IOP Conf. Series: Materials Science and Engineering **237** (2017) 012026 doi:10.1088/1757-899X/237/1/012026

A review: search visualization with Knuth Morris Pratt algorithm

Robbi Rahim[1*], Iskandar Zulkarnain[2] and Hendra Jaya[3]
[1]Postgraduate Ph.D. Student Universiti Malaysia Perlis, Universiti Malaysia Perlis, Malaysia
[1]Department of Health Information, Akademi Perekam Medik dan Infokes Imelda, Jl. Bilal Ujung Medan 20116, Indonesia
[2]Department of ComputerSystem, STMIK Triguna Dharma, Jl. Jenderal Abdul Haris Nasution No.73, Medan 20219, Indonesia
[3]Department of Computer Engineering, STMIK Triguna Dharma, Jl. Jenderal Abdul Haris Nasution No.73, Medan 20219, Indonesia

*usurobbi85@zoho.com

Abstract. In this research modeled a search process of the Knuth-Morris-Pratt algorithm in the form of easy-to-understand visualization, Knuth-Morris-Pratt algorithm selection because this algorithm is easy to learn and easy to implement into many programming languages.

1. Introduction

Searching is a work that is often performed in everyday life. There are times when we look for something with the purpose of just knowing whether the data is in a set of data or not, while at other times we may want the position of the data sought [1].

A significant difference exists in the string matching algorithm, if in the search process it is not necessary to apply a string matching algorithm to test the data accordingly or not enough using IF logic[1], whereas in the string matching algorithm by default use the search concept to make the process better in terms of string matching[2].

The Knuth-Morris-Pratt (KMP) algorithm is one of the string matching algorithms (pattern) with the examination process from left to right[2][3][4][5], in this research the Knuth-Morris-Pratt algorithm is described in the form of visualization to facilitate the string matching process, Visualization of the Knuth-Morris-Pratt process in this study makes it easy for other researchers to find out how the string matching and searching process works. It is expected that the Knuth-Morris-Pratt algorithm works to facilitate its application to various search processes such as search engines, plagiarism detection and so on.

2. Theory

The Knuth-Morris-Pratt algorithm was developed by D. E. Knuth, along with J. H. Morris and V. R. Pratt[1][3][6]. The Knuth-Morris-Pratt algorithm is the development of the previous string search algorithm, the Brute Force algorithm. Brute-Force algorithm is the simplest basic algorithm in solving string matching problem that examines every position in the text between 0 and n-m, where n is the length of text/number of filenames stored on the computer and m is the character length of a pattern (the word to search)[7].

KMP algorithm (Knuth Morris Pratt) is an algorithm used to perform string matching process. This algorithm is a type of Exact String Matching Algorithm which is a precise string matching with the arrangement of characters in a matched string having the number and sequence of characters in the same string. Example: the algorithmic word will show a matching only with the algorithmic word [2][3].

In the KMP algorithm, the information used to do a further shift, not just one character like the Brute Force algorithm. This algorithm performs matching from left to right[5].

3. Result and Discussion

Here is the process of applying the Knuth Morris Pratt algorithm to search for a word in a sentence sequence, as for the process as follows:

a. Given the S string variable with an array of letters as follows:

D	A	E	L	Y	M	M	A	K	R	I	N	A	A	M	R	S	W

b. Given pattern, variable P is the word to be searched in variable S

R	I	N	A

c. The first step
Compare pattern P [1] with string S [1], here is the result

D	A	E	L	Y	M	M	A	K	R	I	N	A	A	M	R	S	W
R	I	N	A														

Pattern [1] does not match sequence [1] then the pattern will shift one position to the right.

d. Step Two
Compare pattern P [1] with string S [2], here is the result

D	A	E	L	Y	M	M	A	K	R	I	N	A	A	M	R	S	W
	R	I	N	A													

Pattern [1] does not match sequence [2] then the pattern will move one position to the right.

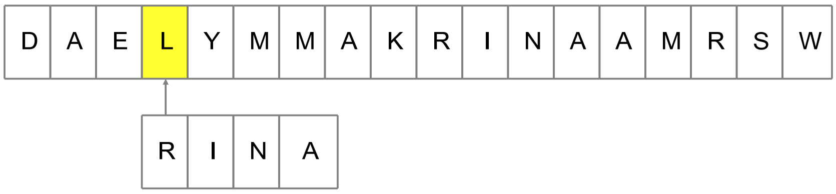
e. Step Three
Compare pattern P [1] with string S [3], here is the result

D	A	E	L	Y	M	M	A	K	R	I	N	A	A	M	R	S	W
		R	I	N	A												

Pattern [1] does not match sequence [3] then the pattern will move one position to the right.

