A Novel Algorithm for the Comparison of Bangla Strings for Sorting According to the Rules of Bangla Academy

Asif Mohammed Samir
Lecturer,
Institute of Information and Communication Technology,
Shahjalal University of Science and Technolgy,
Sylhet-3114, Bangladesh

Md. Ruhul Amin
Assistant Professor,
Department of Computer Science and Engineering,
Shahjalal University of Science and Technology,
Sylhet-3114, Bangladesh

ABSTRACT
In this paper we discuss the comparison of two strings of Bangla language represented by Unicode character set. This comparison order maintains the Bangla Academy rule. This method can be used to sort Bangla words perfectly A few works has been done on this topic but no standard is set up yet to sort Bangla words. Some of these works are based on ASCII representation. As a part of internationalization, Unicode representation is much more preferable than ASCII representation. We discussed an easy way to compare not only the Bangla words but also the Unicode Bangla strings. In our method, a mapping is used which simplified the sorting procedure. This method can compare any Unicode Bangla strings and it is not keyboard dependent.

Keywords
Bangla String Comparison, Bangla Word Sorting, Unicode Bangla Sorting, Bangla Text Sorting

1. INTRODUCTION
Bangla is an eastern Indo-Aryan Language. It is the native language of Bangladesh, the Indian state of West Bengal and parts of the Indian states of Tripura and Assam. It is written with the Bangla script. [1] About 181 million people are the native speaker of this language and nearly 250 million people can speak Bangla in total. It is one of the most spoken languages (ranking sixth) all over the world. [2] It is the national and official language of Bangladesh and one of the 23 official languages recognized by the Republic of India [1]. In order to honor Bangladesh, Bangla is declared as one of the official languages of Sierra Leone also.

As the Bangla language is a rich and widely used language, it must have some standardization such as Bangla keyboard layout, Bangla character recognition, voice synthesis etc. But unfortunately we have advanced a very little in this regard. In a rapidly developing environment of computerization of Bangla language, one of the most important issues is Bangla text sorting. For the development of Bangla database systems, an efficient, versatile sorting algorithm is a must. The problem is not with the sorting algorithm rather it is with how the Bangla strings are compared to maintain the right order. There are some papers on this topic but none of them could set standard for sorting Bangla text. None of the papers maintain sort Bangla strings in proper order. In this paper, we have shown the analysis of the previously proposed sorting algorithms and the comparison among the procedures to represent drawbacks, difficulties and limitations. Based on these observations we have proposed an algorithm based on Unicode to sort Bangla strings accurately, and the complexity is satisfying. The proposed algorithm is readable and very easy to code; hence it has the potential to be considered as standard algorithm for sorting Bangla strings. As Bangla Academy [3] is the national academy for promoting Bangla language in Bangladesh, we are following the Bangla Academy dictionary standard for our proposed method.

2. THE BANGLA LANGUAGE
Base Letters: In the written form of Bangla alphabets, there are 11 vowels and 39 consonants. When we use these alphabets, we call it base letters.
The vowels are -

অ আ ই ঈ উ ঊ ঋ এ ঐ ও ঔ

The consonants are -

ক খ গ ঘ ঙ চ ছ জ ঝ ঞ ট ঠ ড ঢ ণ ত থ দ ধ ন প ফ ব ভ ম য র ল

শ ষ স হ ড় ঢ় য় ৎ ং ঃ ঁ

These are the base letters of Bangla language.
Modifiers: There two types of modifiers in Bangla, vowel modifiers and the consonant modifiers.
10 of the 11 vowels can be used as modifier to the consonants. We call them vowel modifier. They can never be used independently. Here is the list of vowel modifier:

Word	Vowel Modifier	Example
অ	-	কলম /kɔlom/
আ	া	কলাম /kɔlam/
ই	ি	পিঠা /piʈʰa/
ঈ	ী	জীবন /ʤībon/
উ	ু	তুলা /tula/
ঊ	ূ	সূচী /suci/
ঋ	ৃ	বৃষ্টি /briʃʈi/
এ	ে	কেমন /kæmon/
ঐ	ৈ	হৈম /hojmo/
ও	ো	কোমল /komol/
ঔ	ৌ	শৌখিন /ʃowkʰin/

