Генерация всех подмножеств

$A = \{a_1, a_2, \ldots a_n\}$ – множество элементов

$B = \{b_1, b_2, \ldots, b_n\}$ – булев вектор, описывающий подмножество

$b_i = 1$, если $a_i \in$ подмножеству и $= 0$ иначе

$x = (b_n b_{n-1} \ldots b_1)_2$ – число $\in [0..2^n - 1]$

Алгоритм 1 (перебора всех подмножеств) :

$for\ x := 0\ to\ 2^n - 1\ do$

$begin$

$b =$ двоичная запись$(x)$; $\sim \Theta(n)$

$DoSmth(b)$;

$end$; $\sim \Theta(2^n n)$

Рассмотренный порядок перебора - лексикографический
000, 001, 010, 011, 100, 101, 110, 111

Алгоритм 2 (эффективный, лексикографический порядок)

$B = (b_1, b_2, \ldots, b_n, b_{n+1})$ – массив

$for\ i := 1\ to\ n+1\ do\ B[i] := 0;$

$while\ B[n+1] = 0\ do$

$begin$

$DoSmth(B[n], \ldots, B[1])$; $- 2^n$ раз

$j := 1$;

$while\ B[j] = 1\ do\ begin$

$B[j] = 0;\ j := j + 1\ end;$

Всего $2 + 4 + 8 + \ldots + 2^n = 2^{n+1} - 2 \sim \Theta(2^n)$

$B[j] := 1$

$end$; $t_2(n) \sim \Theta(2^n)$

Коды Грея - это упорядоченная последовательность двоичных векторов, в которой соседние вектора отличаются ровно в 1 разряде.
Пример: 000, 001, 011, 010, 110, 111, 101, 100

Пусть $G(n) = G_0, \ldots, G_{2^n - 1}$ – код Грея длины $n$

$G_0, G_1, \ldots$ – двоичные вектора

$T_n = t_1, t_2, \ldots, t_{2^n - 1}$, где $t_i$ – номер изменяемого разряда (считая справа налево) при переходе от $G_{i-1}$ к $G_i$

Двоично – отраженный код грея :

$G(1) = 0,1$

Если $G(n) = G_0, G_1, \ldots, G_{2^n - 1}$

то $G(n+1) = 0G_0, 0G_1, \ldots, 0G_{2^n - 1}, 1G_{2^n - 1}, \ldots, 1G_1, 1G_0$

$T_1 = 1$ $\quad T_{n+1} = T_n, (n+1), T_n$

Пример $\quad n = 3$

$G(1) = 0,1 \quad G(2) = 00,01,11,10$

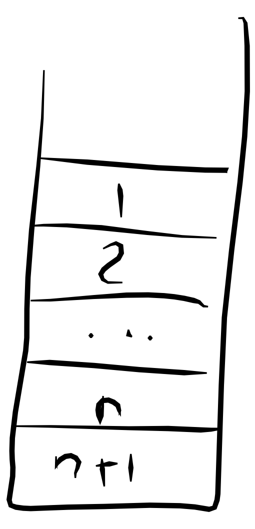
$G(3) = 000,001,011,010,110,111,101,100$

$T_3 = 1,2,1,3,1,2,1$

Далее будем говорить о генерации посл – ти $T_n$, т.к. по ней однозначно строится код Грея

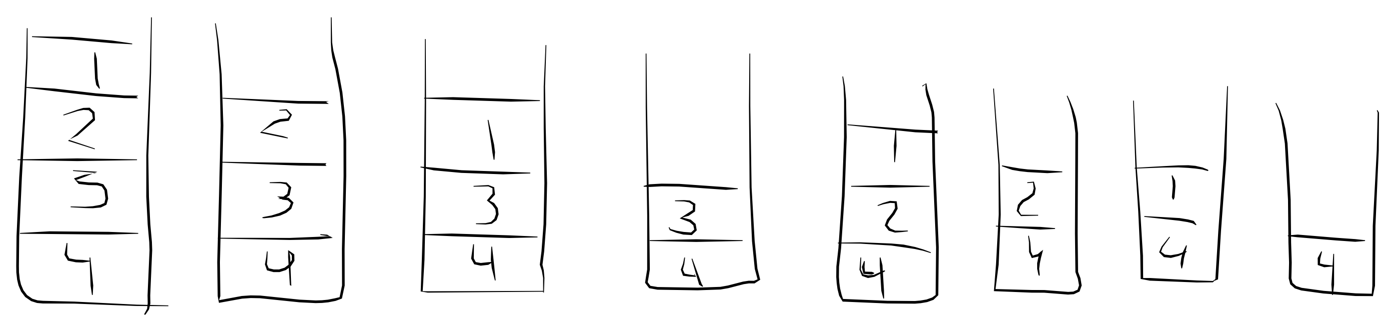
Рассмотрим стек $A$

Начальное заполнение : $n+1, n, n-1, \ldots, 2, 1$




Пусть на шаге k на вершине стека лежит $a_k$

$t_k = a_k$ $\quad$ если $a_k > 1$, то после извлечения добавим в стек числа $(a_k - 1), (a_k - 2), \ldots, 1$



1, 2, 1, 3, 1, 2, 1

Рассмотрим массив $B = (b_0, b_1, \ldots, b_n)$

$B^k$ – состояние массива на шаге $k$

$b_0^k = a_k$

$b_j^k =$ элементу стека $A$, лежащего под числом $j$ (если $j$ есть в $A$)

$b_j^k = j + 1$, если $j$ нет в $A$

$$b_0^{k+1} = \begin{cases} b_1^k, \text{если извлекли 1} \\ 1 \text{ иначе} \end{cases}$$

$$b_j^{k+1} = \begin{cases} j+1, \text{ если } j = 1 \text{ и извлекли 1} \\ b_j^k, \text{если } j > 1 \text{ и извлекли 1} \\ b_j^k, \text{если } j > \text{извлеченного числа и извлекли не 1} \\ j+1, \text{ если } j = \text{извлеченному числу и извлекли не 1} \\ b_{j+1}^k, \text{ если } j = (\text{извлеченное} - 1) \text{ и извлекли не 1} \\ j + 1 = b_j^k, \text{ если } j < (\text{извлеченное} - 1) \text{ и извлекли не 1} \end{cases}$$

Алгоритм 3 (эффективный, коды Грея)

$for\ i := 0\ to\ n\ do\ b[i] := i + 1$

$x := 0$

$while\ x < n + 1\ do$

$begin$

$x := b[0]$;

$ChangeBit(x)$;

$b[0] := 1$;

$b[x-1] := b[x]$;

$b[x] := x + 1$;

$end$