Computer Representation of Bangla Characters and Sorting of Bangla Words

Minhaz Fahim Zibran, Arif Tanvir, Rajiullah Shammi and Md. Abdus Sattar
Department of Computer Science and Information Technology
Islamic University of Tehnology, BoardBazar, Gazipur 1704.
Emails: zibranm@yahoo.com, chanokko@yahoo.com , shammi07@yahoo.com, masattar@iut-dhaka.edu

Abstract: *This is a discussion about the Bangla character representation in computer system and the sorting algorithm for Bangla words as well. Until recently a number of works have been conducted but no standard has been established. In our paper we have highlighted the limitations of those previous works and proposed a suitable solution. This will be convenient for Bangla word processing, voice synthesis, character recognition and compiler construction, etc.*

Keywords: *Bangla Character Representation, Bangla Text sorting.*

1. INTRODUCTION

Bangla is a very rich language and approximately 10% of world's populations speak in Bangla [1]. Hence, the computerization of this language is the inevitable need today, but unfortunately we have advanced a very little in this regard. Even there is no standard in the Bangla keyboard layout or computer representation of Bangla characters. More significantly, for the development of Bangla database systems a standard for representation of Bangla characters and an expedient, efficient, versatile sorting algorithm is a must.

In our paper we are proposing a method to represent Bangla characters internally in the computer systems, which will provide the scope of efficient sorting of Bangla words. To represent any character set, at least two criteria must be fulfilled. First, it should work along with the typical and standard word processors with ease of writing, saving and retrieval. Secondly, it should be able to undergo further mathematical computation in an efficient manner. All the present keyboard interfaces can easily meet the first criterion as they are smoothly being used in MS-WORD and other word processors. But the word format used in various word processors is not suitable for sorting, matching etc. Because the way the character strings are stored in physical devices is not convenient for any mathematical computation such as sorting. Hence, a logical and suitably defined representation of Bangla character set should be evolved. Since we are just a few years ahead from the inception of computerization in our country, if any standard for Bangla is not introduced, with the passage of time it will be very difficult for us to convert all the works to a standard. In the following pages we will introduce all the pros and cons of Bangla character representation and propose a standard for it.

2. The Bangla Language

In the written form of Bangla there are 11 vowels and 39 consonants. Moreover, there are 10 short forms of vowels called vowel modifiers (ie, Kar), 7 short forms of consonants called consonant modifiers (ie, Fala) [1]. Beside these, there are more than about 253 compound characters composed of 2, 3 or 4 consonants (200 compound characters composed of 2 consonants, 51 compound characters composed of 3 consonants and 2 compound characters composed of 4 consonants) [4]. In accordance with the order of Bangla Academy standard, vowels and corresponding vowel modifiers and their placement within words are listed in Table1.

Table-1**:** Vowels and Vowel Modifiers

Vowels	Vowel Modifiers	Placement (with respect to the character to be modified)	Example
অ	none	none	none
আ	া	right	সাহস
ই	ি	left	বিশ্ব
ঈ	ী	right	নীড়
উ	ু	below	সুজলা
ঊ	ূ	below	অপূর্ব
ঋ	ৃ	below	বৃষ্টি
এ	ে	left	সেতার
ঐ	ৈ	left	বৈশাখ
ও	ো	ে at left, া at right	রোদন
ঔ	ৌ	ে at left, ৗ at right	মৌ

According to the standard of Bangla Academy consonants are ordered as follows:

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

Consonant modifiers (ie, Fala) with their corresponding consonants are listed in Table-2 [9]

Besides the vowel, consonant and their modified form we have a special character Hoshonto (হসন্ত ,্).

Table-2: Consonant Modifiers

Consonants	Consonant Modifiers
ন	◌
ব	ব
ম	◌
য	◌
র	◌, ◌
ল	◌

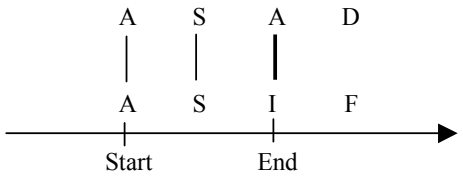
Unlike English Bangla words are not only composed of individual characters placed one after another. In Bangla 2 or 3 consonant can be merged together to form a single compound character. Some examples are given in Table-3:

Table-3: Compound Characters

Number of characters	Compound character	Decomposed form	Example
2	ন্দ	ন + দ	ছন্দ
3	জ্জ্ব	জ + জ + ব	উজ্জ্বল
4	ষ্ট্র্য	ষ + ট + র+ য	স্বাতন্ত্র্য

3. SORTING PROBLEM OF BANGLA TEXT

English words are composed of individual alphabets and so the sorting of English words is quite simple.
To sort two English words we start the comparison from the first letters of both the words and proceed towards the end of the words comparing characters pair by pair. On the basis of the first dissimilar pair of characters, a sorting decision is made.
For example, the sorting of two English words 'ASAD' and 'ASIF' is shown below:



Here the first dissimilar pair is 'A' and 'I'. So decision is to be made from the comparison of these two alphabets. Since 'A' precedes 'I', 'ASAD' is to be placed before 'ASIF' in the sorted list.
In case of Bangla, the scenario is quite different. Bangla words cannot be sorted using such a simple algorithm. In Bangla words vowel and consonant modifiers are placed before, after, above or below any character. Moreover there are frequent uses of compound characters.
For the convenience of typing, all of the currently available Bangla word processors are organized in such a way that, the typist often has to type the modifiers before the consonants.

Moreover, some modifiers such as ে া and ৈ ৗ are fragmented into ে + া and ৈ + ৗ respectively. Keystrokes are stored in the file following the same sequence.
For example, in case of typing সোনালী we first type ে, then স , then া and so on. And in the same order the characters and modifiers are stored in the file. Here two modifiers ে and া are associated with স but actually there is a single modifier ো with স. This results in inconsistency in sorting.
Suppose two Bangla words সকাল and সোনালী are to be sorted. This will be done as described below:
Here স is first compared with ে . Since ে precedes স , সোনালী comes before সকাল in the sorted list. Obviously this sorting is not correct. Because in the word সোনালী, স has the vowel modifier ো া but in case of সকাল, স has no modifier. Hence সকাল should precede সোনালী in the sorted list if we are to follow the standard of Bangla dictionary.

4. POSSIBLE SOLUTIONS

From the proper scrutiny we find that the key problem for sorting is with the vowel modifiers and the compound characters. An efficient way of typing and a proper representation of the characters and modifiers can lead to the solution.

4.1 Method 1: as described in [1]

In order to maintain proper sorting Rahman and Iqbal [1] have proposed an internal representation of Bangla words where a dummy character is placed after the character, which has no modifier. Moreover, it is also ensured that there would be no dummy character between the constituent parts of a compound character. Again, vowel modifiers are included in the character set and they can be typed before or after the characters but for internal representation every time they are to be shifted after the character. 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 @ represents the dummy character:
For sorting the words the relative order in the character set are arranged in the following way-

Null modifier < Vowel modifiers < Vowels < Consonants

This method has the following shortcomings:

- Previously extra vowel modifiers had to be accommodated in the keyboard, which is not needed according to our opinion.
- Shifting of the vowel modifiers adds extra overhead. The keyboard interface has to be complex enough to do this job.

- In the keyboard mapping proposed by them, ঘ is mapped to '[', ঙ is mapped to '\', চ is mapped to ']' and হ is mapped to '{'. But these '[', '\', ']' and '{' symbols are used in Bangla. So they cannot be removed.
- Due to use of the dummy character, a large amount of disk space is consumed to store Bangla words.

Table-4: Internal representation of words by [1]

Word	Internal Representation
অক্ষাংশ	অ ◌ক ষ া ং ◌শ ◌
স্বাগতম	স ব া গ ◌ত ◌ম ◌
কমলা	ক ◌ম ◌ল া
বর্গ	ব ◌র গ ◌
মোড়ক	ম ো ড় ◌ক ◌
কাক	ক া ক ◌

4.2 Method 2: as described in [2]

According to the proposal of Palit and Sattar [2], the keyboard will accommodate vowels, consonants and vowel modifiers. Words are typed as they are spelled. No dummy character is used. The compound characters are divided into their constituent components and stored in the file. The shape of those components will vary based on their relative position in the compound character. All the shapes are stored in the Video ROM and distinct codes are assigned against them. Internal representations of some words are shown in Table-5

Table-5: Internal representation of words by [2]

Word	Internal Representation
কন্ঠ	ক ন্ ঠ্
কষ্ট	ক ষ্ ট্
খদ্দর	খ দ্ দ্ র
উপ্ত	উ প্ ত্
অন্দর	অ ন্ দ্ র
অন্তর	অ ন্ ত্ র

For proper sorting the following order is followed:

Vowels < Consonants associated with the preceding part of the compound characters < Vowel modifiers < Trailing parts of the compound character.