Table-01: Vowel Modifiers

Like the vowel modifiers, the consonants have some short forms when they used with other consonant. They are called –ফলা. Some of them are given below:

Word	Consonant Modifier	Example
ব	ব-ফলা	জ্বর /ʤɔr/

য	য-ফলা	জন্য /ʤonno/
র	র-ফলা	তীব্র /tibro/

Table-02: Consonant Modifiers

Compound Characters: When two or more consonant characters of Bangla alphabet used together, then they are called the compound characters. There are about 270 compound characters in Bangla. Some examples of compound characters are given below:

Word	Compound Character	Decompressed Form	No. of Alphabet Used
উজ্জ্বল /ujjbol/	জ্জ্ব	জ + জ + ব	3
বৃষ্টি /briʃʈi/	ষ্ট	ষ + ট	2
যুদ্ধ /ʤuddho/	দ্ধ	দ + ধ	2
ব্রাহ্মন /brɑmmon/	হ্ম	হ + ম	2

Table-03: Compound Characters

3. DIFFICULTIES OF SORTING BANGLA TEXT

The problems associated with sorting of Bangla words are as follows-

- Bangla words should be sorted according to the Bangla Academy [4] standard. But unfortunately the Unicode for Bangla characters are not in Bangla Academy dictionary order. So, mapping is required to sort words correctly.
- Compound characters (জ + ্ +জ + ্ +ব = জ্জ্ব, জ + ্ + জ = জ্জ) make Bangla sorting complicated.
- In writing, vowel modifiers (ে +ক = েক, ক + া = কা) can precede or follow the base letter in Bangla words, but in computation it should be placed after the base letter for proper sorting.
- Unicode characters র, য়, ঢ়, ড় can be written in two ways. For example, ঢ় can be a single character ঢ় (\u09DD) or compound of ঢ + ় (\u09A2+\u09BC). These two cases should be considered as special case while sorting.
- Bangla characters in the Unicode chart are not aligned to be sorted.

4. PREVIOUS WORKS

Method 1

M. Shahidur Rahman et al. [5] have proposed an alternative representation during computation. According to their proposal a dummy character is placed after the character, which does not have any modifier. Moreover, it is also considered that there would be no dummy character between the constituent parts of a compound character. Generally vowel modifiers can be typed before or after the characters but for this algorithm the modifiers are shifted after the character for the internal representation while computing. In case of compound characters, they are decomposed into their constituent components and stored accordingly. In Table-4 internal representation of few words are shown where <space> represents the dummy character. To sort the words the relative order in the character set are arranged in the following way-

Null modifier < Vowel Modifiers < Vowels < Consonants

Input Word	Internal input Representation	Internal Representation of Sorted Output Word
কুসুম	ক ু স ু ম	ক ম ল া
নিলয়	ন ি ল য়	ক ু স ু ম
মৃণাল	ম ৃ ণ া ল	খ ো ক ন

Table-04: Internal representation Of Words In [5]

This method has the following drawbacks:

- This work is based on ASCII based Bangla words.
- আ is considered as concatenation of অ and া, which is not valid according to the Unicode consortium.
- Provides incorrect result for Bangla string sorting. (E.g. Provided incorrect result- কীএ, কীি)

Method 2

According to Mafizul Haque Khan et al.'s "An Efficient And Correct Bangla Sorting Algorithm" **[6]** a character is represented with two digit unique number for every letter of Bangla alphabet along with the vowel modifiers and the consonant modifiers. The letters and their corresponding numbers are given in Table-05. It is to be noticed that here 'আ' is treated as a set of two characters that is 'অ + া '. The consonant modifiers are having the same number as their original consonants.

