Wojciech "Sushi" Jasiński Anna Karaś Konrad Kleczkowski Krzysztof Łyskawa
Mikołaj Motak Grzegorz Winiarski

Algorytmy i Struktury Danych

egzamin poprawkowy, cz. I

wrzesień 2018

Zadanie 1. W algorytmie znajdowania mediany w zbiorze S poprzez próbkowanie w jednym z pierwszych kroków wybierany jest losowy podzbiór $R \subseteq S$. Ile maksymalnie procent elementów zbioru S może znaleźć się w zbiorze R?

Zadanie 2. Jaką złożoność ma uogólnienie algorytmu Karatsuby, w którym mnożone liczby dzielone są na trzy części? Odpowiedź uzasadnij.

Zadanie 3. Sformułuj zasadę zero-jedynkową (albo inaczej lemat zero-jedynkowy) dla sieci sortujących.

Zadanie 4. Podaj jaki jest pesymistyczny czas wykonywania operacji `find`, gdy operacja `union` wykonywana jest w sposób:

(a) zbalansowany,
(b) niezbalansowany.

Zadanie 5. Opisz (albo zapisz w pseudokodzie), w jaki sposób wykonywana jest operacja znajdowania minimum w drzewie van Emde Boasa. Jak kosztowna jest ta operacja?

Zadanie 6. Czy usunięcie wierzchołka z drzewca o n wierzchołkach może wymagać $\Omega(n)$ operacji rotacji? Jeśli tak, narysuj taki drzewiec i wskaż w nim ten usuwany wierzchołek. Jeśli nie, przedstaw uzasadnienie.

Zadanie 7. Opisz, w jaki sposób sieci sortujące ciągi bitoniczne są użyte w konstrukcji sieci scalającej dwa uporządkowane ciągi.

Zadanie 8. Podaj optymalny (pod względem liczby wykonywanych porównań) algorytm jednoczesnego znajdowania minimum i maksimum w ciągu. Ile porównań wykonuje ten algorytm?

Zadanie 9. Opisz, w jaki sposób wykonywana jest operacja usuwania klucza ze słownika zapamiętywanego w tablicy haszującej z adresowaniem otwartym, w której kolizje rozwiązywane są za pomocą metody podwójnego haszowania.

Zadanie 10. Podaj definicję problemu plecakowego bez powtórzeń i przedstaw pseudowielomianowy algorytm rozwiązujący ten problem. Uzasadnij stwierdzenie, że jest on pseudowielomianowy.

Zadanie 11. Rozwiąż poniższe równanie rekurencyjne:

$$T(n) = \begin{cases} 1 & \text{dla } n = 2 \\ 4T(\sqrt{n}) + \mathcal{O}(1) & \text{dla } n > 1 \end{cases}$$

Zadanie 12. Ile wyniesie pesymistyczna złożoność algorytmu magicznych piątek, jeśli ciąg będziemy dzielić na podciągi trójelementowe zamiast na podciągi pięcioelementowe? Czy w optymistycznym przypadku algorytm będzie działał tak samo (asymptotycznie) jak w przypadku pesymistycznym?

Zadanie 13. Zapisz w pseudokodzie algorytm Knutha-Morrisa-Prata. Załóż, że masz już wyliczoną funkcję π.

Zadanie 14. Podany na wykładzie algorytm obliczający minimalny koszt policzenia iloczynu n macierzy, o wymiarach odpowiednio $d_0 \times d_1, d_1 \times d_2, \ldots, d_{n-1} \times d_n$, wylicza pewne wartości $m_{i,j}$. Podaj ich definicje i sposób wyliczenia. Jeśli nie pamiętasz tego algorytmu, podaj inny algorytm o nie gorszej złożoności, który rozwiązuje ten problem.

Zadanie 15. Algorytm Karpa-Millera-Rosenberga (KMR) nadaje podsłowom tekstu i wzorca numery, by wyszukiwanie wzorca w tekście sprowadzić do sprawdzenia, czy w tekście występuje m-literowe podsłowo o numerze równym numerowi wzorca. Opisz, w jaki sposób algorytm KMR przypisuje numery podsłowom trzydziestoliterowym.

Zadanie 16. Chcemy dla danego grafu nieskierowanego (bez wag na krawędziach) o n wierzchołkach i m krawędziach obliczyć sumę odległości wszystkich wierzchołków od wyróżnionego wierzchołka (powiedzmy od wierzchołka o numerze 0). Opisz ideę optymalnego algorytmu rozwiązującego ten problem. Uzasadnij, że jest on optymalny.

Uwaga 16.1. Graf może mieć cykle. Jak nie umiesz rozwiązać dla grafu z cyklami, to będzie taka mniejsza bombka.

Uwaga 16.2. Tyle że nie ma już podziału na małe i duże bombki.

Zadanie 17. Porównaj następujące klasy funkcji:

(a) $\mathcal{O}((\log^* n)^{\log n})$,
(b) $\mathcal{O}((\log n)^{\log^* n})$,
(c) $\mathcal{O}(\log n)$.

Uwaga 17.1. Muszą być obliczenia i rozumowanie, nie może być wprost z definicji.

Zadanie 18. Opisz ideę algorytmu Karpa-Rabina wyszukiwania wzorca w tekście.

Zadanie 19. Wyjaśnij, na czym polega operacja kaskadowego odcinania w kopcach Fibonacciego.

Zadanie 20. Podaj pseudokod optymalnego algorytmu tworzenia kopca (binarnego).