

ALGORYTMY I STRUKTURY DANYCH

IIUWr. II rok informatyki.

1. (1pkt) Zmień podany na wykładzie algorytm znajdujący najtańszą drogę przejścia przez tablicę tak, by znajdował drogę o drugim co do wielkości koszcie.
2. (1pkt) Uzupełnij podany na wykładzie algorytm sprawdzający przynależność słowa do języka generowanego przez bezkontekstową gramatykę w normalnej postaci Chomsky'ego tak, by w przypadku pozytywnej odpowiedzi wypisywał jego wyprowadzenie.
3. (1,5pkt) *Superciągiem* ciągów X i Y nazywamy każdy taki ciąg Z , że zarówno X jak i Y są podciągami ciągu Z .
Ułóż algorytm, który dla danych ciągów X i Y znajduje ich najkrótszy superciąg.
4. (2pkt) W n -elementowej tablicy A pamiętany jest rosnący ciąg liczb naturalnych. Nie znamy wartości jej elementów, ale możemy się o nie pytać. Pytanie o wartość $A[i]$ kosztuje nas c_i .
Ułóż algorytm, który dla danych liczb naturalnych c_1, c_2, \dots, c_n oraz liczby k obliczy najmniejszym kosztem (liczonym jako suma kosztów zadanych pytań), ile liczb w tablicy A ma wartość większą niż k .
5. (2pkt) Zmodyfikuj algorytm znajdujący najdłuższy wspólny podciąg dwóch ciągów n elementowych, tak by działał w czasie $O(n^2)$ i używał $O(n)$ pamięci.
6. (1,5pkt) Ułóż algorytm, który dla danego drzewa ważonego $T = (V, E; c)$, gdzie $c : V \rightarrow \mathbb{N}$ jest funkcją wagową, znajduje niezależny podzbiór $V' \subseteq V$, którego suma wag wierzchołków jest możliwie największa.
7. (2pkt) Ułóż algorytmy, które dla danych podciągów x i y rozwiązują następujące wersje problemu znajdowania najdłuższego wspólnego podciągu:
 - znajdowanie najdłuższego wspólnego podciągu zawierającego podciąg "aaabb",
 - znajdowanie najdłuższego wspólnego podciągu nie zawierającego podciągu "aaabb",
 - znajdowanie najdłuższego wspólnego podciągu zawierającego podsłowo "aaabb",
 - znajdowanie najdłuższego wspólnego podciągu nie zawierającego podsłowa "aaabb".
8. (1pkt) Rozważmy następujący problem *3-podziału*. Dla danych liczb całkowitych $a_1, \dots, a_n \in \langle -C..C \rangle$ chcemy stwierdzić, czy można podzielić zbiór $\{1, 2, \dots, n\}$ na trzy rozłączne podzbiory I, J, K , takie, że
$$\sum_{i \in I} a_i = \sum_{j \in J} a_j = \sum_{k \in K} a_k.$$
9. (2pkt) Ułóż algorytm, który dla dwóch ciągów liczb naturalnych X i Y , znajduje ich najdłuższy wspólny podciąg, który jest ciągiem rosnącym. Czy założenie, że liczby w ciągach są ograniczone przez pewną stałą k , upraszcza problem?
10. (2pkt) Dwie proste równoległe l' i l'' przecięto n prostymi p_1, \dots, p_n . Punkty przecięcia prostej p_i z prostymi l' i l'' wyznaczają na niej odcinek. Niech Odc będzie zbiorem tych odcinków.
 - (a) Ułóż algorytm, wyznaczający w Odc podzbiór nieprzecinających się odcinków, o największej mocy.
 - (b) Ułóż algorytm, wyznaczający liczbę podzbiorów, o których mowa w poprzednim punkcie.