Character	Number
অ, ই-ঔ, ং, ঃ, ঁ	11-23
ক-ঙ, চ-ঞ	25-34
ট, ঠ, ড, ড়, ঢ, ঢ়, ণ , ৎ	35-42
ত, থ, দ, ধ, ন, প, ফ, ব, ভ, ম	43-52
য, য়, হ, র, ৵, ল, শ, ষ, স	53-61
া, ি, ী, ু, ূ, ৃ, ে, ৈ, ো, ৌ	71-80

Table-05: Representing A Bangla Character Of Two Digit

For vowel modifiers, wherever its position is (i.e. Left, Right or Down), its corresponding number will always be placed after the number of the letter over which it was applied to form larger number. For Example, the words সা, সি, সু, সো, and সে change into numbers '6171','6174','6172','6179' and '6177'. For Compound characters 99 is placed between the number representations of two constituent characters. For example, for the word- ক্ক = 'ক + ্ + ক', the number will be 259925. For সহ্য, it will be - 60619953. Also, to compare words of different length, extra zeros are appended at the end of shorter word. The difference of number of digit between the words গোধূলি (277946755773) and সংস্কৃতি (6021609925764372) is 4. So according to the algorithm, four zero's are appended at the end of the number representing গোধূলি. So finally the number becomes 2779467557730000.

Drawbacks-

- আis considered only as a compound character of অ + া, Unicode does not support অ + া,

- Adding extra zero's at the end of the number with less digit which increases overhead.
- 64 bit integer can hold up to 20 digits, but for example- '*কিংকর্তব্যবিমূঢ়*' requires 30 digits, which requires string comparison rather than sorting integer numbers.
- Does not provide correct result for Bangla string sorting. (E.g. Provided incorrect result- কীএ, কীি)

Method 3

Shah Md. Emrul Islam et al. proposed a method to Sort Unicode Bengali Text Using Ancillary Maps. [7] In this method, the Unicode characters are mapped and given a Sort Weight. The structure of the Ancillary table is like the following:

Unicode	Sort Weight	Remarks
0985	01	
0986	02	
09BE	03	RM
0987	04	
09BF	05	LM
.	.	
.	.	
09B6	56	BL
09B7	57	BL
.	.	
09FA	89	
09BD	90	

Table-06: Structure of Ancillary Maps

For each word the mapped value are concatenated and a decimal point is added after two digits from the starting. Then it becomes a floating point number. By comparing all the floating point numbers, the list of words is sorted. For example, the word কানকো (Unicode Representation 099509BE09A8099509CB) gets the value 25.0346002519. The algorithm uses the decimal number system for determination of the value of a Bangla Word. Let's consider an example of two Bangla words "কর্মচারী" and "কার্যকরভাবে" for which the decimal values become 25.005463510030035407 and 25.035463520025005400500349 15 respectively. Now it's easy to compare the two numbers. By this way a list of Bangla words can be sorted.

Drawbacks

- Adds extra complexity while converting a string to floating point number
- The range for the floating point decimal number may exceed if the length of the word is much longer which will arise round-off error
- Does not provide correct result for Bangla string sorting. (E.g. Provided incorrect result- কীএ, কীি)

Method 4

According to "A New Approach to Sort Unicode Bengali Text", a Bangla word is converted to integers of 64/128/192 bit through a complex procedure. In this approach modifiers are assigned integer number between 1-11, including ্. Characters starting from অ are assigned number multiple of 12. For example- অ is 12(=1*12), আis 24(=2*12), ই is 36(=3*12), and so on. When a modifier number is added to the previous character number, it generates a unique number. Consider a word- ক্ষতি.

