**УДК 62.507**

**В.В. Мінзюк**
Національний університет “Львівська політехніка”,
кафедра радіоелектронних пристроїв та систем

# СПОСІБ СОРТУВАННЯ ЦІЛИХ ЧИСЕЛ ДЛЯ ЗАДАЧ МІНІМІЗАЦІЇ БУЛЬОВИХ ФУНКЦІЙ

© *Мінзюк В.В., 2011*

**Запропоновано модифікацію методу низхідного побітового сортування цілих чисел. Метод доповнено процедурою підрахунку потужності підмножин, одержаних на етапах сортування, для виявлення таких підмножин, що можуть бути представлені у вигляді кон’юнктерма із поглинутими молодшими розрядами.**

**Ключові слова: сортування, мінімізація, бульова функція, кон’юнктерм.**

**In this paper modification of most significant digit binary-radix sort has been considered. The method is supplemented by procedure of cardinality counting for subsets, wich are derived on steps of sorting. This procedure helps to detect such sets wich can be represented by conjuncterms with absorbed low-order bits.**

**Key words: sort, minimization, boolean function, conjuncterm.**

## Вступ

У багатьох задачах опрацювання даних виникає потреба упорядкування цих даних за певною ознакою. Як правило розглядають скінченні множини елементів даних. За означенням множину *A* називають скінченною, якщо вона порожня чи існує таке натуральне *m*, що множина *A* може бути взаємно однозначно відображена на підмножину $N_m \subset \mathbf{N}$: $N_m = \{x | x \in \mathbf{N}, x \le m\}$, де **N** – множина усіх натуральних чисел. Інакше кажучи, множина *A* може бути занумерована не більше ніж *m* числами. Розглядатимемо випадок, коли упорядкованою вважається лише одна послідовність елементів множини *A*, причому *A* не є мультимножиною (тобто не містить тотожних елементів). Скінченну послідовність $A = \langle a_1, a_2, ..., a_m \rangle$ можна трактувати як взаємно однозначне відображення (функцію) $f_A: \{1,2,...,m\} \to \{a_1, a_2, ..., a_m\}$ так, що $f_A(i) = a_i$ (*i*=1,2,…,*m*).

Аналогічно, множину *B*, що є послідовністю *m* невід’ємних цілих чисел із їх природним порядком можна подати взаємно однозначним відображенням $f_B: \{1,2,...,m\} \to \{b_1, b_2, ..., b_m\}$ так, що $f_B(i) = b_i$ (*i*=1,2,…,*m*). Оскільки $f_A$ та $f_B$ є взаємно однозначними відображеннями $N_m \to A$ та $N_m \to B$ відповідно, то відображення $B \to A$ є взаємно однозначним і становить суперпозицію відображень $f_B^{-1} \circ f_A$, де $f_B^{-1}$ – відображення обернене до $f_B$. Отже, означена вище задача упорядкування даних набуває вигляду сортування лінійного списку невід’ємних цілих чисел.

Такий підхід має особливе практичне значення для обчислювальної техніки. Зокрема, у задачі мінімізації бульових функцій, що є однією з основних складових логікового синтезу комбінаційних мереж, виникає потреба упорядкування лінійного списку мінтермів [1].

З іншого боку, цілі невід’ємні числа можна трактувати, як числовий запис вершин логікового простору (мінтермів), та використати для їх упорядкування математичний апарат алгебри логіки.

## Постановка задачі

Нехай задано список цілих невід’ємних чисел $B = \langle k_0, k_1, ... k_{m-1} \rangle$, у якому немає тотожних елементів. Необхідно переставити елементи списку *B* так, щоб дістати упорядкований список $B' = \langle k_{i_0}, k_{i_1}, ... k_{i_{m-1}} \rangle$, де для будь-якого цілого числа *g* від 0 до *m*-1 ($0 \le g \le (m-1)$) справджується умова $k_{i_g} < k_{i_{g+1}}$.



Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua

Такі відомі методи, як обмінне сортування, сортування вставкою, сортування за допомогою вибору мають простий алгоритм та не вимагають додаткової пам’яті, але є повільними. Більшу швидкодію мають алгоритми сортування квадратичним вибором, швидке сортування, сортування злиттям та розподільне сортування, але вони вимагають додаткової пам’яті.

У цій роботі запропоновано модифікацію методу низхідного побітового сортування, що є різновидом порозрядного сортування. Нехай елементи лінійного списку $B$ є $n$-цифрові невід’ємні цілі двійкові числа, $d(j, k)$ – цифра у $j$-му розряді двійкового числа $k$. Необхідно упорядкувати масив із $m$ елементів.

