insert(i)$, $DeleteMin$ oraz $Min(i)$ takim, że dla każdego i, $1 \leq i \leq n$, instrukcja $Insert(i)$ występuje co najwyżej jeden raz. Mając dany ciąg σ naszym zadaniem jest znaleźć ciąg liczb usuwanych kolejno przez instrukcje $DeleteMin$. Podaj algorytm rozwiązujący to zadanie.
   Uwaga: Zakładamy, że cały ciąg σ jest znany na początku, czyli interesuje nas wykonanie go *off-line*.
   WSKAZóWKA: Rozdział 4.8 z książki Aho,... .

3. (2pkt) Rozważamy ciągi instrukcji: $Link(r, v)$ oraz $Depth(v)$ wykonywanych na lesie rozłącznych drzew o wierzchołkach z etykietami ze zbioru $\{0, ..., n-1\}$ (różne wierzchołki mają różne etykiety). Operacja $Link(r, v)$ czyni r, korzeń jednego z drzew, synem v, wierzchołka innego drzewa. $Depth(v)$ oblicza głębokość wierzchołka v.

   Naszym celem jest napisanie algorytmu, który dla danego ciągu σ wypisze w sposób on-line wyniki instrukcji $Depth$ (tzn. wynik każdej instrukcji $Depth$ ma być obliczony przed wczytaniem kolejnej instrukcji z ciągu σ). Pokaż jak zastosować drzewiastą strukturę danych dla problemu $Union - Find$ do rozwiązania tego problemu.
   WSKAZóWKA: Rozdział 4.8 z książki Aho,... .

4. (1pkt) Rozważ taką wersję wykonywania kompresji ścieżek, w której wierzchołki wizytowane podczas wykonywania operacji $Find$ podwieszane są pod własnego dziadka. Czy analiza złożoności przeprowadzona na wykładzie da się zastosować w tym przypadku?

5. (1,5pkt) Dany jest graf $G = (V, E)$, wyróżniony wierzchołek v oraz ciąg jego krawędzi $e_i = \{v_i, u_i\}$, $(i = 1, 2, \ldots, m)$. Ułóż algorytm, który dla każdego wierzchołka u wyznaczy minimalną wartość j, taką że po usunięciu z grafu G krawędzi $e_1, \ldots, e_j$, nie istnieje ścieżka łącząca wierzchołki u i v.