ক্ষতি	ক	্	ষ	ত	ি
	180(15*12)	11	576(48*12)	396(33*12)	2
Number representation	191(=181+11)		576	398(=396+2)	

Now, these numbers are converted to their binary representation and if they are less than 10 digits, then 0's are added in front of each binary number. E.g.-

ক্ষতি	ক	্	ষ	ত	ি
	180(15*12)	11	576(48*12)	396(33*12)	2
Number representation	191(180+11)		576	398(396+2)	
Binary representation	**00**10111111		1001000000	**0**110001110	

Concatenating these binary values we get: **00**10111111 1001000000 **0**110001110**.** In the next step, if the length of the binary value is less than 64 bit, extra 0's are added to make the length of the binary string 64 bit. After adding 0's, the binary value is- **00**10111111 1001000000 **0**110001110 00000000000000000000000000000000

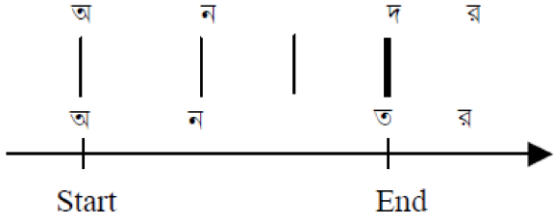
This is equivalent to hexadecimal value- 2FE4063800000000H. The binary value is then converted to integer and it is ready for sorting.

Drawbacks

- Cannot sort strings in order using described method.(Tested)
- The authors describes this process saves space, but, this process requires 3 arrays, one for the words, one for the integer, and one for the map, requires 3 arrays. This absolutely does not save space.
- Cannot sort random strings. (E.g. Provided incorrect result- কীএ, কীি)

Method 5

Sorting process described in *"Computer Representation of Bangla Characters And Sorting Of Bangla Words"* proceeds by comparing a pair of characters of same index of two words. Whenever difference is found between the pair of characters, the word which has character with higher precedence is returned smaller. For example-



According to this approach, if the pair of characters found has same precedence, then it is checked whether there are link characters followed by characters of current index. If, there are link characters or no link characters for both the words, then the process proceeds. Else, the word with link character is declared

bigger than the other one. For example, for two words সকাল, সানালী, সকাল is declared smaller সানালী.

If the comparing process ends, the word with the higher length is declared bigger than the other word. For example, for two words, সূচিতা, সূচি; সূচিতা is greater than সূচি.

Drawbacks

- Described for ASCII character set.
- Cannot sort random string. (E.g. Provided incorrect result- কীএ,কীি)

5. PROPOSED SOLUTION

As we see that not any procedure proposed to sort the Bangla strings could complete the job properly, our observation is that the lack of standard comparison method is the main reason behind the failure of all the existing procedure. So we propose a standard comparison method.

Main Process

The rules followed in our approach are-

1. Any character without any modifier is considered as character followed by null modifier.
2. Any character with vowel modifier is considered as character followed by vowel modifier.
3. Any character with consonant modifier is considered as character followed by link character followed by consonant followed by a null modifier.
4. Any compound character is considered as character followed by link character followed by character followed by null modifier.
5. Any modifier not followed by character is considered as null character followed by the modifier.

In our algorithm the precedence of the Bangla character is maintained using the following rule:

Null Character / Null Modifier < Digits < Vowel Modifier < Consonant Modifier < Vowel < Consonant

Explanation

Why characters cannot be compared to modifier and vice versa or, why do we need null modifier?

Consider two words, for example- অংশা, অংশক

অ	ং	শ	া(12)

অ	ং	শ	ক(37)

When last character of the first word is compared to last character of the second word, the first word is smaller than the second word, which is incorrect.

There can be found several words like these. For example- বাসি, বাসন or কিরন, করণীয় etc.

If we introduce null modifier, the strings will be sorted properly. For example, for the example above, we introduce null modifier below.

অ	ং	শ	া(12)	

↓

অ	ং	শ	Null Modifier(0)	ক(37)

The fourth character of the first word is compared to the null modifier for the second word. Now the strings can be sorted properly.

Why do we need Null Character?

Consider two strings- ৌাটতমি or ৌএই.

Complication arises when two modifiers come side by side. For example- ি া. Here is the explanation what happens. According to the rule, the characters are in even position and modifiers are in odd position.