This method has the following drawbacks:

- Both vowels and vowel modifiers are accommodated in the keyboard while the accommodation of only the vowels is sufficient.
- Since different codes are assigned to different shapes of the constituent parts of the compound character, a wide range of shapes and their corresponding codes are to be maintained.

4.3 Method 3: the proposed method

4.3.1 Keyboard Mapping

We propose that only the vowels, consonants and necessary symbols (i.e., @, &, +, = etc) will be accommodated in the keyboard. Moreover a specific key, suppose the key for G, will be used for a link character Δ. Bangla characters are mapped to appropriate ASCII values in such a way that the ordering prescribed by Bangla Academy is strictly maintained. The vowel modifiers are very frequently used. To represent a character modified by a vowel modifier, the modified character may be followed by a link character which is again followed by the vowel corresponding to that vowel modifier. This frequent use of the link character exploits significant amount of memory space.

Since there are only 10 vowel modifiers, for each modifier we can easily assign 10 distinct ASCII values higher than those of the consonants (within the available range from 128 to 255). This omits the use of the link character without hampering sorting and thus saves memory space.

4.3.2 Way of Typing

The words will be typed as they are spelled. Suppose সো is to be typed. At first we have to type স, then we will press the link key G and then ও. When any vowel is typed after the link key, the keyboard interface will store the ASCII value of the vowel modifier corresponding to that typed vowel. The internal representation of সো in the file will be- স ো.

While typing compound characters, for example ন্ত , we have to type first ন , then the link key and then ত . Keyboard interface will store the ASCII values of ন, Link character and ত respectively. Hence, internally ন্ত will be stored in the file as- ন Δ ত.

It is the responsibility of the display interface to show সো and ন্ত appropriately on the display unit.

The internal representations of some words are shown in Table-6.

Table-6: Representation of words by proposed method

Words	Internal Representation
সোনালী	স ো ন া ল ী
সকাল	স ক া ল
সূচি	স ূ চ ি
সূচিতা	স ূ চ ি ত া
অন্দর	অ ন Δ দ র
অন্তর	অ ন Δ ত র

4.3.3 Sorting

When Bangla words are internally represented according to our proposed format, the sorting can be easily done with the algorithm used in sorting English words. The simple algorithm for sorting two Bangla words is given below:

1) Sorting will start with comparison of the first characters of the words and proceed comparing pair by pair towards the end.
2) If the characters mismatch, the word containing the character with higher precedence will go first.
3) If two characters being compared are same, then the following steps are followed:
   i. If these characters are the last characters of the words, then the words are same.
   ii. If the character is the last character of the first word but not of the second, then the first word will be placed before the second.
   iii. Otherwise we have to check whether the characters are followed by the link character Δ. There may be three cases:

   Case-1 (None of the characters are followed by the link character Δ):
   With the next pair of characters the operations starting from step 2 are repeated.

   Case-2 (One of the characters is followed by the link character Δ):
   The word containing the character followed by no link character will go first.

   Case-3 (Both the characters are followed by the link character Δ):
   The operations starting from step 2 are repeated with the pair of characters followed by link character Δ.

For sorting, we will follow the same order as used in Bangla dictionaries:

Vowels < Consonants < Vowel Modifiers

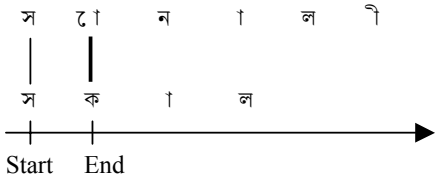
4.3.3.1 Algorithm for Sorting Two Bangla Words

