

Zajęcia D-5: "Silnie spójne składowe"

Cel zajęć i efekty uczenia

Główne cele zajęć / materiał do opanowania:

- Pojęcie silnie spójnej składowej, algorytm Kosaraju znajdowania silnie spójnych składowych

Zadania do rozwiązania na sprawdzarce

Euro 2102

Dany jest graf skierowany oraz pewna liczba zapytań par wierzchołków. Dla każdej pary odpowiedzieć, czy istnieje ścieżka w obie strony między nimi.

Plan zajęć

Szacunkowy czas trwania: 2 godziny lekcyjne.

1. Pojęcie silnie spójnej składowej
 - *Samo pojęcie jest dość intuicyjnie, dobry przykład-rysunek na tablicy powinien wystarczyć uczniom do zrozumienia.*
 - *Warto w tym momencie zadać pomocnicze zadanie: "Dany jest graf skierowany - wyznaczyć silnie spójną składową wierzchołka numer 1." Aby je rozwiązać, wystarczy po prostu znaleźć wszystkie wierzchołki osiągalne z 1, a potem wszystkie wierzchołki, z których można dojść do 1, odwracając krawędzie grafu. Znalezienie tego uproszczonego algorytmu daje dobrą intuicję przed właściwym.*
2. Algorytm Kosaraju znajdowania silnie spójnych składowych
 - *Jest to najbardziej znany algorytm rozwiązujący ten problem, dostępny w wielu miejscach, w tym na [Wikipedii](#). Warto zwrócić uczniom uwagę na kwestie techniczne, w tym najbardziej typowe błędy, jakie można w nim popełnić:*
 - *Algorytm zawiera dwie rekurencyjne procedury DFS. Bardzo częstą pomyłką jest nazwanie ich podobnie (np. "DFS1" i "DFS2"), skopiowanie jednego, otrzymać drugi, ale pozostawienie oryginalnej nazwy przy wywoływaniu rekurencyjnym. W ten sposób np. "DFS2" wywołuje rekurencyjnie "DFS1" zamiast siebie samego.*
 - *Pomiędzy pierwszym a drugim DFS-em trzeba odwrócić graf, albo po prostu przechowywać w pamięci dwie kopie grafu, w tym jedną z odwróconymi krawędziami. Łatwo zapomnieć o odwróceniu, albo też drugi DFS (zwłaszcza przy skopiowaniu!) wywołać znowu na pierwszym grafie.*
 - *Pierwszy DFS znajduje pewną kolejność wierzchołków, drugi DFS należy wykonywać na wierzchołkach w (odwróconej) właśnie kolejności. Łatwo się pomylić i wywołać DFS-a po prostu w kolejności od największych*

numerów wierzchołków [napisać "for(i=n-1; i>=0; i--) dfs(i); zamiast for(i=n-1; i>=0; i--) dfs(tablica_kolejnosci[i]);]. Jest to o tyle problematyczne, że algorytm z tym błędem (podobnie jak w przypadku dwóch powyżej podanych) działa poprawnie na małych przykładach, więc trudno jest zrozumieć, czemu sprawdzarka zgłasza błędną odpowiedź.

3. [opcjonalnie] Dowód poprawności algorytmu

- *Łatwo widać, że algorytm jest liniowy, znacznie trudniej pokazać, że zawsze właściwie znajduje silnie spójne składowe. Być może warto w ogóle pominąć ten konkretny dowód poprawności.*
- *Jeśli zdecydujemy się dowód przedstawić uczniom, polecam zaczerpnąć go z książki Jeffa Ericksona "Algorithms", za to odradzam niezbyt fortunny i zagmatwany argument z podręcznika "Wprowadzenie do algorytmów" Cormena i innych.*