Algorytmy i Struktury Danych

Lista zadań 13 - DFT, sieci sortujące

1. Dyskretną transformatą Fouriera ciągu $(a_1, ..., a_n)$ nazywamy ciag $(A_1, ..., A_n)$ taki, że:

$$A_k = \sum_{p=0}^{n-1} a_p e^{2\pi i k p / n} \qquad \text{dla } k = 0 \dots n-1. \tag{1}$$

   Odwrotną dyskretną transformatą Fouriera ciągu $(A_1, \dots, A_n)$ nazywamy ciąg $(b_1, ..., b_n)$ taki, że:

$$b_q = \frac{1}{n} \sum_{k=0}^{n-1} A_k e^{-2\pi i q k / n} \qquad \text{dla } q = 0 \dots n-1 \tag{2}$$

   Wstawiając wzór (1) do prawej strony wzoru (2) udowodnij że $b_q = a_q$ dla $q = 0 \dots n-1$, czyli, że wynikiem odwrotnej dyskretnej transformaty Fouriera, wykonanej na transformacie ciągu $(a_1, \dots, a_n)$ jest faktycznie wyjściowy ciąg liczb $(a_1, \dots, a_n)$.

2. Wyznacz DFT dla ciągów (5, 3), (1, 5, 3, 1) oraz (1,2,3,4,5,6,7,8):
   (a) z definicji,
   (b) symulując działanie algorytmu FFT podanego na wykładzie.

3. Udowodnij zasadę zero-jedynkową dla sieci sortujących: *Jeśli sieć poprawnie sortuje wszystkie możliwe n-elementowe ciągi zer i jedynek, to dobrze sortuje dowolne ciągi liczb rzeczywistych.*

4. * Udowodnij, że `bitonic_half_cleaner(2n)` (patrz Cormen) działa poprawnie dla dowolnego ciągu bitonicznego złożonego z zer i jedynek. To znaczy, że dla dowolnego bitonicznego ciągu zer i jedynek o długości $2n$ danego na wejściu, wynikiem jest: albo ciąg którego lewa połowa to zera a prawa jest bitoniczna, albo ciąg którego lewa połowa jest bitoniczna a prawa to jedynki.

5. Narysuj sieć sortującą n liczb dla $n = 2, 4, 8, 16$. Powinna to być opisana na wykładzie sieć implementująca równoległą wersję algorytmu `mergesort`, działająca w czasie $O((\log n)^2)$. Prześledź działanie sieci o $n = 8$ dla ciągu wejściowego: 8 4 2 3 7 5 6 1, rysując jakie liczby wchodzą i wychodzą z każdego komparatora (na kartkach).