Some New Combinatorial Agorithms with Appropriate Representations of Solutions

Hoang Chi Thanh[1,*], Nguyen Quang Thanh[1]

Khoa Môi trường, Trường Đại học Khoa học Tự nhiên, ĐHQGHN, 334 Nguyễn Trãi, Hà Nội, Việt Nam
Faculty of Environmental Sciences, VNU University of Science, 334 Nguyễn Trãi, Hanoi, Vietnam
Faculty of Chemistry, VNU University of Science, 19 Lê Thánh Tông, Hoàn Kiếm, Hanoi, Vietnam
Khoa Hoá học, Trường Đại học Khoa học Tự nhiên, ĐHQGHN, 19 Lê Thánh Tông, Hà Nội, Việt Nam
Khoa Luật, Đại học Quốc gia Hà Nội, 144 Xuân Thủy, Cầu Giấy, Hà Nội
VNU School of Law, 144 Xuân Thủy, Cầu Giấy, Hanoi, Vietnam

Received 10 October 2014
Revised 30 October 2014; Accepted 30 November 2014

Nhận ngày 10 tháng 10 năm 2014
Chỉnh sửa ngày 30 tháng 10 năm 2014; Chấp nhận đăng ngày 20 tháng 11 năm 2014

Abstract: Combinatorial problems are those problems, whose requirements are an association of some conditions. The construction of efficient algorithms to find solutions of the combinatorial problems is still an interesting matter. In this paper, we choose appropriate representations for

1. Introduction

Permutations and partitions of a finite set are applied in many areas of sciences and technologies, e.g. scheduling problem, control problem and path finding problem... So modifying an exciting algorithm or constructing a new algorithm to generate permutations or partitions are attracted many researches [1-6,8].

* Corresponding author. Tel.: 84-912011765
Email: thanhhc@vnu.vn
Tác giả liên hệ. ĐT.: 84-912011765
Email: thanhhc@vnu.vn

2. Permutations generation by inv

ersion vectors

Let X be a finite set. *A permutation* of the set X is a checklist of its all elements. It is easy to see that each permutation of the set X is a bijection from X to itself.

2.1 Permutation problem

It is easy to show that the number of permutations is $n!$. The number of elements n is as big as great the time to find all permutations.

2.3.1 Generating inversion vectors

To perform the step 1 we consider each inversion vector $(d_1, d_2, ..., d_n)$ in V_n as a word $d_1\ d_2\ ...\ d_n$ on the alphabet $\mathcal{N}$. We sort these words in ascending by the alphabetical order (see Example 2.3).

Example 2.3: The sequences of permutations and inversion vectors of an 3-element set

No	Permutations	Inversion vectors
1	1 2 3	0 0 0
2	1 3 2	0 0 1
3	2 3 1	0 0 2
4	2 1 3	0 1 0
5	3 1 2	0 1 1
6	3 2 1	0 1 2

3. Conclusion

In this paper, we propose two new efficient algorithms to generate all permutations and all partitions of a finite set. Permutations are represented by inversion vectors whilst partitions by sequences of indices. The alphabetical order is used to sort representations of the problem's solutions in both algorithms. The obtained results point out that choosing appropriate representations for desirable solutions takes a great part in algorithm design. It makes an algorithm simpler, shorter and faster.

Acknowledgment

The author would like to acknowledge the Asia Research Center and the Korea Foundation for Advanced Studies as the sponsors for the research project.

References

[1] K. Cameron, E.M. Eschen, C.T. Hoang and R. Sritharan, The list partition problem for graphs, *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, New Orleans 2004, pp. 391-399.

[2] J. Ginsburg, Determining a permutation from its set of reductions, *Ars Combinatoria*, No. 82, 2007, pp. 55-57.

[3] T. Kuo, A new method for generating permutations in lexicographic order, *Journal of Science and Engineering Technology*, Vol. 5, No. 4, 2009, pp. 21-20.

[4] M. Monks, Reconstructing permutations from cycle minors, *The Electronic Journal of Combinatorics*, No. 16, 2009, #R19.

[5] W. Lipski, Kombinatoryka dla programistów, WNT, Warszawa, 1982.

[6] H.C. Thanh, *Combinatorics*, VNUH Press, 1999 (in Vietnamese).

[7] H.C. Thanh, *Bounded sequence problem and some its applications*, Proceedings of Japan-Vietnam Workshop on Software Engineering, Hanoi - 2010, pp. 74-83.

[8] H.C. Thanh, N.T.T. Loan, N.D. Ham, *From Permutations to Iterative Permutations*, International Journal of Computer Science Engineering and Technology, Vol. 2, Issue 7, 2012, pp. 1310-1315.