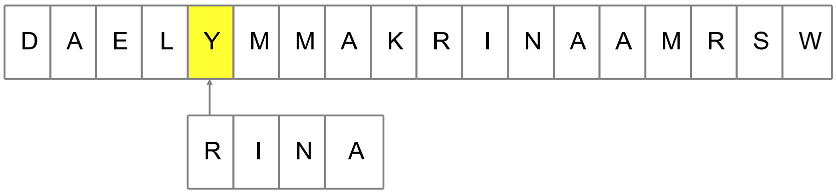
f. Step Four
Compare pattern P [1] with string S [4], here is the result

IOP Conf. Series: Materials Science and Engineering **237** (2017) 012026 doi:10.1088/1757-899X/237/1/012026



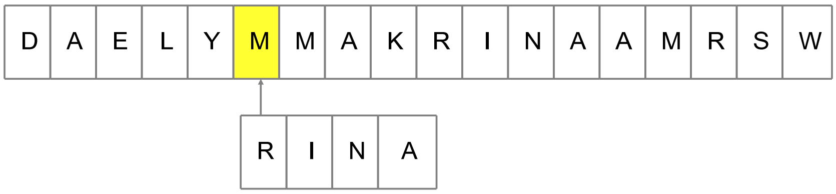
Pattern [1] does not match string [4] then the pattern will shift one position to the right.

g. Step Five
Compare pattern P [1] with string S [5], here is the result



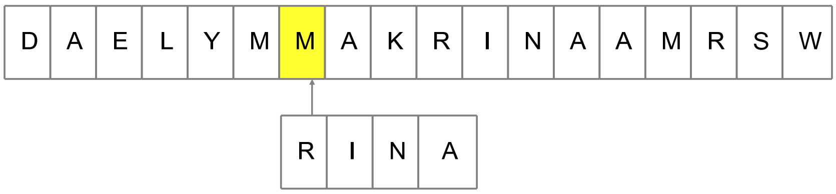
Pattern [1] does not match sequence [5] then the pattern will shift one position to the right.

h. Step Six
Compare pattern P [1] with string S [6], here is the result



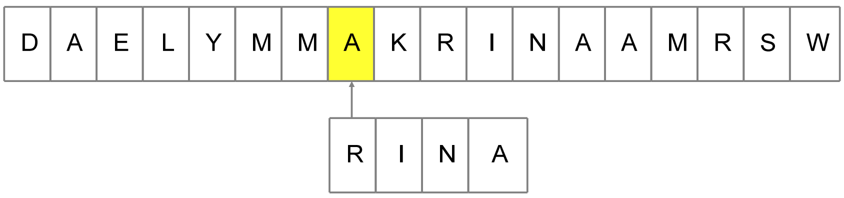
Pattern [1] does not match sequence [6] then pattern will shift one position to right

i. Step Seven
Compare pattern P [1] with string S [7], here is the result



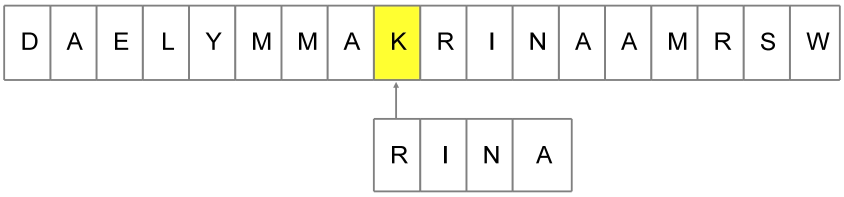
Pattern [1] does not match string [7] then pattern will move one position to right

j. Step Eight
Compare pattern P [1] with string S [8], here is the result



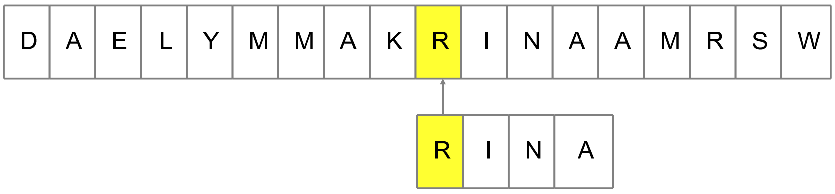
Pattern [1] does not match sequence [8] then pattern will shift one position to right

k. Step Nine
Compare pattern P [1] with string S [9], here is the result

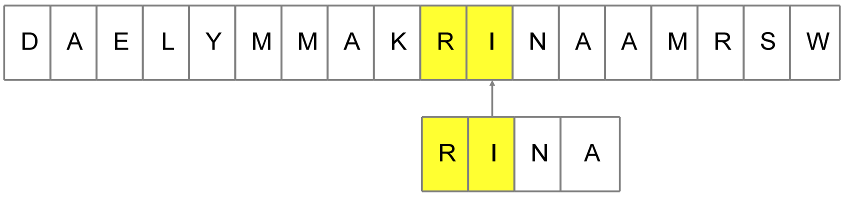


Pattern [1] does not match string [9] then pattern will move one position to right

