

THE ALGORITHM OF GIVING AUTOMATIC HARAKAT ON ARABIC SCRIPT

^aDini Hidayatul Qudsi, ^bMaksum Ro'is Adin Saf

^aInformation System Department, Politeknik Caltex Riau, Pekanbaru-Indonesia

^bInformatics Engineering Department, Politeknik Caltex Riau, Pekanbaru-Indonesia

E-mail: dinihq@pcr.ac.id

Abstract

The majority of Arabic reference books in Arabic writing without a vowel/harakat cause students to have difficulty in learning Arabic. Reading Arabic Script without harakat requires an understanding of the science of Nahwu-Sharf (Arabic Grammar) to be able to understand its contents. Therefore, in this study the basic rules of Nahwu-Sharf Science have been translated into an algorithm that can be used to provide an automatic harakat on Arabic script (without harakat). Based on the interview results with the expert, six examples of sentences that represented the types(conditions) of basic sentences in the Arabic language have been used as scenarios to examine the generated algorithm. The allowed sentence is limited to one sentence only and cannot be in the form of poetry or satire. Additionally, Cyclomatic Complexity Testing is used to examine scenarios in a test case. All the steps of the Cyclomatic Complexity have been performed, which are creating the flowgraph, calculating the independent path, and testing scenario. The scenarios were examined in a test case through 13-path test case examination and result match with expected output. In addition, 24 examples which represented the six conditions have been utilized to examine for more detailed analysis and also results accuracy 100% in accordance with the basic rules of Nahwu-Sharf science.

Keywords: Algorithm, Arabic Grammar (Nahwu Sharf Science), Cyclomatic Complexity Testing, Harakat

INTRODUCTION

Learning Foreign languages provide many opportunities and benefits not just for personal needs but also in businesses, academics and communications. Arabic is one foreign language that is not less important to master, especially for Muslims around the world. Sirajudeen&Adebisi [1] suggested that Arabic is a Mother Language of over 300 million people and is used as a National Language in 20 countries. In Indonesia alone, some Indonesian words are adopted from Arabic. Chik [2] stated that there are 3303 Arabic words adopted into Malay.

Given the majority of Indonesian people are Muslims, learning Arabic can facilitate and improve their understanding of Islamic knowledge because Arabic is the language of the holy book of Al Quran and Hadith Scripture which are references for every Muslim in worship and daily life. Oxford [3] suggested that appropriate styles and strategies in learning language can be regarded as major factors in determining student's success in learning a foreign language.

However, in its implementation, Arabic writing which is not supplemented with a vowel/harakat making it difficult for students who want to master Arabic language in which they are accustomed to reading Arabic with harakat and only able to guess how to read Arabic writing without a vowel. This statement is supported by Greene [4] who stated that one of the challenges in learning Arabic is reading a text without vowels.

Al-Jurjani in Abdurrahman [5] also stated that Nahwu is the science that can specify how to arrange the structure of words correctly. In Arabic Grammar (Nahwu-Sharf Science), sentence is called Jumlah, which means a collection of several words. Sentences in Arabic are divided into two types, namely: Jumlah Ismiyah (noun) and Jumlah Fi'liyah (verb) [6]. There are certain conditions in a particular case that depends on the position and the function of the word in the sentence, which are: Rafi', Nasb, Jarr and Jazm [7]. If the word is singular, then to be able to see which case it belongs will depend on what harakah is at the end of the word [8].

In addition, nowadays computer technology can be found anywhere and has become a part of every day's life. Web-based learning media

that utilize technology can be a solution for students

to learn reading Arabic writing without harakat. A correct translation of Nahwu-Sharf science into programming algorithms can be used in developing an automated harakat system of Arabic Writing.

An algorithm is a procedure that underlies each command in a computer program that can solve problems involving data processing, computations, or both [9]. There are several ways to specify the algorithm, which are by using the natural language, pseudo code or flowchart. Natural language is the simplest technique to identify algorithms, but often programmers find it difficult to translate into programming languages. Therefore, programmers prefer to utilize other techniques that is clearer and more accurate, such as pseudo code, which is a combination of natural language and programming language. Meanwhile, flowchart can also be used to represent the algorithm in a graphical representation [10].

In order to test the algorithm, Pressman [11] defines Cyclomatic Complexity as a software matrix that provides quantitative measurements of the logical complexity of a program. This method determines the number of independent paths and provides a limit on the number of tests to be performed to ensure all statements have been executed at least once. Preferably, the Cyclomatic Complexity value is limited to 10 because the more complex the module, the more likely error will occur, the more difficult it is to understand, tested and modified [12].

From those statements above, the aim of this study is to develop an algorithm that can provide harakat automatically in Arabic script without harakat. Furthermore, Cyclomatic Complexity Testing will be utilized to examine the accuracy of the algorithm.

