Discrete Mathematics and Theoretical Computer Science DMTCS vol. **9:2**, 2007, 145–152

Gray code order for Lyndon words

Vincent Vajnovszki[1]

[1] *LE2I – UMR CNRS, Université de Bourgogne, B.P. 47 870, 21078 DIJON-Cedex France*
vvajnov@u-bourgogne.fr

received 29 Dec 2004, revised 8 Jun 2005, accepted 21 Jun 2005.

At the 4*th* Conference on Combinatorics on Words, Christophe Reutenauer posed the question of whether the *dual reflected order* yields a Gray code on the *Lyndon family*. In this paper we give a positive answer. More precisely, we present an $O(1)$-average-time algorithm for generating length n binary pre-necklaces, necklaces and Lyndon words in Gray code order.

Keywords: Lyndon words, Gray codes, generating algorithms

1 Introduction and motivation

A *k-Gray code* for a set of binary strings $B \subset \{0,1\}^n$ is an *ordered* list for B such that the Hamming distance between any two consecutive strings in the list is at most k; if k is minimal then the list is called a minimal Gray code.

The conjugacy class of a string $\mathbf{x} = \mathbf{uv}$ is the set of all strings $\mathbf{vu}$. A (binary) *necklace* is a binary string which is minimal with respect to lexicographic order in its conjugacy class and a *Lyndon word* is an aperiodic necklace. A *pre-necklace* is a binary string which is the prefix of a some necklace, or equivalently (12), of some Lyndon word. Generation of necklaces and Lyndon words lexicographically have been widely studied in the literature: (2; 3; 5; 6; 8; 13; 14; 15; 16; 17; 22).

Binary strings (necklaces, Lyndon words) with a given number of ones are called binary strings (necklaces, Lyndon words) with fixed density. There are two algorithms for generating a Gray code for necklaces with fixed density. The first one published is due to Wang and Savage in 1996 (23) (where the representative used is not the lexicographically smallest); the second one, where Lyndon words with fixed density are also considered, is due to Ueda in 2000 (18). It is worth mentioning that the second paper was submitted two years before the first one. Both Gray codes are 2-Gray codes and therefore minimal.

To the best of our knowledge, there is no Gray code for necklaces and Lyndon words with no density restriction. The aim of this note is to present the first one and it is an updated form of the preliminary conference version of the paper (20).

In the following, lower case bold letters represent length n binary strings, e.g., $\mathbf{x} = x_1x_2\ldots x_n$. The lexicographical order is defined as: $\mathbf{x}$ is less than $\mathbf{y}$ iff $x_k = 0$ and $y_k = 1$, where k is the leftmost position with $x_k \neq y_k$; and any proper prefix of a string $\mathbf{x}$ is less than $\mathbf{x}$. The *reflected*

object		absorbent	reflected order	dual reflected order
binary strings	any	yes	1-Gray code *F. Gray (7)*	1-Gray code
	fixed density	no	2-Gray code *Liu-Tang (9)*	2-Gray code
Dyck words		no	no	4-Gray code not minimal *experimental*
generalized Fibonacci strings		no	no	1-Gray code *Vajnovszki (19)*
generalized Lucas strings		no	no	1- or 2-Gray code minimal *Baril-Vajnovszki (1)*
pre-necklaces necklaces Lyndon words	any	yes	no	**3-Gray code not minimal**
	fixed density	no	no	no

Tab. 1: Combinatorial objects for which the reflected order or the dual reflected order yields Gray codes.

Gray code order on $\{0,1\}^n$, due to Frank Gray in 1953 (7), is given by (4): $\mathbf{x} < \mathbf{y}$ iff $\sum_{j=1}^{k} x_j$ is even (and $\sum_{j=1}^{k} y_j$ is odd), where k is the leftmost position with $x_k \neq y_k$.

We say that an order relation on a set of strings induces a *k-Gray code* if the set listed in this order yields a k-Gray code. For example, the reflected Gray code order induces a 1-Gray code on $\{0,1\}^n$; and its restriction to the strings with fixed density (i.e., strings with a constant number of 1's) induces a 2-Gray code, called *revolving door code* by Nijenhuis and Wilf (11).

The next definition gives a slight modification of reflected order and like lexicographical order, both of them are particular cases of *genlex order* (21), that is, any set of strings listed in such an order has the property that strings with a common prefix are contiguous.

