Lista zadań. Nr 2. 5 marca 2013

Algorytmy i Struktury Danych

IIUWr. II rok informatyki.

1. (0pkt) Przeczytaj notatkę do wykładu o algorytmach zachłannych.

2. (1pkt) Danych jest n odcinków $I_j = \langle p_j, k_j \rangle$, leżących na osi OX, $j = 1, \ldots, n$. Ułóż algorytm znajdujący zbiór $S \subseteq \{I_1, \ldots, I_n\}$, nieprzecinających się odcinków, o największej mocy.

3. (1pkt) Rozważ następującą wersję problemu wydawania reszty: dla danych liczb naturalnych a, b ($a \leq b$) chcemy przedstawić ułamek $\frac{a}{b}$ jako sumę różnych ułamków o licznikach równych 1. Udowodnij, że algorytm zachłanny zawsze daje rozwiązanie. Czy zawsze jest to rozwiązanie optymalne (tj. o najmniejszej liczbie składników)?

4. (2pkt) Udowodnij poprawność algorytmu Boruvki(Sollina).

5. (2pkt) Ułóż algorytm, który dla danego spójnego grafu G oraz krawędzi e sprawdza w czasie $O(n + m)$, czy krawędź e należy do jakiegoś minimalnego drzewa spinającego grafu G. Możesz założyć, że wszystkie wagi krawędzi są różne.

6. (2pkt) System złożony z dwóch maszyn A i B wykonuje n zadań. Każde z zadań wykonywane jest na obydwu maszynach, przy czym wykonanie zadania na maszynie B można rozpocząć dopiero po zakończeniu wykonywania go na maszynie A. Dla każdego zadania określone są dwie liczby naturalne a_i i b_i określające czas wykonania i-tego zadania na maszynie A oraz B (odpowiednio). Ułóż algorytm ustawiający zadania w kolejności minimalizującej czas zakończenia wykonania ostatniego zadania praz maszynę B

7. (2pkt) Ułóż algorytm, który dla danych liczb naturalnych a i b, sprawdza, czy zachłanna strategia dla problemu wydawania reszty jest poprawna, gdy zbiór nominałów jest równy $X = \{1, a, b\}$.

8. (2pkt) Dla ważonego drzewa $T = (V, E; c)$, gdzie $c : V \to \mathcal{R}_+$, określamy jego *zewnętrzną długość* $EL(T)$ jako:

$$EL(T) = \sum_{v - liść\ T} c(v) \cdot d(v),$$

gdzie $d(v)$ jest długością ścieżki od korzenia do liścia v (mierzoną liczbą krawędzi na ścieżce).

Rozważmy następujący problem. Dany jest n-elementowy zbiór $\{w_1, \ldots, w_n\}$ dodatnich liczb rzeczywistych. Zadaniem jest znalezienie ważonego drzewa binarnego T o n liściach, takiego, że każda liczba w_i jest wagą dokładnie jednego liścia oraz T ma minimalną wagę $EL(T)$ pośród wszystkich drzew o tej własności.

ZADANIA DODATKOWE - DO SAMODZIELNEGO ROZWI[1]*zywania*

1. (1pkt) Wykaż, że zachłanny algorytm wydawania reszty, w sytuacji gdy monety mają nominały $c^0, c^1, \ldots, c^k$ dla pewnych stałych naturalnych $c > 1$ i $k \geq 1$, daje optymalne rozwiązanie.

2. (1pkt) Udowodnij, że spójny graf, którego wszystkie krawędzie mają różne wagi, posiada dokładnie jedno minimalne drzewo spinające.

3. (1pkt) Udowodnij poprawność algorytmu Prima.

4. (2pkt) Na wykładzie przedstawiono zachłanny algorytm dla następującego problemu *Pokrycia zbioru*:

Dane:	rodzina $S = \{S_1, \ldots, S_k\}$ podzbiorów zbioru $\{1, \ldots, n\}$ oraz funkcja $c : S \to \mathcal{R}_+$.
Rozwiązanie:	podrodzina $S' \subseteq S$, taka, że $\cup_{T \in S'} T = \{1, \ldots, n\}$.
Cel:	Znaleźć rozwiązanie optymalne, tj. minimalnym koszcie, zdefiniowanym jako $c(S') = \sum_{T \in S'} c(T)$.