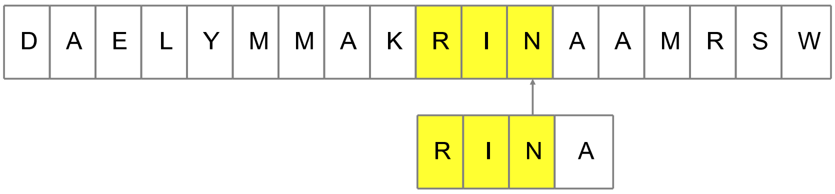
1. Step Ten
   Compare pattern P [1] with string S [10], here is the result



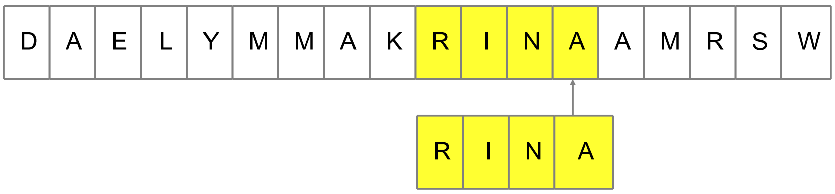
Pattern [1] matches the string [10]. Because there is a match, the Knuth Morris Pratt algorithm will store this information, and the pattern will not shift and continue matching pattern [2] with string [11].



Pattern [1, 2] matches the string [10, 11]. Because there is a match, the Knuth Morris Pratt algorithm will store this information, and the pattern will not shift and continue matching pattern [3] with string [12].



Pattern [1, 2, and 3] matches the string [10, 11, and 12]. Because there is a match, the Knuth Morris Pratt algorithm will store this information, and the pattern will not shift and continue matching pattern [4] with string [13].



Pattern [4] matches the string [13]. Because there is a match, the Knuth Morris Pratt algorithm will store this information, and the pattern will not shift and continue matching pattern [5] with string [14]. However, since the number of patterns is only four letters, the search will be stopped and the result is that the P pattern matches S-string by 100 percent.

IOP Conf. Series: Materials Science and Engineering **237** (2017) 012026 doi:10.1088/1757-899X/237/1/012026

4. Conclusion

The process visualization of the Knuth-Morris-Pratt (KMP) algorithm allows researchers or scholars to learn how the KMP algorithm works, and in the application development it will be easier to create a function for word search and can be implemented into many search processes.

References

[1] N. R. Dalal and P. Jadhav, 2015, "A Composite Algorithm for String Matching," *International Journal of Modern Trends in Engineering and Research (IJMTER),* vol. 2, no. 7, pp. 68-73.

[2] R. Y. Tsarev, A. S. Chernigovskiy, E. A. Tsareva, V. V. Brezitskay, A. Y. Nikiforov and N. A. Smirnov, 2015, "Combined string searching algorithm based on Knuth-morris-pratt and Boyer-Moore algorithms," in *XIX International Scientific Conference Reshetnev Readings*, Rusia.

[3] I. M. Abu-Zaid and E. K. El-Rayyes, 2012, "Parallel Search Using KMP Algorithm in Arabic String," *International Journal of Science and Technology,* vol. 2, no. 7, pp. 427-431.

[4] R. Janani and S. Vijayarani, 2016, "An Efficient Text Pattern Matching Algorithm for Retrieving Information from Desktop," *Indian Journal of Science and Technology,* vol. 9, no. 43, pp. 1-11.

[5] C. S. Rao, K. B. Raju, and S. V. Raju, 2013, "Parallel String Matching with Multi-Core Processors - A Comparative Study for Gene Sequences," *Global Journal of Computer Science and TechnologyHardware & Computation,* vol. 13, no. 1, pp. 26-41.

[6] G. Pandey and G. L. Prajapati, 2016, "Applying Bi-Directional Search Strategy in Selected String Matching Algorithms," *International Journal of Computer Applications,* vol. 143, no. 11, pp. 40-43.

[7] D. Shapira and A. Daptardar, 2006, "Adapting the Knuth–Morris–Pratt algorithm for pattern matching in Huffman encoded texts," *Elsevier: Information Processing and Management,* vol. 42, pp. 429-439.