

ALGORYTMY I STRUKTURY DANYCH

STRATEGIA ZACHŁANNA

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. [2 pkt.] Wykaż, że strategia zachłanna zawsze pozwala znaleźć optymalne rozwiązanie dla ciągłego problemu plecakowego.
2. [2 pkt.] Rozważmy dyskretny problem plecakowy, w którym kolejność przedmiotów uporządkowanych rosnąco według ciężaru jest taka sama, jak przy uporządkowaniu malejącym według wartości. Podaj efektywny algorytm znajdujący optymalne rozwiązanie dla takiej wersji problemu plecakowego i uzasadnij jego poprawność.
3. [1 pkt.] Dany jest n -elementowy zbiór znaków wraz z częstościami ich występowania. Udowodnij, że jeśli znaki są uporządkowane niemalejąco względem częstości, to długości odpowiadających im słów kodowych wygenerowanych algorytmem Huffinana tworzą ciąg nierosnący.
4. [3 pkt.] Dana jest lista n wykładów, które mają się odbywać w tej samej sali. Każdy z wykładowców upiera się przy swoim terminie przeprowadzenia zajęć: $[p_1, k_1), [p_2, k_2), \dots, [p_n, k_n)$. Nie można wszystkich usatysfakcjonować, gdyż dysponujemy tylko jedną salą wykładową. Podaj algorytm, który wybierze wykłady w taki sposób, aby odbyło się ich jak najwięcej. Czy jest on również skuteczny, gdy chcemy zmaksymalizować czas wykorzystania sali?
5. [2 pkt.] Danych jest n zadań o czasach wykonania odpowiednio $t_1, t_2, \dots, t_n$ oraz system komputerowy z jednym procesorem. Zadania te ustawione w kolejkę będą po kolei wykonywane. W jakiej kolejności należy je wykonać, aby zminimalizować czas ich przebywania w systemie? Podaj algorytm rozwiązujący ten problem i udowodnij, że znajduje on optymalne rozwiązanie.
6. [2 pkt.] Danych jest n zadań o czasach wykonania odpowiednio $t_1, t_2, \dots, t_n$ oraz system komputerowy z c procesorami. Zadania te porozdzielane do poszczególnych procesorów i ustawione w kolejkach będą po kolei wykonywane. Jak porozdzielać je do poszczególnych procesorów, aby zminimalizować czas ich przebywania w systemie? Podaj algorytm rozwiązujący ten problem i udowodnij, że znajduje on optymalne rozwiązanie.