Definition 1 *(19) We say that* $\mathbf{x}$ *is less than* $\mathbf{y}$ *in* dual reflected order *(or* local reflected order*), denoted by* $\mathbf{x} \prec \mathbf{y}$*, if* $x_1x_2\ldots x_k$*, the length* k *prefix of* $\mathbf{x}$ *contains an odd number of* 0*'s, where* k *is the leftmost position with* $x_k \neq y_k$*.*

Remark 1

1. $\mathbf{x} \prec \mathbf{y}$ *iff* $\overline{\mathbf{x}} > \overline{\mathbf{y}}$ *in reflected Gray code order, with* $\overline{\mathbf{x}}$ *and* $\overline{\mathbf{y}}$ *the bitwise complement of* $\mathbf{x}$ *and* $\mathbf{y}$*,*

2. *Like reflected Gray code order, dual reflected order* $\prec$ *induces a* 1*-Gray code on* $\{0,1\}^n$ *and a* 2*-Gray code on the strings in* $\{0,1\}^n$ *with fixed density,*

3. *In (19) and (1) it is shown that, unlike reflected Gray code order, dual reflected order* $\prec$ *induces a* 1*-Gray code on the set of generalized Fibonacci strings (the set of length* n *binary*

strings such that there are no p consecutive 1's) and on the set of generalized Lucas strings (the set of Fibonacci strings where the pattern 1^p is regarded circularly).

See Table 1.

2 The Gray code

Definition 2 *A binary string set $X \subset \{0,1\}^n$ is called* absorbent *if for any $\mathbf{x} \in X$ and any k, $1 \le k < n$, $x_1 x_2 \dots x_k 1^{n-k}$ is also a string in X.*

The next lemma shows that, in contrast to the Fibonacci or Lucas strings, the sets of pre-necklaces, necklaces and Lyndon words are absorbent (see again Table 1). Its proof is based on the following result (10, Proposition 5.1.2): a binary string is a Lyndon word iff it is lexicographically less than any of its proper suffixes. Equivalently, using the definition of lexicographic order, we have the following characterization: a binary string is a Lyndon word iff any of its proper prefixes is less than its suffix of same length.

Lemma 1 *The sets of length n Lyndon words, necklaces and pre-necklaces are absorbent.*

Proof: Let $\mathbf{x} = x_1 x_2 \dots x_n$ be a binary string and $\mathbf{x}' = x_1 x_2 \dots x_k 1^{n-k} \ne \mathbf{x}$ with $1 \le k < n$. When $\mathbf{x}$ is a Lyndon word it is easy to check that $\mathbf{x}'$ satisfies the above characterization of Lyndon words.

If $\mathbf{x}$ is a necklace or a pre-necklace (but not a Lyndon word) then there exists j, $1 \le j < n$, such that the prefix $x_1 x_2 \dots x_j$ is a Lyndon word and $\mathbf{x} = (x_1 x_2 \dots x_j)^p \, x_1 x_2 \dots x_r$ with $p = \lfloor \frac{n}{j} \rfloor$ and $r = n \mod j$. The final segment $x_1 x_2 \dots x_r$ is empty iff $\mathbf{x}$ is a necklace, see for instance (12). Thus a suffix of $\mathbf{x}'$ must have the form $x_s \dots x_j \, x_1 \dots x_t 1^{n-k}$ with the initial segment $x_s \dots x_j$ possibly empty. Such a suffix is greater than the same length prefix and therefore $\mathbf{x}'$ is a Lyndon word. □

Let $X \subset \{0,1\}^n$, $\mathbf{x} \in X$ and k be the leftmost position where $\mathbf{x}$ differs from its predecessor in X in $\prec$ order (assuming that $\mathbf{x}$ is not the first string in X in this order). In other words, since $\prec$ is a genlex order, $x_1 x_2 \dots x_k$ is the shortest prefix of $\mathbf{x}$ such that for all $\mathbf{y} \prec \mathbf{x}$ one has $y_1 y_2 \dots y_k \prec x_1 x_2 \dots x_k$. We call $x_{k+1} x_{k+2} \dots x_n$ the *first discriminant suffix of* $\mathbf{x}$. Similarly, $x_{k+1} x_{k+2} \dots x_n$, the *last discriminant suffix of* $\mathbf{x}$, is the $n-k$ suffix of $\mathbf{x}$ where k is the leftmost position where $\mathbf{x}$ differs from its successor in $\prec$ order (assuming that $\mathbf{x}$ is not the last string in X in this order). Let $|\mathbf{y}|_0$ denote the number of 0's in the binary string $\mathbf{y}$.

