

Java Characters Word Processing

Rudy Adipranata
Informatics Department
Petra Christian University
Surabaya, Indonesia
rudya@petra.ac.id

Gregorius Satia Budhi
Informatics Department
Petra Christian University
Surabaya, Indonesia
greg@petra.ac.id

Rudy Thedjakusuma
Informatics Department
Petra Christian University
Surabaya, Indonesia

ABSTRACT

Java writing has its own unique characteristics compare to Latin writing. Therefore, Java writing is more complicated to be learned and applied. Along with the development of technology, the typing method of Java characters could be helped by creating Java characters word processing software which is helpful to write Java characters more easily. Also, the software will be very helpful to preserve cultural values. This word processing has been tested and running well to produce the raw Java characters include basic, supplementary and 'pasangan' characters.

Keywords

Word processing, Java characters

1. INTRODUCTION

Indonesia is a country which composed of many islands. Indonesia has a national language, Bahasa Indonesia. In addition to Bahasa Indonesia, there are many local languages in each region. The diversity of local languages is a cultural value that should be preserved for new generations.

Javanese language is one of the local languages that is widely used instead of Bahasa Indonesia. Javanese language has unique Java characters and different compared to Latin characters. Because of its uniqueness, Java characters are more complicated to learn and apply in the community. Along with the development of technology, the typing method of Java characters could be helped by creating Java characters word processing software which is helpful to write Java characters more easily. In this paper, we present the development of Java characters word processing that can produce basic, supplementary and 'pasangan' characters.

2. WORD PROCESSING

Word processing is an application or computer software used for creating, editing, and printing documents. Word processing software can combine images, graphics and text [2]. In addition to these advantages, there are several other advantages, namely, insert text: text to include in any document section, delete text: to remove words, cut and paste: to remove (cut) some text from one place between the document and paste elsewhere, copy and paste: copy some text from one place and paste to another place, find: search the text in document, print: sends document to printer.

3. AUTOMATA

Automata are abstract machines that can recognize, accept, or generate a sentence in a particular language [2]. Automata is also

a system consists of a finite number of states of the abstract machine that receives input and generate output in the form of a discrete issue (one at a time). The state is a condition that describe information about the last input. State is also regarded as the machine's memory. The input of the automata considered the language to be recognized by the machine. After receive input, automata engine will determine whether the input is acceptable or not acceptable. Automata in a simple circuit are shown at Figure 1.

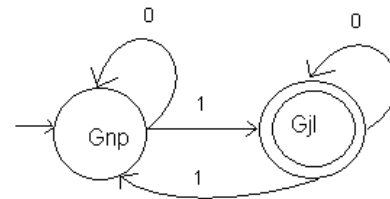


Figure 1. Simple Circuit Automata

Characteristics of automata: receiving input, generate output, has a buffer, able to make decisions in transforming inputs into outputs. Automata are also distinguished by the type of temporary memory it has, consists of three types: finite automata, pushdown automata and turing machine [2].

Finite automata don't have a temporary memory. Finite automata are a class of machines with the most limited abilities. Finite automata are used to aid the design of lexical analyzer, a text editor application, pattern recognition, fault tolerant systems, etc.

Pushdown automata (PDA) have LIFO (last in, first out) temporary memory mechanism. This engine is more powerful because of the presence of stack memory unit. Pushdown automata are used to identify the language that has context-free grammar, dictionary data, query, scripting, parsing, etc.

Turing machine has random access temporary memory mechanism. Turing machine can be used to identify the impossibility of writing a computer program. Turing machine is a mathematical model that is widely used for today's computers.

Some basic understanding of automata: the symbol is an abstract entity (similar to point in geometry). A letter or a number is an example of a symbol. String is a finite row symbols. For example, if a, b, and c are three symbols then abcb is a string constructed from those symbols. If w is a string then the string length is expressed as $|w|$ and defined for the number of symbols that make up the string. For example, if $w = abcb$ then $|w| = 4$. Empty string is a string with no symbols. Empty string expressed by the symbol ϵ or \wedge so that $|\epsilon| = 0$. Empty string can be viewed

as an empty symbol because both are composed of no symbols. Alphabet is the finite set of symbols

Grammar can be defined as a collection of sets of variables, terminal symbols, a symbol of the beginning, which is limited by rules. The example of automata can be seen at Figure 2.