AUTOMATIC HAKARAT IN ARABIC SCRIPT

The research method starts from understanding the basic rules of the Nahwu-Sharf Science (Arabic Grammar) of giving automatic harakat in Arabic script without harakat, translating the rules into algorithm and

finally, testing the algorithm whether it is in accordance with the rules or not (see Figure 1)



Figure 1. Research Method

Data Collection Procedure

Some interviews with an Arabic Expert have been conducted to collect the data to understand the basic rules of Nahwu-Sharf Science (Arabic Grammar) of giving automatic harakat. Based on the interview results, the types of basic sentences in the Arabic language can be classified into some conditions as shown in the Table 1.

Table 1. The basic rules of Arabic Grammar

No	Condition	Example Sentences	Expected Result
1	1. If the first word is a conjunction which is not لم or لما or ل or لا 2. The second word should be <i>majrur</i> 3. Then the third word should be <i>marfu'</i> 4. Then the forth word should be <i>marfu'</i>	في الفصل تلميذ جميل	في الفصل تلميذ جميل
2	1. if the first word is a conjunction which can be لم or لما 2. The second word should be <i>majzum</i> 3. Then the third word should be <i>marfu'</i> 4. And the forth word should be <i>manshub</i>	لم يأكل محمد الرز	لم يأكل محمد الرز
3	1. if the first word is a conjunction which is لا 2. The second word should be <i>majzum</i> 3. Then the third word is <i>manshub</i>	لا تنس المعروف	لا تنس المعروف
4	1. If the first word is <i>marfu'</i> 2. The second word should be <i>marfu'</i> 3. IF the third word is not a	كتب أحمد درس	كتب أحمد درس

conjunction, then it is a *manshub*

- | | | | |
|---|---|-----------------------|-----------------------|
| 5 | 1. If the first word is <i>marfu'</i> and the second word is a conjunction
2. Then the third word should be <i>majrur</i> | القلم علي المكتب | القلم علي المكتب |
| 6 | 1. If the first word is <i>marfu'</i>
2. Then the second word should be <i>marfu'</i>
3. And if the third word is a conjunction, then the forth should be <i>majrur</i> | يذهب أحمد إلى المدرسة | يذهب أحمد إلى المدرسة |

Translating the Rules into Algorithm

There are steps performed to translate the rules into algorithm which are design database and create flowchart of giving automatic harakat.

Database Design

Two tables were designed in the database to store “dictionary” and “conjunction”. ‘Dictionary’ table stores the words without harakat and the words with their harakat based on I’rab rules, which are: marfu, mansub, majzum and majrur (See Table 2). While, ‘Conjunction’ table is created to store the conjunction of the sentence. ‘Conjunction’ table consists of two fields, which are the conjunction word without harakat (field word) and its word with harakat (field *huruf*). (See Table 3).

Table 2. Table Structure for Table ‘Dictionary’

Column Name	Type	Null	Default
idDictionary	Integer	No	
word	Varchar(25)	No	
marfu	Varchar(25)	No	
Manshub	Varchar(25)	No	
Majzum	Varchar(25)	No	
Majrur	Varchar(25)	No	

Table 3: Table Structure for Table ‘Conjunction’

Column Name	Type	Null	Default
idHuruf	Integer	No	
word	Varchar(25)	No	
huruf	Varchar(25)	No	

The Flowchart of Giving Automatic Harakat

Figure 2 shows the flowchart of giving automatic algorithm on Arabic Script.

