MATEMATYKA DYSKRETNA - wykład 5

dr inż Krzysztof Bryś

Algorytmy generujące podstawowe obiekty kombinatoryczne

1. Algorytm generujący listę wszystkich permutacji zbioru $[n]$

Krok 1.

$$L_1 := (1)$$

Dla każdego $k = 2, \ldots, n$ wykonaj krok k:

Krok k. Załóżmy, że mamy wygenerowaną listę wszystkich permutacji zbioru $[k-1]$:

$$L_{k-1} = (p_1, p_2, \ldots, p_{(k-1)!})$$

Niech $p_{i,j}$ będzie permutacją zbioru $[k]$ powstałą z permutacji p_i zbioru $[k-1]$ przez dodanie k na j-tej pozycji. Wtedy

$$L_k := (p_{1,1}, p_{1,2}, \ldots, p_{1,k}, p_{2,1}, p_{2,2}, \ldots, p_{2,k}, \ldots, p_{(k-1)!,1}, p_{(k-1)!,2}, \ldots, p_{(k-1)!,k})$$

2. Algorytm generujący wszystkie k-elementowe podzbiory (kombinacje) zbioru $[n]$

Niech

$G(n, k)$ będzie listą wszystkich k-elementowych podzbiorów (kombinacji) zbioru $[n]$ oraz

$G(n-1, k-1) \cup \{n\}$ będzie listą powstałą z $G(n-1, k-1)$ przez dodanie liczby n do każdego zbioru na liście $G(n-1, k-1)$.

Wtedy algorytm generuje listę $G(n, k)$ w następujący sposób:

$$G(i, 1) = (\{1\}, \{2\}, \ldots, \{i\}) \text{ dla każdego } i = 1, \ldots, n$$

$$G(i, i) = ([i]) \text{ dla każdego } i = 1, \ldots, n$$

$$G(n, k) = (G(n-1, k), G(n-1, k-1) \cup \{n\}) \text{ jeśli } 1 < k < n$$

3. Algorytm generujący wszystkie podziały zbioru $[n]$

Krok 1.

$$L_1 := (1)$$

Dla każdego $k = 2, \ldots, n$ wykonaj krok k:

Krok k. Załóżmy, że mamy wygenerowaną listę L_{k-1} wszystkich podziałów zbioru $[k-1]$. Listę L_k tworzymy w następujący sposób:

Każdy podział $\{B_1, B_2, \ldots, B_m\}$ zbioru $[k-1]$ występujący na liście L_{k-1} zastępujemy następującymi podziałami zbioru $[k]$:

$$\{B_1 \cup \{k\}, B_2, \ldots, B_m\}$$

$$\{B_1, B_2 \cup \{k\}, \ldots, B_m\}$$

$$\ldots$$

$$\{B_1, B_2, \ldots, B_m \cup \{k\}\}$$

$$\{B_1, B_2, \ldots, B_m, \{k\}\}$$

(Nowy element k jest dołączany w kolejnych podziałach do kolejnych bloków oryginalnego podziału a ostatni podział powstaje przez dołączenie do podziału zbioru $[k-1]$ nowego bloku zawierającego jedynie element k.)