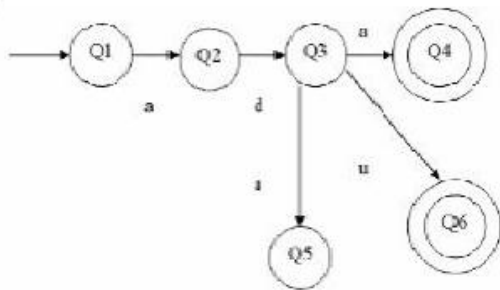


Figure 2. Sample of Automata

The rule of machine in Figure 2 is input string will be accepted if and only if it reaches the final state which is symbolized by a double circle. The machine has 6 states {Q1, Q2, Q3, Q4, Q5, Q6}. The machine has the initial state Q1. The machine has final state {Q4, Q6}. For example, if machine receives string “ada” or “adu”, it will be accepted as correct input, but if machine receives string “add”, it will be rejected.

4. JAVA CHARACTERS

4.1 Basic Java Characters

There are 20 basic Java characters, can be seen at Table 1 [4].

Table 1. Basic Java Characters

	Read		Read		Read		Read
ᮀ	ha	ᮁ	da	ᮂ	pa	ᮃ	ma
ᮄ	na	ᮅ	ta	ᮆ	dha	ᮇ	ga
ᮈ	ca	ᮉ	sa	ᮊ	ja	ᮋ	ba
ᮌ	ra	ᮍ	wa	ᮎ	ya	ᮏ	tha
ᮐ	ka	ᮑ	la	ᮒ	nya	ᮓ	nga

4.2 Supplementary Characters

Supplementary characters in Java characters are used for several purposes. The first is to write vowels (i, u, é, e, o), because basic Java characters are actually syllables ended with ‘a’. If the syllable begin with vowel, basic character ‘ha’ is used and combine with supplementary character. The second, supplementary is used to write some consonant in the end of word (r, h, ng) and to mark the end of the word if the word ended with consonant. Also, supplementary is used to write double consonant which one of the consonants is r or y (ex: pre, kya) and to write comma and period [4].

Table 2. Supplementary Characters

Symbol	Example	Read
ᮏ	ᮏᮁ	yi
ᮐ	ᮐᮁ	ye'
ᮑ	ᮑᮁ	ye
ᮒ	ᮒᮁ	yo
ᮓ	ᮓᮁ	yu
ᮔ	ᮔᮁ	yar
ᮕ	ᮕᮁ	yah
ᮖ	ᮖᮁ	yang
ᮗ	ᮗᮁ	k (consonant at the end of word)
ᮘ	ᮘᮁ	kra
ᮙ	ᮙᮁ	kre
ᮚ	ᮚᮁ	kya
ᮛ	ᮛᮁ	pa, (comma)
ᮜ	ᮜᮁ	pa. (period)

4.3 Consonant at the End of Syllable

Java character are actually syllables ended with ‘a’. To write consonant at the end of syllable, there are symbols called ‘pasangan’. Each basic character has its own ‘pasangan’ [4]. ‘Pasangan’ has function to connect a closed syllable (syllable ending with consonant) to the next syllable.

For example, in the word ‘banda’ consists of two syllables: ‘ban’ and ‘da’. To write the syllable ‘ban’, the character ‘ba’ should be written and followed by character ‘na’. It would construct ‘bana’. There is ‘a’ at the end of ‘bana’ This ‘a’ should be eliminate by using ‘pasangan’ of the next syllable. So we should write ‘pasangan’ ‘da’ instead of character ‘da’, to construct ‘banda’. The ‘pasangan’ ‘da’ has function to connect the closed syllable

with the next syllable. It means the vowels followed the character will be turned off so that it becomes consonant.

All the basic characters have their respective 'pasangan'. Some of these 'pasangan' should be written below basic character and some of them should be written aligned with basic character. To understand more clearly how to write 'pasangan', can look at Table 3 [4].

Table 3. Pasangan

Basic Char.	Pasangan	Read	Basic Char.	Pasangan	Read
ᮊ	ᮊ	ha	ᮋ	ᮋ	pa
ᮌ	ᮌ	na	ᮍ	ᮍ	dha
ᮎ	ᮎ	ca	ᮏ	ᮏ	ja
ᮐ	ᮐ	ra	ᮑ	ᮑ	ya
ᮒ	ᮒ	ka	ᮓ	ᮓ	nya
ᮔ	ᮔ	da	ᮕ	ᮕ	ma
ᮖ	ᮖ	ta	ᮗ	ᮗ	ga
ᮙ	ᮙ	sa	ᮚ	ᮚ	ba
ᮛ	ᮛ	wa	ᮜ	ᮜ	tha
ᮞ	ᮞ	la	ᮟ	ᮟ	nga

4.4 Numbers

Number symbols in Java character, can be seen at Table 4 [3].

Table 4. Number Symbols

Symbol	Num.	Symbol	Num.	Symbol	Num.
ᮠ	0	ᮡ	4	ᮢ	8
ᮣ	1	ᮤ	5	ᮥ	9
ᮦ	2	ᮧ	6	ᮨ	10
ᮩ	3	᮪	7	᮫	11

5. SYSTEM DESIGN

We design the Java characters word processing to be used with common Latin character keyboard. System will receive input from keyboard, and will generate the Java characters depend on the combination of Latin characters input. For example, if we press character 'm', followed by 'u' in keyboard, system will generate Java character for 'mu' (consist of basic character 'ma' and supplementary character 'u'). If we type sentence 'banda', system will generate Java character 'ba', 'na', and 'pasangan' 'da'.

Because not all Java characters could represent Latin character, the conversion should be used: Latin character 'z' will convert to 'j' in Java character, Latin 'q' will convert to 'k' in Java, Latin 'v' and 'f' will convert to 'p' in Java, and Latin 'x' will convert to 'ks' in Java (basic character 'ka' and 'pasangan' 's'). To differentiate between e and é in Java character, the Shift key is used for é.

In system design, DFA (deterministic finite automata) is used to determine validity of input combination. The DFA design can be seen in Figure 3.

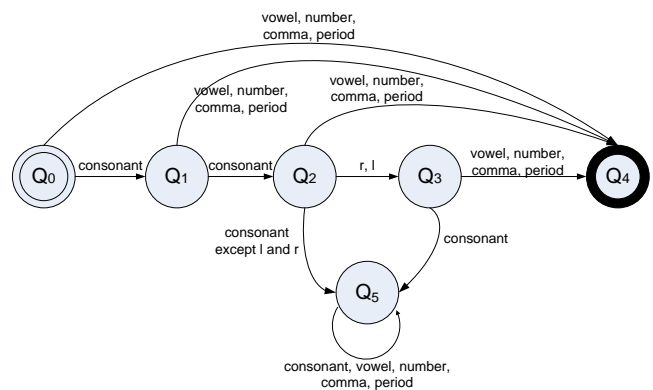


Figure 3. DFA Design

Input are consist of consonant (all alphabets except a,i,u,e,o), vowel (a,i,u,e,o), number (0..9), comma and period. Other input is discarded. Q₀ is beginning state of the system, if user input is consonant, it will go to state Q₁, but if user input is number, vowel, comma or period, it will go to accepted state Q₄. State Q₁ is similar to state Q₀, if user input is consonant, it will go to next state, but if user input is number, vowel, comma or period, it will go to accepted state Q₄. In state Q₂, the valid input only r, l, number, vowel, comma or period. If user input is r or l, it will go to state Q₃, and if user input is number, vowel, comma or period, it will go to accepted state Q₄. In state Q₃, valid input is only number, vowel, comma or period that will go to accepted state Q₄. Q₅ is error state. After system reach accepted state Q₄, system will execute rule to generate Java characters, but if system reach state Q₅, system will generate error message.

The rule to generate Java character is shown in pseudocode *GenerateChar* below. Input string *x* must be already passed the validation checking by DFA.

```

Procedure GenerateChar(string x);
//state Q0
if IsVowel(x[0])
  ShowCharacter('h'+x);
  
```


Java text area. If the user typed a character that does not comply with the rules of Java word, then they will be displayed on the wrong text area. For example, if user typed any characters that can't be translated to Java characters as seen at Figure 5, the characters will be shown at wrong text area.

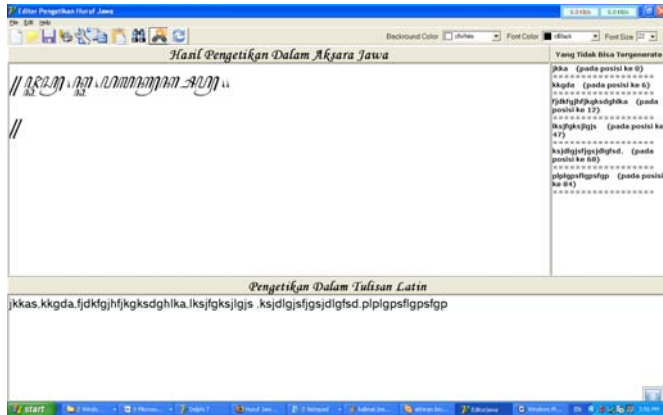


Figure 5. Writing any Characters

In this word processing, there are some standard features, such as copy/cut & paste, find & replace, change Java font color and font size and print.

The example of paragraph in the word processing can be seen at Figure 6.

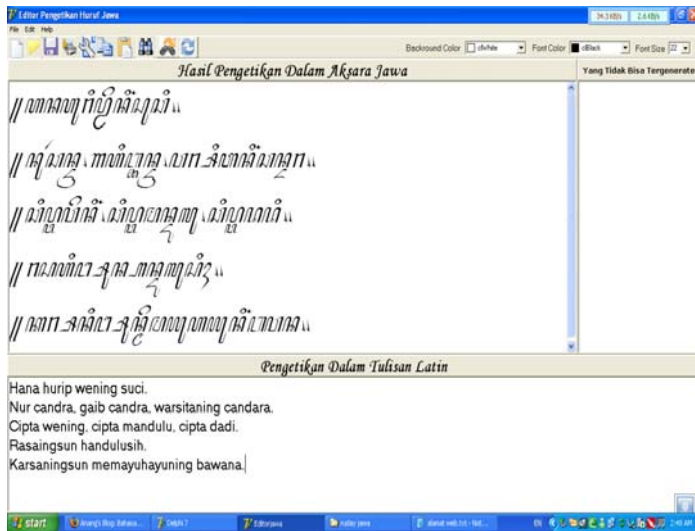


Figure 6. Example of Paragraph

7. CONCLUSION

We have developed Java character word processing that can be used to type Java character easily by using common Latin character keyboard. This word processing can display the combination of Java characters include basic, supplementary and 'pasangan' characters, according to the typing keys on the keyboard, and has features like copy, cut, paste, find, replace and print. Using this word processing, people can convert Latin characters into Java characters even if they do not understand Java characters. This word processing could be further expanded to include another Java characters such as *murda*, *swara* or *rekaan* characters.

8. REFERENCES

- [1] Eugene X. 2004. Theory of automata, formal languages and computation. New Age International
- [2] Barnes, Ian. 2006. Preservation of word processing documents. Australian National University
- [3] Daryanto, 1999. Kawruh basa jawa pepak. Apollo, Surabaya.
- [4] Tofani, Abi & Nugraha, Setyo. n.d. Tatanan anyar pinter basa jawi pepak. Pustaka Agung Harapan, Surabaya