Two Bangla words are internally organized into two strings A1 and A2. This algorithm will sort this two words.

```
Sort_Bangla(A1, A2)
1       i ← 0
2       While TRUE do
3        i ← i + 1;
4               if A1[i] = A2[i]  then
5                       if A1[i] = NULL then
6                           message(' The words  are
                            same')
7                       exit
8                           else  continue
9                           end if
10              else if A1[i] = 'Δ' then
11                      Place A2 before A1
12                      exit
13              else if A2[i] = 'Δ' then
14                      Place A1 before A2
15                      exit
16              else if A1[i] = NULL and A2[i] ≠ NULL
        then
17                      Place A1 before A2
18                      exit
19              else if A2[i] = NULL and A1[i] ≠ NULL
        then
20                      Place A2 before A1
21                      exit
22              end if
```

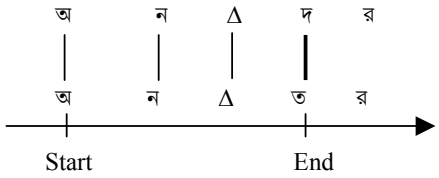
4.3.3.2 Examples of Sorting

Example 1: The sorting of the two Bangla words সোনালী and সকাল



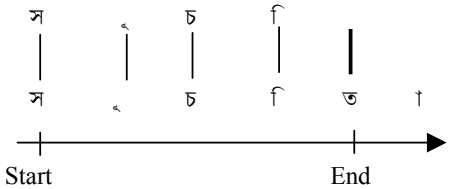
Here the starting character of সোনালী and সকাল is স, but m of সকাল is not followed by link character while in সোনালী m is followed by the link character. Therefore, সকাল will go before সোনালী in the sorted list.

Example 2: The sorting of the two Bangla words অন্দর and অন্তর -



Here the first mismatch occurs in the pair ত and দ. Since ত precedes দ, after sorting অন্তর goes before অন্দর.

Example 3: The sorting of the two Bangla words সূচি and সূচিতা



Here the last compared character pair is same and সূচি ends while সূচিতা has more characters. So, সূচি goes before সূচিতা.

4.3.4 Role of Display Interface

During typing, the words are stored in the file according to the internal representation we have shown. So the file will contain the vowels, consonants and link characters. The shapes of vowels, consonants and modifiers are stored in the Video ROM. While forming compound characters, consonants take different shapes as specified in [2]. The Video ROM will also contain these shapes. The display interface is to properly display the compound characters and the characters along with their modifiers. When the link character ∆ is encountered between two characters, the Display Interface extracts the appropriate shape of the second or first character and displays with proper relative placement.

For example, কো will be stored in the file as ক ∆ ও. The display interface, encountering the link character ∆ between ক and ও, extracts the corresponding vowel modifier (here ে া) and displays কো.

Again, incase of র্গ, this is stored in file as র ∆ গ. Finding link character between র and গ, the display interface will display র্গ, in the display unit.

Similarly, when the display interface will encounter link character between ক and স, it will display ক্স.

5. CONCLUDING THOUGHT

In this paper we have proposed the internal representation of Bangla words which will be convenient for sorting, matching and other mathematical computation. We have included only the vowels and consonants in the keyboard, not their modified form. This will provide ease in Bangla word processing.

From the proper study of Bangla words if it is found that the modified form of certain characters are more frequently used than the characters themselves, then the keyboard may accommodate those modified forms instead of the original characters. It will obviously minimize the key-strokes. If the character itself is accommodated then the user can get the modifier by pressing that key along with the link key and vice versa.

References

[1]. 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.

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

[3]. Ali, Md. Ameer, "Development of Bangla Keyboard". B.Sc.Engg.Thesis, Department of Computer Science and Engineering, BUET, August 2001.

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

[5]. Khan Ferdous, "Haraf Shamashha", Munir Chowdhuri Rachanabali, Vol.3, pp.551-553, 1984.

[6]. Khan, Mozammel Haq Azad, "Optimal Realization of Bengali Keyboard and Character Encoding for Computer Application", M.Sc Engg Thesis, Department of Computer Science and Engineering, BUET, 1986.

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

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

[9]. Mohammad, Kazi Din: " *Adhunik Bangla Byakoron O Rochona",* First Edition, June,1999.

[10].Ezzel, Ben and Blaney, Jim, "NT 4/Windows 95 Developer's Hand Book", BPB book centre, First Edition, 1997, pp : 16-183, 1230-1295.

[11].Pappas, Chris H. and Murry, Willium H., "Visual C++ 5: The Complete Reference", Tata McGraw Hill Publishing Company Limited, New Delhi, 1998, pp:682-732.