Lemma 2 *If $\mathbf{x}$ is a string in an absorbent subset X of $\{0,1\}^n$ and $x_{k+1} x_{k+2} \dots x_n$ the first or the last discriminant suffix of $\mathbf{x}$, then $|x_{k+1} x_{k+2} \dots x_n|_0 \le 1$.*

Proof: The proof is by contradiction. Let $x_{k+1} x_{k+2} \dots x_n$ be a first discriminant suffix of $\mathbf{x}$ such that $|x_{k+1} x_{k+2} \dots x_n|_0 > 1$. Then there are at least two indices i and j, $k+1 \le i < j \le n$, such that $x_i = x_j = 0$ and $x_{i+1} = x_{i+2} = \dots = x_{j-1} = 1$. Since X is absorbent, $\mathbf{x}' = x_1 x_2 \dots x_{i-1} 1^{n-i+1}$ and $\mathbf{x}'' = x_1 x_2 \dots x_{j-1} 1^{n-j+1}$ are both in X. In addition, by Definition 1, either $\mathbf{x}' \prec \mathbf{x}$ or $\mathbf{x}'' \prec \mathbf{x}$. So, since $\prec$ is a genlex order on X, the leftmost position where $\mathbf{x}$ differs from its predecessor is greater than k; this is in contradiction with $x_{k+1} x_{k+2} \dots x_n$ is the first discriminant suffix of $\mathbf{x}$. The case where $x_{k+1}x_{k+2}\ldots x_n$ is the last discriminant suffix of $\mathbf{x}$ is similar. □

Corollary 1 *If X is an absorbent subset of $\{0,1\}^n$ then X listed in $\prec$ order is a 3-Gray code.*

Proof: Let $\mathbf{y}$ be the successor of $\mathbf{x}$ in X with respect to $\prec$ order and k be the leftmost position where $\mathbf{x}$ differs from $\mathbf{y}$. Then, $x_{k+1}x_{k+2}\ldots x_n$ (resp. $y_{k+1}y_{k+2}\ldots y_n$) is the last (resp. the first) discriminant suffix of $\mathbf{x}$ (resp. $\mathbf{y}$). By Lemma 2, $x_i = y_i = 1$ for all i, $k+1 \le i \le n$, except possibly two of them. Since $x_i = y_i$ for all i, $1 \le i \le k-1$, the Hamming distance between $\mathbf{x}$ and $\mathbf{y}$ is at most three. □

We now state the main result of this note, which follows directly from Lemma 1 and Corollary 1.

Theorem 1 *The $\prec$ order yields a 3-Gray code on the sets of pre-necklaces, necklaces and Lyndon words of length n.*

Notice that similar results can be obtained for *bracelets* (minimal necklaces under the operation of reversal). Also, the first element in these Gray code lists is 01^{n-1} and the last one is 1^n (for pre-necklaces and necklaces) and 001^{n-2} (for Lyndon words and $n \ge 3$). Thus all these Gray codes are circular, i.e., the first and last strings differ also in at most three positions.

3 Generating algorithm

Here we give an efficient algorithm for generating length n pre-necklaces, necklaces and Lyndon words in dual reflected order. It is based on Ruskey's implementation (12) of the FKM algorithm (6; 8), which generates pre-necklaces, necklaces and Lyndon words in lexicographical order. In our Gray code version, when two values are possible for $a[t]$, the current entry in the generated string, then either 0 or 1 is assigned first, depending upon the parity of the number of 0's in the prefix $a[1]a[2]\ldots a[t-1]$. This suffices to impose dual reflected order on the outputs and is obtained by adding a new binary parameter z which has the same parity as the number of 0's in this prefix.

Like the Ruskey-FKM algorithm, our algorithm *gen_Gray* below generates length n pre-necklaces; if $Print(a)$ is called only when p is a divisor of n or when $p = n$, then the algorithm lists, respectively, necklaces or Lyndon words. The main call is $gen\_Gray(0,1,1)$ and $a[0] = 0$.

In (14) is shown that the Ruskey-FKM algorithm, and therefore also our Gray code version, generates necklaces and Lyndon words in constant average time; see also (12). The output of our algorithm for $n = 5$ is shown in Table 2.

```
procedure gen_Gray(z, t, p)
if t > n then Print(a)
else  if a[t − p] = 1
        then a[t] := a[t − p]; gen_Gray(z, t + 1, p)
        else  a[t] := z; gen_Gray(1, t + 1, p)
              a[t] := 1 − z; gen_Gray(0, t + 1, t)
end.
```

pre-necklaces	necklaces	Lyndon words
0 1 1 1 1	0 1 1 1 1	0 1 1 1 1
0 1 1 1 0	0 1 0 1 1	0 1 0 1 1
0 1 1 0 1	0 0 0 1 1	0 0 0 1 1
0 1 0 1 0	0 0 0 0 0	0 0 0 0 1
0 1 0 1 1	0 0 0 0 1	0 0 1 0 1
0 0 0 1 1	0 0 1 0 1	0 0 1 1 1
0 0 0 1 0	0 0 1 1 1	
0 0 0 0 0	1 1 1 1 1	
0 0 0 0 1		
0 0 1 0 1		
0 0 1 0 0		
0 0 1 1 0		
0 0 1 1 1		
1 1 1 1 1		

Tab. 2: The set of length 5 pre-necklaces, necklaces and Lyndon words in dual reflected order. Note that for $n \leq 5$ the Hamming distance between successive strings never exceeds two.