ৌ	Null Modifier(0)	এ

ৌ	া(12)	ট

The second character of first string is compared to the second character of the second string, which results second string is smaller than the first one, which is incorrect.

Some other strings which cannot be sorted are- ২য়, ২২তম and কীি, কীয় and ১১, ১ম etc.

If we introduce null character for the above example keeping all other rules as it was, we see-

Null Character(0)	ৌ	Null Character(0)	া

↓

Null Character(0)	ৌ	এ (30)	ু

3rd / Null character of first string is compared to the character of the second string, and give the proper result. That is why null character is introduced.

Mapping

As stated earlier, Bangla characters are not in order in the Unicode chart. So, the characters need to be mapped in order to get the proper result. The whole mapping serial is given below-

(NULL),০,১,২,৩,৪,৫,৬,৭,৮,৯,া,ি,ী,ু,ূ,ৃ,ে,ৈ,ো,ৌ,্,অ,আ,ই,ঈ, উ,ঊ,ঋ,এ,ঐ,ও,ঔং,ঃ,ঁ,ক,খ,গ,ঘ,ঙ,চ,ছ,জ,ঝ,ঞ,ট,ঠ,ড,ঢ়,ণ,ৎ,ত,থ,দ,ধ, ন,প,ফ,ব,ভ,ম,য,য়,র,ল,শ,ষ,স,হ

Steps for Sorting Bangla Text

5.4.1 Word Sorting

Consider two words- কলম and কলাম

Computer Representation of these words-

Word: 1			
ক	ল	◻	ম

Word: 2		
ক	ল	ম

Comparison Steps

Step-1	Step-2	Step-3	
ক	ল	12 (া)	ম

▼ Compare

ক	ল	01 (NULL MODIFIER)	ম

**The upper word weighs greater than the lower, so the correct sorting sequence will be returned is-

কলম < কলাম

5.4.2 String Sorting

Consider two strings- কীি and কীএ. **Computer Representation of these words-**

String 1		
ক	ী	ি

String 2		
ক	ী	এ

Comparison Steps

Step-1	Step-2	Step-3	
ক	ী	01 (NULL CHARACTER)	ি

Compare

ক	ী	এ	

**The upper word weighs less than the lower, so the correct sorting sequence will be returned as- কীি and কীএ

Algorithm

Bangla characters are inserted in array called MappedInt maintaining the mapping order described in section 5.3.

We used Merge Sort to sort; to compare two words following Comare method is used.

Compare (Word1, Word2)
a. Len= minimal length between Word1 and Word2
b. For i=1 to Len
i. IF MappedInt [Word1 [i]] = MappedInt [Word2 [i]]
Flag= FALSE
IF Word1 [i] is a Character
Flag= TRUE
CONTINUE
ii. IF Word1 [i] is Character *AND* Word2 [i] is Modifier *AND* Flag=TRUE
Return 1
iii. ELSE IF Word1 [i] is Modifier *AND* Word2 [i] is Character *AND* Flag=TRUE
Return -1
iv. ELSE
IF MappedInt [Word1 [i]] < MappedInt [Word2 [i]]
Return 1
ELSE
Return -1
c. IF Length(Word1) < Length(Word2)
Return 1
d. ELSE
Return Length(Word1) > Length(Word2) ? -1 : 0

Complexity

Complexity for the comparison of two Bangla strings = **O (L)**, where L is the minimum number of character in one of the strings. Since the proposed comparison method has linear time complexity like the comparison method of English string algorithm hence the complexity to sort the N bangla words depend on the complexity of sorting algorithm. Also we have mapped each bangla unicode character once so we are ignoring it. We have used merge sort algorithm here. So the complexity for sorting bangla string algorithm is similar to the complexity of the standard sorting algorithm. In our case we used Merge Sort algorithm; hence complexity is **O (NlogN).**