   Przedstawiona strategia konstruuje rozwiązanie startując od $S' = \emptyset$. W kolejnej iteracji do S' dodawany jest ten z podzbiorów S_i, dla którego wartość wyrażenia $c(S_i)/|S_i \setminus \cup_{T \in S'} T|$ jest minimalna. Postępujemy tak do czasu aż $\cup_{T \in S'} T = \{1, \ldots, n\}$ (ten moment nastąpi, ponieważ zakładamy, że $\cup_{T \in S} T = \{1, \ldots, n\}$.

   - Udowodnij, że powyższy algorytm daje rozwiązania, które są co najwyżej $\log n$ razy gorsze od rozwiązania optymalnego.
   - Pokaż, że istnieją dane, dla których rozwiązania znajdowane przez powyższy algorytm są $\Omega(\log n)$ gorsze od rozwiązań optymalnych.

5. (1pkt) Ułóż algorytm rozwiązujący następujący problem szeregowania zadań dla c procesorów:

Dane:	t_i - czas obsługi i-tego zadania $(i = 1, 2, ..., n)$.
Problem:	każde z zadań przypisać do jednego z c procesorów oraz ustalić kolejność wykonywania zadań przez każdy z procesorów tak, by zminimalizować wartość: $T = \sum_{i=1}^{n}$ (czas przebywania i-tego zadania w systemie).

   Udowodnij, że Twój algorytm zawsze znajduje optymalne rozwiązanie.

   Jaki jest czas działania Twojego algorytmu?

6. (1pkt) Ułóż algorytm rozwiązujący następujący problem szeregowania zadań dla c procesorów:

Dane:	t_i - czas obsługi i-tego zadania $(i = 1, 2, ..., n)$.
Problem:	każde z zadań przypisać do jednego z c procesorów oraz ustalić kolejność wykonywania zadań przez każdy z procesorów tak, by zminimalizować wartość: $T = \sum_{i=1}^{n}$ (czas przebywania i-tego zadania w systemie).

   Udowodnij, że Twój algorytm zawsze znajduje optymalne rozwiązanie.

   Jaki jest czas działania Twojego algorytmu?

7. (2pkt) Rozważamy grafy skierowane, w których każda para wierzchołków połączona jest przynajmniej jedną krawędzią. Podaj algorytm wyznaczający dla takiego grafu *ścieżkę Hamiltona*, tj. ścieżkę przechodzącą dokładnie jeden raz przez wszystkie wierzchołki. Udowodnij, że Twój algorytm działa poprawnie.

8. (2pkt) Udowodnij, że w grafie nieskierowanym o n wierzchołkach, w którym każdy wierzchołek ma co najmniej $n/2$ sąsiadów, istnieje ścieżka Hamiltona. Podaj algorytm, który dla takich grafów znajduje tę ścieżkę.

9. (1pkt) Przypomnij sobie algorytm Dijkstry znajdowania najkrótszych ścieżek od zadanego wierzchołka do wszystkich pozostałych wierzchołków. Udowodnij jego poprawność.

10. (2pkt) Zaproponuj implementację algorytmu Boruvki. Jaka jest jej złożoność czasowa?

Problemy (chyba) trudne

1. Podaj warunek konieczny i wystarczający na to, by dla danego ciągu nominałów monet podany na wykładzie algorytm zachłanny wydawania reszty był poprawny.

2. Jaka jest złożoność następującego problemu?
   PROBLEM.
   dane: ciąg $C = c_1, \ldots, c_k$
   wynik: "TAK" - jeśli zachłanny algorytm wydawania reszty dla ciągu C i dowolnej wartości reszty daje optymalne rozwiązanie;
   "NIE" - w przeciwnym przypadku.

Krzysztof Loryś