Algorytmy i struktury danych
Lista 3

Zadanie 1.
Podaj algorytm scalający k posortowanych list tak aby powstała jedna posortowana lista nb (liczba wszystkich elementów na listach to n) działający w czasie $O(n \log k)$.

Zadanie 2.
Zdefiniujmy algorytm k-MergeSort jako uogólnienie algorytmu sortowania przez scalanie. Różni się od omawianego na wykładzie algorytmu sortowania przez scalanie tym, że dzieli sortowana tablice rekurencyjnie na k równych części (zakładamy, że liczba elementów w tablicy jest potęgą k ($n = k^l$)). Używając wyniku z zadania 1 proszę wykazać dla jakiego k algorytm ma najmniejsza asymptotyczną złożoność obliczeniową liczby porównań (górne ograniczenie $O()$).

Zadanie 3.
Załóżmy że tablica $A = [a_1, \dots, a_n]$ jest do pewnego momentu k posortowana malejąco i dalej rosnąco (tzn. dla $\forall i \leq k \quad a_i > a_{i+1}$ oraz dla $\forall i \geq k \quad a_i < a_{i+1}$). Zaprojektuj algorytm znajdujący minimalny element w tablicy A, którego złożoność obliczeniowa będzie wynosić $O(\log n)$. Udowodnij poprawność działania zaproponowanego algorytmu.

Zadanie 4.
Zdefiniujmy liczbę

$$G_n = \begin{cases} 0 & \text{if} \quad n = 0 \\ 1 & \text{if} \quad n = 1 \vee n = 2 \\ G_{n-1} + G_{n-2} + G_{n-3} & \text{if} \quad n \geq 3 \end{cases}$$

Zaprojektuj macierzowy algorytm wyliczania liczby G_n w czasie $O(\log n)$. Udowodnij poprawność działania zaproponowanego algorytmu.

Zadanie 5.
Pokaż procedurę `Partition` dla której czas działania algorytmu Quicksort jest $\Theta(n \lg n)$ gdy wszystkie elementy sortowanej tablicy są takie same.

Zadanie 6.
Załóżmy, że procedura Partition dzieli tablice zawsze w proporcjach $1 - \alpha$ do α, gdzie $0 \leq \alpha \leq \frac{1}{2}$ jest stałą. Pokaż, że minimalna głębokość liścia w drzewie rekursji wynosi $-\frac{\lg n}{\lg \alpha}$, a maksymalna głębokość liścia wynosi $-\frac{\lg n}{\lg(1-\alpha)}$. Odpowiedź uzasadnij.

Zadanie 7.
Wykaż, że nie istnieje algorytm sortujący, który działa w czasie liniowym dla co najmniej połowy z $n!$ możliwych danych wejściowych długości n. Czy odpowiedź ulegnie zmianie jeśli zapytamy o ułamek $\frac{1}{n}$ lub $\frac{1}{2^n}$ wszystkich permutacji?

Zadanie 8.
Zaprojektuj algorytm, który sortuje n liczb całkowitych z przedziału od 1 do n^2 w czasie $O(n)$.