6. DISCUSSION

In this paper we have proposed an efficient and proper way to compare Bangla strings that conforms with the proper structure of Bangla words i.e. each modifier comes with a base letter. Our main effort was to maintain the right order according to the standard set by Bangla Academy while sorting and to preserve the general complexity of standard sorting algorithm. We have also tested the algorithm with more than 56,000 data taken randomly from Samsad Bengali-English Dictionary [17] in our algorithm and the output is completely in proper sequence as represented in Bangla Academy Dictionary. So this algorithm has the potential to be considered as the standard procedure for sorting Bangla strings based on Unicode character

7. CONCLUSION

So far the works carried out related to Bangla sorting procedure have at least one of two major problems: either their time and space complexity is higher or produce wrong results. Our goal was to design an algorithm that can be used to sort Bangla strings accurately without any additional cost of time and space complexity. We devoted ourselves to find out the basic rules for sorting Bangla strings. In order to find out those rules, we had to implement all the procedures described in section 4. Then we came up with theories and ideas described in section 5.2 and put them in rigorous tests to check their validity. We designed the compare method the way a normal compare method works for English. It returns 0, 1 and -1 for equal, greater than and less than respectively. So we propose this method as the standard for comparing Bangla strings.

8. REFERENCES

[1] Bengali language http://en.wikipedia.org/wiki/ Bengali_language Retrieved 2011-05-11

[2] List of languages by number of native speakers http://en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers Retrieved 2011-05-11

[3] Bangla Academy: http://en.wikipedia.org/ wiki/Bangla_Academy

[4] *Bangla Academy Bengali-English Dictionary*, First Edition June, 1994, Bangla Academy, Dhaka, Bangladesh.

[5] Rahman, Md. Shahidur and Iqbal, Md. Zafar, "*Bangla Sorting Algorithm: A Linguistic Approach*". Proceedings of International Conference on Computer and Information Technology, Dhaka, 18-20 December 1998, pp. 204-208.

[6] Mafizul Haque Khan, S M Rafizul Haque, Md. Sharif Uddin, Rahat Khan, A B M Tariqul Islam, "*An Efficient And Correct Bangla Sorting Algorithm*" 7th ICCIT, 2004 Page 125.

[7] Shah Md. Emrul Islam and Muhammad Masroor Ali "*An Approach to Sort Unicode Bengali Text Using Ancillary Maps*", BUET, Dhaka.

[8] Rahman M.A., Sattar, M.A., *A New Approach to Sort Unicode Bengali Text,* 5th International Conference on Electrical and Computer Engineering,

[9] Zibran, M.F., Tanvir, A. et al., *Computer Representation of Bangla Characters and Sorting Bangla Words*, 5th ICCIT-2002_p191-p195, http://homepage.usask.ca/~mfz946/my Papers/Bangla_Processing.pdf

[10] Cormen, Thomas and Leiserson, Charles and Rivest, Ronald: *"Introduction to Algorith*m", Prentice – Hall of India Private Limited, 1999.

[11] Ellis Horowitz and Sartaz Shani,: "Fundamentals of Computer Algorithm", Galgotia Publications Limited.

[12] Unicode Consortium- http://www.unicode.org/charts/ PDF/U0980.pdf

[13] Rajesh Palit, Md. Abdus Sattar, "Representation of Bangla Characters in the Computer Systems", Bangladesh Journal of Computer and Information Technology, Vol. 7, No. 1, December, 1999.

[14] Masum, Md. Salahuddin, "*Study of Bangla Conjunctive Characters for Recognition*", B.Sc.Engg.Thesis, department of Computer Scince and Engineering, BUET, August 2001.

[15] Deitel and Santry "*Advanced Java 2 Platform*", Prentice Hall Publications.

[16] Knuth, Donald "*The Art of Computer Programming*", Addison-Wisely Publications, Boston

[17] Samsad Bengali-English Dictionary - http://dsal.uchicago.edu/dictionaries/biswas-bengali/

[18] Ishida, Richard - Bengali script notes http://rishida.net/scripts/bengali