4 Final remarks

It can be shown that, in general, there is no listing for length n necklaces in which successive representatives differ in just one bit (23). The existence of Gray codes for necklaces or Lyndon words where successive objects differ in at most two positions and a loopless implementation ($O(1)$ in the worst case) of our algorithm remain open problems. Finally, the author would like to thank Frank Ruskey for providing improvements to the algorithm and one of the referees for helpful comments.

Note added in proof: A recent subsequent generalization of our results to a k-ary alphabet ($k > 2$) appears in (24).

References

[1] J.L. Baril, V. Vajnovszki. Minimal change list for Lucas strings and some graph theoretic consequences. *Theoret. Comput. Sci.*, **346** (2-3), 189-199, 2005.

[2] J. Berstel, M. Pocchiola. Average cost of Duval's algorithm for generating Lyndon words. *Theoret. Comput. Sci.* **132**(1–2) (1994), 415–425.

[3] K. Cattell, F. Ruskey, J. Sawada, M. Serra, C.R. Miers. Fast algorithms to generate necklaces, unlabeled necklaces, and irreducible polynomials over GF(2). *Journal of Algorithms* **37**(2) (2000), 267–282.

[4] P.J. Chase. Combination generation and graylex ordering. *Congr. Numer.* **69** (1989), 215–242.

[5] J.-P. Duval. Génération d'une section des classes de conjugaison et arbre des mots de Lyndon de longueur bornée. *Theoret. Comput. Sci.* **60**(3) (1988), 255–283.

[6] H. Fredricksen, J. Maiorana. Necklaces of beads in k colors and k-ary de Bruijn sequences. *Discrete Math.* **23**(3) (1978), 207–210.

[7] F. Gray. Pulse code communication. U. S. Patent 2632058 (1953).

[8] I.J. Kessler, H. Fredricksen. An algorithm for generating necklaces of beads in two colors. *Discrete Math.* **61**(2–3) (1986), 181–188.

[9] C.N. Liu, D.T. Tang. Algorithm 452, enumerating m out of n objects. *Comm. ACM* **16** (1973), 485.

[10] M. Lothaire. *Combinatorics on words.* Cambridge Mathematical Library, 1973.

[11] A. Nijenhuis, H.S. Wilf. *Combinatorial Algorithms for Computers and Calculators.* Academic Press, 1978.

[12] F. Ruskey. *Combinatorial generation.* In preparation.

[13] F. Ruskey, C.R. Miers, J. Sawada. The number of irreducible polynomials and Lyndon words with given trace. *SIAM J. Discrete Math.* **14**(2) (2001), 240–245.

[14] F. Ruskey, C. Savage, T.M.Y. Wang. Generating necklaces. *Journal of Algorithms* **13**(3) (1992), 414–430.

[15] F. Ruskey, J. Sawada. An efficient algorithm for generating necklaces with fixed density. *SIAM J. Comput.* **29**(2) (1999), 671–684.

[16] J. Sawada. A fast algorithm to generate necklaces with fixed content. *Theoret. Comput. Sci.* **301**(1–3) (2003), 477–489.

[17] J. Sawada, F. Ruskey. Generating Lyndon brackets. *Journal of Algorithms* **46**(1) (2003), 21–26.

[18] T. Ueda. Gray codes for necklaces. *Discrete Math.* **219**(1-3) (2000), 235–248.

[19] V. Vajnovszki. A loopless generation of bitstrings without p consecutive ones. *Discrete Mathematics and Theoretical Computer Science - Springer*, 2001, 227–240.

[20] V. Vajnovszki. Gray viziting Lyndons, *Journées Montoises d'Informatique Théorique à Liège*, 8–11 Septembre 2004.

[21] T. Walsh. Generating Gray codes in $O(1)$ worst-case time per word. *DMTCS03, Lect. Notes Comput. Sci. – Springer* **2731** (2003), 72–88.

[22] T.M.Y. Wang, C.D. Savage. Generating necklaces. *Journal of Algorithms* **13**(3) (1992), 414–430.

[23] T.M.Y. Wang, C.D. Savage. A Gray code for necklaces of fixed density. *SIAM J. Discrete Math.* **9**(4) (1996), 654–673.

[24] M. Weston, V. Vajnovszki. Gray codes for necklaces and Lyndon words of arbitrary base, *GASCOM'06*, pp. 55-62, Dijon/France, 11-15 Septembre 2006. To appear in *Pure Mathematics and Applications/Algebra and Theoretical Computer Science*.