З точки зору алгебри логіки заданий список $B$ можна вважати двійковим зображенням бульової функції від $n$ змінних, а елементи списку $B$ — зображенням мінтермів цієї бульової функції, де кожен двійковий розряд є зображенням відповідної бульової змінної.

Починаємо розподіл зі старшого розряду. Крім того, одночасно із сортуванням підраховуємо кількості елементів.

## Побітове сортування зі склеюванням

Для кожного двійкового розряду (починаючи зі старшого) здійснюємо процедуру сортування множини елементів на дві підмножини. У першу входять усі елементи множини, що містять нуль у заданому розряді, в іншу — ті, що містять одиницю. Одержані підмножини розглядатимуться окремо під час сортування за наступним розрядом.

Одночасно із сортуванням підраховуємо елементи у кожній підмножині (тобто визначаємо потужність підмножини). Якщо потужність підмножини становить $2^j$, де $j$ – номер розряду, за яким відбувається сортування, то усі кон’юнктерми цієї підмножини склеюються по молодших від $j$ розрядах. Тоді одержану підмножину можна записати у вигляді кон’юнктерма, у якого усі розряди, молодші від $j$, є поглинуті. Такі кон’юнктерми вже зайняли своє місце у списку і пропускаються під час сортування за розрядами молодшими від $j$.

По завершенні процедури сортування за усіма розрядами замість кон’юнктермів із поглинутими розрядами необхідно вписати список елементів від 0 до $2^r$-1, де $r$ — кількість поглинутих розрядів. Непоглинуті розряди розглядуваного кон’юнктерма дописати як старші розряди до кожного елемента утвореного списку.

Рис. 1 ілюструє дерево побітового сортування зі склеюванням на прикладі маски літералів чотирирозрядних двійкових чисел, де $l$ позначено літерал, що може приймати значення “0” чи “1”. Рискою “–” позначено поглинуті розряди. Вихідній множині елементів списку $B$ (позначимо її $\boldsymbol{B}$) відповідає маска літералів $l_3l_2l_1l_0$ (перший рядок на рис. 1). Після сортування по третьому розряду одержимо дві підмножини: $\boldsymbol{B}_0$ і $\boldsymbol{B}_1$ (другий рядок на рис. 2). $\boldsymbol{B}_0$, містить усі елементи множини $\boldsymbol{B}$ із нулем у третьому розряді (її маска літералів $0l_2l_1l_0$), і $\boldsymbol{B}_1$, містить усі елементи множини $\boldsymbol{B}$ із одиницею у третьому розряді (її маска літералів $1l_2l_1l_0$). Якщо потужність одержаної підмножини $\boldsymbol{B}_0$ чи $\boldsymbol{B}_1$ становить $2^3$, то замінюємо її зображенням у вигляді кон’юнктерма (0 – - -) чи (1 – - -) відповідно. Аналогічно сортуємо за наступним розрядом.

<table>
<tr><td colspan="16">$l_3l_2l_1l_0$</td></tr>
<tr><td colspan="8">$0l_2l_1l_0$<br>Якщо $|\boldsymbol{B}_0| = 2^3$, то існує (0 – - -)</td><td colspan="8">$1l_2l_1l_0$<br>Якщо $|\boldsymbol{B}_1| = 2^3$, то існує (1 – - -)</td></tr>
<tr><td colspan="4">$00l_1l_0$<br>(00 – -) при $|\boldsymbol{B}_{00}|=2^2$</td><td colspan="4">$01l_1l_0$<br>(01 – -) при $|\boldsymbol{B}_{01}|=2^2$</td><td colspan="4">$10l_1l_0$<br>(10 – -) при $|\boldsymbol{B}_{10}|=2^2$</td><td colspan="4">$11l_1l_0$<br>(11 – -) при $|\boldsymbol{B}_{11}|=2^2$</td></tr>
<tr><td colspan="2">$000l_0$<br>(000 -)</td><td colspan="2">$001l_0$<br>(001 -)</td><td colspan="2">$010l_0$<br>(010 -)</td><td colspan="2">$011l_0$<br>(011 -)</td><td colspan="2">$100l_0$<br>(100 -)</td><td colspan="2">$101l_0$<br>(101 -)</td><td colspan="2">$110l_0$<br>(110 -)</td><td colspan="2">$111l_0$<br>(111 -)</td></tr>
<tr><td>0000<br>0</td><td>0001<br>1</td><td>0010<br>2</td><td>0011<br>3</td><td>0100<br>4</td><td>0101<br>5</td><td>0110<br>6</td><td>0111<br>7</td><td>1000<br>8</td><td>1001<br>9</td><td>1010<br>10</td><td>1011<br>11</td><td>1100<br>12</td><td>1101<br>13</td><td>1110<br>14</td><td>1111<br>15</td></tr>
</table>

*Рис. 1. Дерево побітового сортування зі склеюванням на прикладі чотирирозрядної маски літералів*



Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua

**Приклад 1.** Упорядкувати множину чисел за їх природним порядком

$\{14,7,13,6,12,5,11,4,10,3,9,2,8,0,1\}$.

