Algorytmy probabilistyczne

Lista zadań nr 2

1. Rozważmy następujące decyzyjne wersje problemu *Min-Cut*:

   (A) Dla danego grafu G i stałej K sprawdź czy rozmiar *min-cut(G)* jest równy K.

   (B) Dla danego grafu G i stałej K sprawdź czy rozmiar *min-cut(G)* jest mniejszy od K.

   Pokazać, że problem (A) należy do $\mathcal{BPP}$ a problem (B) do $\mathcal{RP}$.

2. (a) Podać metodę generowania losowych permutacji n liczb z rozkładem jednostajnym. Twój algorytm ma dostęp do źródła niezależnych bitów losowych generowanych z rozkładem jednostajnym. Efektywność algorytmu mierzona jest zarówno czasem obliczeń jak i liczbą bitów losowych. Jakie dolne granice potrafisz udowodnić dla tego problemu?

   (b) Rozważmy następującą metodę generowania losowych permutacji n-elementowych: losujemy niezależnie, według tego samego rozkładu (np. jednostajnego w przedziale $\langle 0, 1\rangle$), n liczb $X_1, \ldots, X_n$. Twierdzimy, że permutacja porządkująca te liczby niemalejąco jest permutacją losową. Czy twierdzenie to jest prawdziwe? Jak efektywna jest ta metoda?

   (c) Rozważmy następującą implementację metody zasugerowanej w (b): Binarną reprezentacją ułamka X_i jest ciąg niezależnych bitów losowych. W dowolnej fazie algorytmu sortującego znamy tylko tyle bitów każdego X_i, ile potrzeba do ustalenia wyników wszystkich, wykonanych do tej pory, porównań. Jeśli podczas porównania X_i i X_j, bieżące prefiksy ich rozwinięć binarnych nie określają wyniku porównania, to rozszerzamy te prefiksy przez wylosowanie kolejnych bitów. Obliczyć ścisłe granice na oczekiwaną liczbę bitów losowych użytych przez tę implementację.

3. Rozważmy ciąg n rzutów monetą. Niech H_i oznacza wartość bezwzględną różnicy liczby orłów i reszek po i rzutach. Zdefiniujmy $H = \max_i H_i$. Na wykładzie udowodniono, że $\mathbf{E}(H_i) = \Theta(\sqrt{i})$. Udowodnić, że $\mathbf{E}(H) = \Theta(\sqrt{n})$?

4. Załóżmy, że ze zbioru wszystkich permutacji n-elementowych wybieramy losowo z rozkładem jednostajnym permutację π. Niech $L(\pi)$ oznacza najdłuższy podciąg rosnący w permutacji π. Udowodnij, że $\mathbf{E}(L(n))$ jest $\Omega(\sqrt{n})$. Podaj najlepsze możliwe oszacowanie dolne na stałą ukrytą pod Ω.

5. Rozważmy następujący algorytm znajdowania k-tego elementu w n-elementowym zbiorze S: Wybieramy w S losowy element $M(S)$ i dzielimy S na dwa podzbiory S_1 i S_2, elementów odpowiednio mniejszych i większych od $M(S)$. Jeśli $|S_1| = k-1$ to $M(S)$ jest szukanym elementem i algorytm zatrzymuje się. Jeśli $|S_1| \geq k$, to rekurencyjnie szukamy k-tego elementu w zbiorze S_1; w przeciwnym przypadku rekurencyjnie szukamy $k-1-|S_1|$-ego elementu w zbiorze S_2.Pokaż, że algorytm znajduje k-ty element zbioru S w oczekiwanym czasie $O(n)$.

6 marca 2019 Marek Piotrów