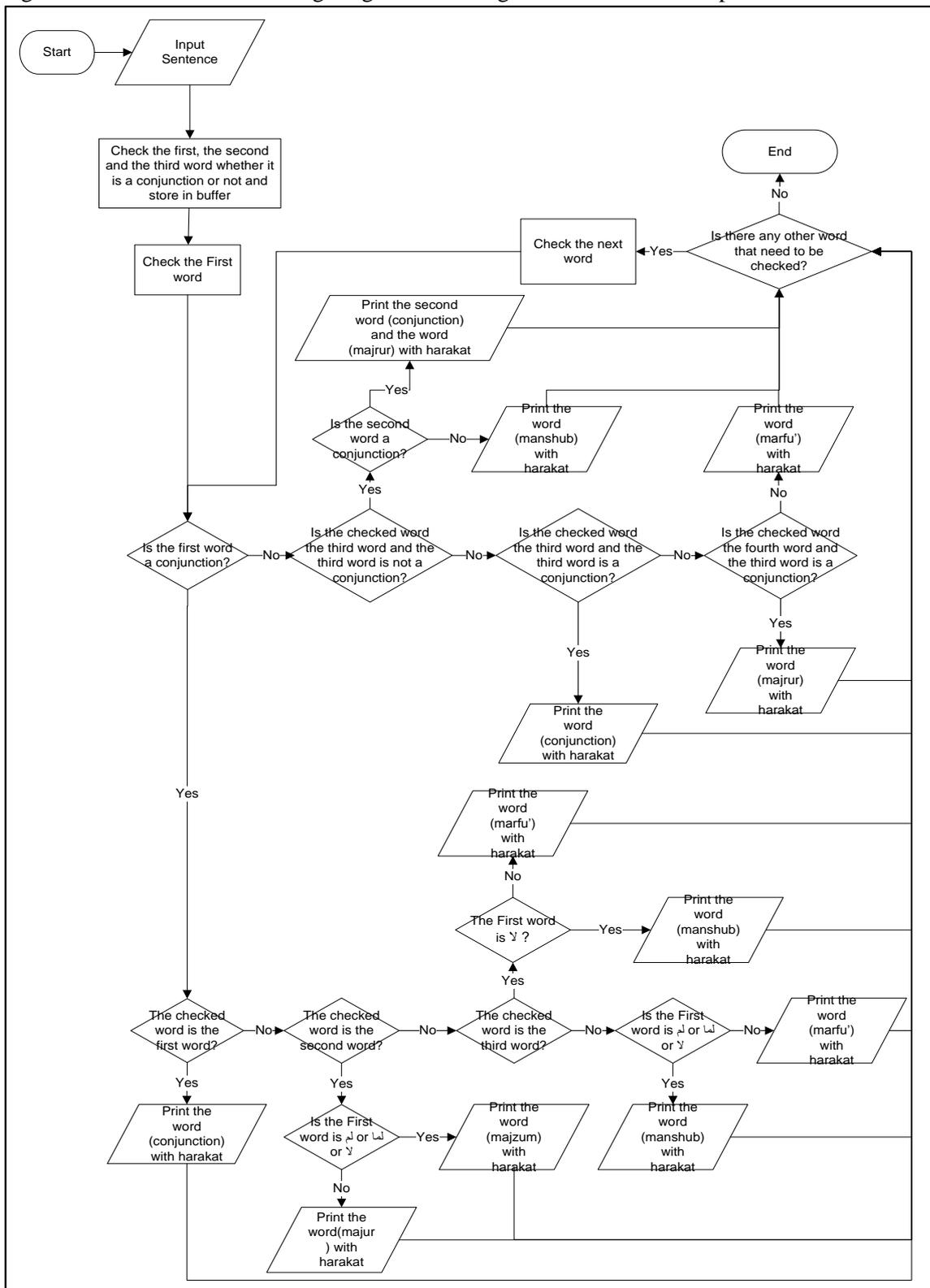


Figure 2. Flowchart of Automatic Giving Harakat on Arabic Script

The Algorithm of Giving Automatic Harakat

The finding of this research is the algorithm that was designed based on the conditions explained in Table 1. Figure 3 shows the algorithm to give harakat automatically in Arabic script without harakat.

The explanation of the algorithm from Figure 3 is that:

Number 1 to 2 (refer to the number written on the left in Figure 3): firstly, user will input a sentence into the system. One sentence consists of a maximum 4 words. System will check the first, the second and the third word whether it is a conjunction or not and save them as variables. Then, system will check word by word starting from the first.

Number 3 to 5: System will check the database whether the first word is a conjunction. If yes, system will continue checking whether the word being checked is the first word. If yes, then the system will print the word (conjunction) with harakat.

Number 6 to 9: If the checked word is not the first word, system will check whether the word being checked is the second word. If yes, then system will continue checking whether the first word is *لما* or *ولما* or *ولا* or *ل*. If yes, then the system will print the word (*majzum*) with harakat. If no, then the system will print the word (*majrur*) with harakat.

Number 10 to 13: If the checked word is not the second word, system will check whether the word being checked is the third word. If yes, then system will continue checking whether the first word is *لا*. If yes, then the system will print the word (*manshub*) with harakat. If no, then the system will print the word (*marfu'*) with harakat.

Number 14 to 17: If the checked word is also not the third word, then system will check whether the first word is *لما* or *ولما* or *ولا* or *ل*. If yes, then the system will print the word (*manshub*) with harakat. If no, then the system will print the word (*marfu'*) with harakat.

Number 18 to 21: If the first word is not a conjunction, then system will check whether the checked word is the third word and the third word is not a conjunction. Then, system will continue checking whether the second word is a conjunction. If yes, then system will print the word (*majrur*) with harakat. If no, system will print the word (*manshub*) with harakat.

Number 22 to 23: If the checked word is the third word and the third word is a conjunction, then system will print the word (conjunction) with harakat.

Number 24 to 25: If the checked word is not the third word and the third word is not a conjunction, then system will check whether the checked word is the fourth word and the third word is a conjunction. If yes, system will print the word (*majrur*) with harakat. If no, then system will print the word (*marfu'*) with harakat.

Number 26 to 28: Other than above conditions mentioned, means that the checked word is *marfu'*, then the system will print the word (*marfu'*) with harakat. Finally, after finish checking all the words one by one, the system will display the sentence with harakat.

```

Function: automaticHarakat
1:  read sentence;
   breaks a string (from the sentence) into an array;
   check the first, the second and the third word, whether it is a
   conjunction or not and store in buffer;
2:  for array i=0; while i < total words; increment i;
   read word [i];
3:  if the first word is a conjunction
4:  if (i==0)
5:  check 'huruf' table
   then print the word (conjunction) with harakat
6:  elseif (i==1)
7:  if (First word is ل or و or لا)
8:  check 'dictionary' table
   then print the word (majzum) with harakat
9:  else
   Check 'dictionary' table
   Then print the word (majrur) with harakat
10: elseif (i==2)
11: If (First word is لا)
12: check 'dictionary' table
   then print the word (manshub) with harakat
13: Else
   Check 'dictionary' table
   Then print the word (marfu') with harakat
14: Else
15: If (First word is ل or و or لا)
16: Check 'dictionary' table
   Then print the word (manshub) with harakat
17: Else
   check 'dictionary' table
   then print the word (marfu') with harakat
   Endif
18: elseif ((i==2) and the third word is not a conjunction)
19: if the second word is a conjunction
20: Check 'dictionary' table
   Then print the second word (conjunction) and the
   word (majrur) with harakat
21: Else
   check 'dictionary' table
   Then print the word (manshub) with harakat
22: elseif ((i==2) and the third word is a conjunction)
23: Check 'huruf' table
   Then print the word (conjunction) with harakat
24: elseif ((i==3) and the third word is a conjunction)
25: check 'dictionary' table
   Then print the word (majrur) with harakat
26: else
   Check 'dictionary' table
   Then print the word (marfu') with harakat
27: Endif
28: Endfor
   Display sentence with harakat

```

Figure 3. The Algorithm of Giving Automatic Harakat.

The Algorithm Testing

The algorithm testing has been done by utilizing Cyclomatic Complexity White Box Testing. There are three steps in this testing method, as follows:

1) Create the Flowgraph

Figure 4 shows the flowgraph of the algorithm. The number of the node in the flowgraph refer to the number in the left from algorithm in Figure 3 (number 1 until 28).

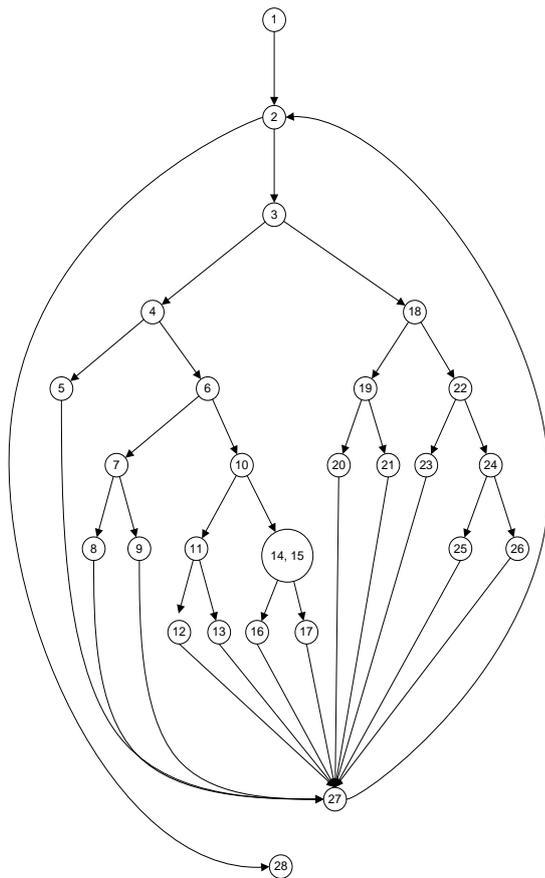


Figure 4. Flowgraph

2) Calculate the Independent Path

Total of Region = 13
 $V(G) = \text{Edge} - \text{Node} + 2$
 $V(G) = 38 - 27 + 2 = 13$
 $V(G) = P + 1$
 $V(G) = 12 + 1 = 13$

So, there will be 13 independent paths that need to be examined which is shown in Table 3.

Table 3: Independent Path

Path No.	Independent Path
1	1-2-3-4-5-27-2-28
2	1-2-3-4-6-7-8-27-2-28
3	1-2-3-4-6-7-9-27-2-28
4	1-2-3-4-6-10-11-12-27-2-28
5	1-2-3-4-6-10-11-13-27-2-28
6	1-2-3-4-6-10-14,15-16-27-2-28
7	1-2-3-4-6-10-14,15-17-27-2-28
8	1-2-3-18-19-20-27-2-28
9	1-2-3-18-19-21-27-2-28
10	1-2-3-18-22-23-27-2-28
11	1-2-3-18-22-24-25-27-2-28
12	1-2-3-18-22-24-26-27-2-28
13	1-2-28

3) Create Test Case

The independent path obtained in Table 3 will be examined using a test case scenario. Based on the interview results with experts conducted earlier and have been explained also in Table 1, there are six examples of sentences that will be used for the test case scenario. See Table 4.

Table 4. Sample Sentences

No	Input Sentence	Expected Result
1	في الفصل تلميذ جميل	فِي الْفَصْلِ تَلْمِيذٌ جَمِيلٌ
2	لم يأكل محمد الرز	لَمْ يَأْكُلْ مُحَمَّدٌ الرُّزَّ
3	لا تنس المعروف	لَا تَنْسَ الْمَعْرُوفَ
4	كتب أحمد درس	كَتَبَ أَحْمَدُ الدَّرْسَ
5	القلم علي المكتب	الْقَلَمُ عَلَى الْمَكْتَبِ
6	يذهب أحمد إلى المدرسة	يَذْهَبُ أَحْمَدُ إِلَى الْمَدْرَسَةِ

Test case of the algorithm can be seen in Table 5.

Table 5: Test Case of the Algorithm

Example No.	Input	Path No.	Independent Path	Expected Result (Output)	Actual Result	
1	في الفصل تلميذ جميل			System will display: في الفصل تلميذ جميل	As expected	
		في	1	3: if the first word is a conjunction 4:if (i==0) 5: check 'huruf' table then print the word (conjunction) with <i>harakat</i>	System will read the word from database: في	As expected
		الفصل	3	6: elseif (i==1) 7: if (First word is لم or لما or لا) 9: else Check 'dictionary' table. Then print the word (<i>majrur</i>) with harakat	System will read the word from database: الفصل	As expected
		تلميذ	5	10: elseif (i==2) 11: if (First word is لا) 13: Else Check 'dictionary' table Then print the word (<i>marfu</i>) with harakat	System will read the word from database: تلميذ	As expected
		جميل	7	14,15: Else If (First word is لم or لما or لا) 17: Else check 'dictionary' table then print the word (<i>marfu</i>) with harakat	System will read the word from database: جميل	As expected
	If (no more words need to be checked)	13	28: endfor display sentence with <i>harakat</i>	Looping stop, then display in the system all of the words with harokah into a sentence	As expected	
2	لم يأكل محمد الرز			System will display: لَمْ يَأْكُلْ مُحَمَّدٌ الرُّزَّ	As expected	
		لم	1	3: if the first word is a conjunction 4:if (i==0) 5: check 'huruf' table then print the word (conjunction) with <i>harakat</i>	System will read the word from database: لَمْ	As expected

Table 5: Test Case of the Algorithm (contd.)

Example No.	Input	Path No.	Independent Path	Expected Result (Output)	Actual Result
3	يَأْكُل	2	6: elseif (i==1) 7: if (First word is لم or لما or لا) 8: check 'dictionary' table then print the word (<i>majzum</i>) with harakat	System will read the word from database: يَأْكُل	As expected
	مَحْمَد	5	10: elseif (i==2) 11: If (First word is لا) 13: Else Check 'dictionary' table Then print the word (<i>marfu</i>) with harakat	System will read the word from database: مُحَمَّدٌ	As expected
	الرَّزْز	6	14: Else 15: If (First word is لم or لما or لا) 16: Check 'dictionary' table Then print the word (<i>manshub</i>) with harakat	System will read the word from database: الرَّزْزُ	
	If (no more words need to be checked)	13	27: endfor display sentence with <i>harakat</i>	Looping stop, then display in the system all of the words with harokah into a sentence System will display: لَا تَتَسَّنِ الْمَعْرُوفَاتُ	As expected
4	لا تنس المعروف	1	3: if the first word is a conjunction 4:if (i==0) 5: check 'huruf' table then print the word (conjunction) with <i>harakat</i>	System will read the word from database: لَا	
	تنس	2	6: elseif (i==1) 7: if (First word is لم or لما or لا) 8: check 'dictionary' table then print the word (<i>majzum</i>) with harakat	System will read the word from database: تَتَسَّنِ	
	المعروف	4	10: elseif (i==2) 11: If (First word is لا) 12: check 'dictionary' table then print the word (<i>manshub</i>) with harakat	System will read the word from database: المَعْرُوفَاتُ	
	كتب أحمد الدرس			System will display: كَتَبَ أَحْمَدُ الدَّرْسَينِ	As expected

Table 5: Test Case of the Algorithm (contd.)

Example No.	Input	Path No.	Independent Path	Expected Result (Output)	Actual Result
	كتب	12	3: if the first word is a conjunction 18: elseIf ((i==2) and the third word is not a conjunction) 22: elseif ((i==2) and the third word is a conjunction) 24: elseif ((i==3) and the third word is a conjunction) 26: else Check 'dictionary' table Then print the word (<i>marfu</i>) with harakat	System will read the word from database: كَتَبَ	As expected
	أحمد	12	3: if the first word is a conjunction 18: elseIf ((i==2) and the third word is not a conjunction) 22: elseif ((i==2) and the third word is a conjunction) 24: elseif ((i==3) and the third word is a conjunction) 26: else Check 'dictionary' table Then print the word (<i>marfu</i>) with harakat	System will read the word from database: أَحْمَدُ	As expected
	الدرس	9	18: elseIf ((i==2) and the third word is not a conjunction) 19: if the second word is a conjunction 21:Else check 'dictionary' table Then print the word (<i>manshub</i>) with harakat	System will read the word from database: الدَّرْسِ	As expected
	If (no more words need to be checked)	13	27: endfor display sentence with <i>harakat</i>	Looping stop, then display in the system all of the words with harokah into a sentence	As expected
5	القلم علي المكتب			System will display: القَلَمُ عَلَي المَكْتَبِ	As expected
	القلم	12	3: if the first word is a conjunction 18: elseIf ((i==2) and the third word is not a conjunction) 22: elseif ((i==2) and the third word is a conjunction) 24: elseif ((i==3) and the third word is a conjunction) 26: else Check 'dictionary' table Then print the word (<i>marfu</i>) with harakat	System will read the word from database: القَلَمُ	As expected

Table 5: Test Case of the Algorithm (contd.)

Example No.	Input	Path No.	Independent Path	Expected Result (Output)	Actual Result
	علي	-	-	System will read the word from database: -	As expected
	المكتب	8	18: elseif ((i==2) and the third word is not a conjunction) 19: if the second word is a conjunction 20: Check 'dictionary' table Then print the second word (conjunction) and the word (<i>majrur</i>) with harakat	System will read the word from database: عَلِي الْمَكْتَبِ	As expected
6	يذهب أحمد إلي المدرسة			System will display: يَذْهَبُ أَحْمَدُ إِلَى الْمَدْرَسَةِ	As expected
	يذهب	12	3: if the first word is a conjunction 18: elseif ((i==2) and the third word is not a conjunction) 22: elseif ((i==2) and the third word is a conjunction) 24: elseif ((i==3) and the third word is a conjunction) 26: else Check 'dictionary' table Then print the word (<i>marfu'</i>) with harakat	System will read the word from database: يَذْهَبُ	As expected
	أحمد	12	3: if the first word is a conjunction 18: elseif ((i==2) and the third word is not a conjunction) 22: elseif ((i==2) and the third word is a conjunction) 24: elseif ((i==3) and the third word is a conjunction) 26: else Check 'dictionary' table Then print the word (<i>marfu'</i>) with harakat	System will read the word from database: أَحْمَدُ	As expected
	إلي	10	22: elseif ((i==2) and the third word is a conjunction) 23: Check 'hurf' table Then print the word (conjunction) with harakat	System will read the word from database: إِلَى	As expected
	المدرسة	11	24: elseif ((i==3) and the third word is a conjunction) 25: check 'dictionary' table Then print the word (<i>majrur</i>) with harakat	System will read the word from database: الْمَدْرَسَةِ	As expected
	If (no more words need to be checked)	13	27: endfor display sentence with <i>harakat</i>	Looping stop, then display in the system all of the words with harokah into a sentence	As expected

Furthermore, 24 sentences have been added to be examined for more detailed analysis.

Table 6: More Samples Examination

Condition Number	Input Sentence	Expected Result	Actual Result
1	في الفصل تلميذ جميل	فِي الْفَصْلِ تَلْمِيذٌ جَمِيلٌ	فِي الْفَصْلِ تَلْمِيذٌ جَمِيلٌ
	في البستان الكلب ينام	فِي الْبُسْتَانِ الْكَلْبُ يَنَامُ	فِي الْبُسْتَانِ الْكَلْبُ يَنَامُ
	في النار الحديد يذوب	فِي النَّارِ الْحَدِيدُ يَذُوبُ	فِي النَّارِ الْحَدِيدُ يَذُوبُ
	على الأرض الثمر يتساقط	عَلَى الْأَرْضِ الثَّمَرُ يَتَساقَطُ	عَلَى الْأَرْضِ الثَّمَرُ يَتَساقَطُ
	على الشجرة العصفور يغرد	عَلَى الشَّجَرَةِ الْعَصْفُورُ يُغْرِدُ	عَلَى الشَّجَرَةِ الْعَصْفُورُ يُغْرِدُ
2	لم يذهب تلميذ	لَمْ يَذْهَبْ تَلْمِيذٌ	لَمْ يَذْهَبْ تَلْمِيذٌ
	لم يأكل محمد الرز	لَمْ يَأْكُلْ مُحَمَّدٌ الرَّزَّ	لَمْ يَأْكُلْ مُحَمَّدٌ الرَّزَّ
	لم يجلس الضيف	لَمْ يَجْلِسْ الضَّيْفُ	لَمْ يَجْلِسْ الضَّيْفُ
	لم يكتب التلميذ درس	لَمْ يَكْتُبِ التَّلْمِيذُ الدَّرْسَ	لَمْ يَكْتُبِ التَّلْمِيذُ الدَّرْسَ
	لم يحضر محمد	لَمْ يَحْضُرْ مُحَمَّدٌ	لَمْ يَحْضُرْ مُحَمَّدٌ
3	لا تشرب الماء	لَا تَشْرَبِ الْمَاءَ	لَا تَشْرَبِ الْمَاءَ
	لا تنس المعروف	لَا تَنْسِ الْمَعْرُوفَ	لَا تَنْسِ الْمَعْرُوفَ
	لا تكتب درس	لَا تَكْتُبِ الدَّرْسَ	لَا تَكْتُبِ الدَّرْسَ
	لا تأكل الخبز	لَا تَأْكُلِ الْخُبْزَ	لَا تَأْكُلِ الْخُبْزَ
	لا تضرب الكلب	لَا تُضْرِبِ الْكَلْبَ	لَا تُضْرِبِ الْكَلْبَ
4	كتب أحمد درس	كَتَبَ أَحْمَدُ الدَّرْسَ	كَتَبَ أَحْمَدُ الدَّرْسَ
	يداعب إسماعيل القط	يُدَاعِبُ إِسْمَاعِيلَ الْقَطَّ	يُدَاعِبُ إِسْمَاعِيلَ الْقَطَّ
	يحصد الفلاح القمح	يَحْصُدُ الْفَلَّاحُ الْقَمْحَ	يَحْصُدُ الْفَلَّاحُ الْقَمْحَ
	يحب الولد البريق	يُحِبُّ الْوَلَدُ الْبَرِيقَ	يُحِبُّ الْوَلَدُ الْبَرِيقَ
	يحترق الحطب	يَحْتَرِقُ الْحَطْبُ	يَحْتَرِقُ الْحَطْبُ
5	الطائر فوق الشجرة	الطَّائِرُ فَوْقَ الشَّجَرَةِ	الطَّائِرُ فَوْقَ الشَّجَرَةِ
	العصفور في القفص	العَصْفُورُ فِي الْقَفْصِ	العَصْفُورُ فِي الْقَفْصِ
	المداد في الدواة	الْمِدَادُ فِي الدَّوَاةِ	الْمِدَادُ فِي الدَّوَاةِ
	القمر في السماء	الْقَمَرُ فِي السَّمَاءِ	الْقَمَرُ فِي السَّمَاءِ
	الطفل على السرير	الطِّفْلُ عَلَى السَّرِيرِ	الطِّفْلُ عَلَى السَّرِيرِ

Table 6: More Samples Examination (contd.)

Condition Number	Input Sentence	Expected Result	Actual Result
6	يذهب أحمد إلى المدرسة	يَذْهَبُ أَحْمَدُ إِلَى الْمَدْرَسَةِ	يَذْهَبُ أَحْمَدُ إِلَى الْمَدْرَسَةِ
	يذهب أحمد إلى المدرسة	يَذْهَبُ أَحْمَدُ إِلَى الْمَدْرَسَةِ	يَذْهَبُ أَحْمَدُ إِلَى الْمَدْرَسَةِ
	سمعت النصيحة من الأستاذ	سَمِعْتُ النَّصِيحَةَ مِنَ الْأَسْتَاذِ	سَمِعْتُ النَّصِيحَةَ مِنَ الْأَسْتَاذِ
	يسطع الثور في الحجرة	يَسْطَعُ الثَّوْرُ فِي الْحَجْرَةِ	يَسْطَعُ الثَّوْرُ فِي الْحَجْرَةِ
	تجري السفينة على الماء	تَجْرِي السَّفِينَةُ عَلَى الْمَاءِ	تَجْرِي السَّفِينَةُ عَلَى الْمَاءِ

System Implementation

Figure 5 shows the interface of user where user can input a sentence without harakat. After clicking the 'Put Harakat' button, user can see the sentence with harakat.



Figure 5. System Implementation

RESULT AND DISCUSSION

The results of the test case from Table 5 and Table 6 match the expected output (as written 'as expected' in Table 5). Furthermore, twenty-four examples (Table 6) have been examined and also match the expected output.

Initially this research refers to the research conducted by Sembok [13] & Ata and Hadni et al [14], where Arabic Stemming and Arabic Text Mining give considerations how to provide harakat to Arabic sentence by searching the basic word of each words to determine the end of the harakat of the word based on Arabic grammar. However, as the study progresses, the process of searching the basic word is perceived to make the algorithm works too long and ineffective. Hence, the author insists on applying the Nahwu-Sharf science not to the code (by searching for the base word of each words, then determines the appropriate harakat at the end of the letter of the word), but apply the Arabic Grammar (Nahwu-Sharf science) to the database. Where is in the database, words are separated according to the rules, which are: marfu',

manshub, majzum, majrur and the word without harakat. And this method result accuracy 100% in accordance with the rules of Nahwu-Sharf science.

Besides, the value of Cyclomatic Complexity obtained is thirteen which is more than 10. Kanjilal [15] stated that if the complexity cyclomatic of the algorithm is less than or equal to 10, the algorithm is assumed to be well-structured, highly testable and maintainable. Kanjilal [15] also suggested to use Polymorphism which is able to replace the conditional constructs of code/algorithm to reduce the cyclomatic complexity of the algorithm.

REFERENCES

- [1] A. Sirajudeen and A. Adebisi, "Teaching Arabic as a second language in Nigeria," *Procedia-Soc. Behav. Sci.*, vol. 66, pp. 126–135, 2012.
- [2] Chick, A.R., "Al-Alfaz al-arabiyyah al-muqtaradhah fi al-lughah al-malayuwiiyah: 'ardhunwatahlil," 2011. [Online]. Available: http://portal.arabtime.com/article_preview.cfm?Action=Article&Preview=ViewOnly&ArticleID=21827. [Accessed: 10-Nov-2017].
- [3] R. L. Oxford, *Language Learning Styles and Strategies*. Mouton de Gruyter, 2003.
- [4] Greene, R., "Why learning Arabic is so hard.," 2017. [Online]. Available: http://www.slate.com/articles/arts/culturebox/2005/06/im_trying_to_learn_arabic.html. [Accessed: 10-Nov-2017].
- [5] M. Abdurrahman, "Rule of Nahwiyah Variations in Arabiyah Bayna Yadaik Book and its Contribution on Arabiyah Asasiyah Subject," *J. Educ. Learn. EduLearn*, vol. 9, no. 3, pp. 256–266, 2015.
- [6] Shanhaji, A., *Al Imam Matn al jurmiyah*. Jakarta: Al Idrus, 2000.
- [7] "Madinah Arabic Learn Arabic For Free," 2017.[Online].Available:<http://www.madinaharabic.com/articles/the-story-of-irab.html>. [Accessed: 10-Nov-2017].
- [8] "Basic Arabic Grammar (Nahu)," *Ummah.com - Muslim Forum*, 2017.[Online].Available:<https://www.ummah.com/forum/forum/library/learn-arabic-and-other-languages/218451-basic-arabic-grammar-nahu>. [Accessed: 10-Oct-2017].
- [9] Ö. Ergül, *Guide to programming and algorithms using R*. Springer, 2013.
- [10] A. A. Puntambekar, *Analysis of algorithm and design*. Technical Publications, 2009.
- [11] R. S. Pressman, *Software Engineering*. Palgrave Macmillan, 2001.
- [12] "Cyclomatic Complexity," 2017. [Online]. Available:<http://www.math.unipd.it/~tullio/IS-1/2004/Approfondimenti/cyclomatic.htm>. [Accessed: 10-Oct-2017].
- [13] T. M. T. Sembok and B. A. Ata, "Arabic word stemming algorithms and retrieval effectiveness," in *Proceedings of the World Congress on Engineering*, 2013, vol. 3, pp. 3–5.
- [14] M. Hadni, S. A. Ouatik, and A. Lachkar, "Effective Arabic stemmer based hybrid approach for Arabic text categorization," *Int. J. Data Min. Knowl. Manag. Process*, vol. 3, no. 4, p. 1, 2013.
- [15] Kanjilal, J., "How to analyze the cyclomatic complexity in your code | InfoWorld," 2017.[Online].Available:<https://www.infoworld.com/article/3097404/application-development/how-to-analyze-the-cyclomatic-complexity-in-your-code.html>. [Accessed: 10-Oct-2017].

CONCLUSION

Since the entered word is according to the conditions of the algorithm (refer to Table 1), the results of the test case match the expected output. Thus, it can be concluded that the generated algorithm has been proven in accordance with the rules of Nahwu-Sharf science through Cyclomatic Complexity testing. This algorithm is expected to be implemented on a development system that can assist students and institutions who learn Arabic to automatically generate harakat based on user input.