**Розв'язання 1.** У двійковому зображенні задана множина має вигляд

$\{1110,0111,1101,0110,1100,0101,1011,0100,1010,0011,1001,0010,1000,0000,0001\}$.

Сортуємо за старшим (третім) розрядом:

$\{0111,0110,0101,0100,0011,0010,0000,0001\}$, $\{1110,1101,1100,1011,1010,1001,1000\}$.

У підмножині елементів, що містять нуль у третьому розряді, кількість елементів дорівнює $2^3$, тому першу підмножину можна записати у вигляді кон'юнктерма $\{(0---)\}$, що вже зайняв своє місце у списку і не бере участі у подальшому сортуванні.

$\{(0---)\}$, $\{1110,1101,1100,1011,1010,1001,1000\}$

Сортуємо за другим розрядом:

$\{(0---)\}$, $\{1011,1010,1001,1000\}$, $\{1110,1101,1100\}$.

$\{(0---)\}$, $\{(10--)\}$, $\{1110,1101,1100\}$.

Сортуємо за першим розрядом

$\{(0---)\}$, $\{(10--)\}$, $\{1101,1100\}$, $\{1110\}$.

$\{(0---)\}$, $\{(10--)\}$, $\{(110-)\}$, $\{1110\}$.

Перейдемо від зображення у вигляді кон'юнктерма $\{(0---)\}$ до впорядкованої множини чисел. Для цього, оскільки поглинуто три розряди, запишемо у двійковому вигляді числа від 0 до ($2^3$-1). До кожного числа допишемо зліва непоглинуті розряди вихідного кон'юнктерма. Одержимо:

$\{0000,0001,0010,0011,0100,0101,0110,0111\}$.

Після відновлення усіх поглинутих розрядів

$\{0000,0001,0010,0011,0100,0101,0110,0111\}$, $\{1000,1001,1010,1011\}$, $\{1100,1101\}$, $\{1110\}$.

Остаточний результат у десятковому зображенні

$\{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14\}$.

---

Рис. 2 ілюструє у десятковому вигляді розв'язання прикладу 1. Рядок на рисунку відповідає сортуванню за певним розрядом. Перекреслений кон'юнктерм означає, що відповідна множина не склеюється і підлягає подальшому сортуванню за наступним (молодшим) розрядом. В останньому (нижньому) рядку із кон'юнктермів відновлено мінтерми та наведено десяткові еквіваленти мінтермів.

<table>
<tr><td colspan="15">{14,7,13,6,12,5,11,4,10,3,9,2,8,0,1}</td></tr>
<tr><td colspan="8">{7,6,5,4,3,2,0,1}<br>(0 – - -)</td><td colspan="7">{14,13,12,11,10,9,8}<br>~~(1 - - -)~~</td></tr>
<tr><td colspan="8" rowspan="2"></td><td colspan="4">{11,10,9,8}<br>(10 – -)</td><td colspan="3">{14,13,12}<br>~~(11 - -)~~</td></tr>
<tr><td colspan="4"></td><td colspan="2">{13,12}<br>(110 -)</td><td>{14}</td></tr>
<tr><td>0000<br>0</td><td>0001<br>1</td><td>0010<br>2</td><td>0011<br>3</td><td>0100<br>4</td><td>0101<br>5</td><td>0110<br>6</td><td>0111<br>7</td><td>1000<br>8</td><td>1001<br>9</td><td>1010<br>10</td><td>1011<br>11</td><td>1100<br>12</td><td>1101<br>13</td><td>1110<br>14</td></tr>
</table>

*Рис. 2. Ілюстрація до розв'язання прикладу 1*

## Висновки

Деякі методи мінімізації логікових функцій (зокрема, метод порозрядного вирощування простих кон'юнктермів) [1] вимагають попереднього сортування мінтермів у двійковому зображенні. Застосування побітового сортування зі склеюванням дає змогу одержати частину кон'юнктермів меншого рангу ще до початку процесу мінімізації. До того ж стає непотрібним подальший пошук склеювань мінтермів за молодшим розрядом, оскільки усі такі склеювання вже знайдено протягом сортування.

*1. Лузин С.Ю. Асимптотически оптимальный метод получения простых импликант. / Автоматика и вычислительная техника – 2000. – № 1. – С. 80–84. 2. Дональд Кнут. Искусство программирования, том 3. Сортировка и поиск = The Art of Computer Programming, vol.3. Sorting and Searching. – 2-е изд. – М.: Вильямс, 2007. – С. 824.*



Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua