*МАТЕМАТИКА И МАТЕМАТИЧЕСКО ОБРАЗОВАНИЕ, 1999*
*MATHEMATICS AND EDUCATION IN MATHEMATICS, 1999*
*Proceedings of Twenty Eighth Spring Conference of*
*the Union of Bulgarian Mathematicians*
*Montana, April 5–8, 1999*

# DEVELOPMENT OF LINGUISTIC SOFTWARE FOR WORD 7.01*

**Hristo Dimitrov Krushkov**

The word processor WORD for WINDOWS is one of the most employed systems in this field. Lately proofing tools for Bulgarian language (BL) were implemented in the WORD environment to become user friendly for Bulgarian users (check spelling, hyphenation etc.). In this paper the possibilities for development of additional linguistic software to help the linguists using WORD are presented. The realisation of special finding of wordforms with definite grammatical features is described.

**1. Introduction.** The word processing is one of widely used computational tools providing text creating and formatting. The standard linguistic functions of word processors are limited in searching (eventually case sensitive) of words or parts of words, automatic check spelling as well as hyphenation. Thesauri are also used making easy the text editing. Any word processors like WinWord perform checking of some basic grammar and style rules [4].

There are two main problems in front of the users of word processors using them for linguistic investigations of Bulgarian texts. The former is these word processing systems are developed for English speaking or western users and the above mentioned tools are accessible only for them. The latter is the lack of tools for searching of all Bulgarian wordforms derived from a base form as well as searching of wordforms with definite grammatical features. There is no contemporary word processor providing such operations.

This problem is solved for Bulgarian language in general [1]. An Integrated Linguistic Environment has been developed for MS DOS, WINDOWS 3.11 as well as for WINDOWS 95/NT. It contains text editor which owns described above tools. The main disadvantage of this environment is that it manipulates only files in text format. There are no possibilities to maintain files in WinWord format.

In this paper a realisation of linguistic tools (searching of wordforms with definite grammatical features) as a part of WinWord 7.0 for linguistic investigations of Bulgarian texts is described. The programming languages Word Basic (WB) for WORD and Delphi 2.0, both for WINDOWS 95/NT are used.

*This paper is partly supported by the project NSF-I-608-96 "Modelling of structural text characteristics by knowledge-based schemata"

**2. A formal model of bulgarian morphology. Morphological processor.**

Bulgarian language belongs to the group of inflective languages. Bulgarian inflection is described as a number of grammatical rules. A classification of Bulgarian inflection in view of the mentioned rules and the grammatical features of the words is presented in [2]. There are 187 different inflectional types in that classification divided into parts of speech (POS) as follows: 75 for the nouns, 14 for the adjectives, 41 for the pronouns, 11 for the numerals and 42 for the verbs. Every Bulgarian inflecting word can be classified as a member of some of these types.

From a mathematical point of view the Bulgarian words are divided into disjoint classes of equivalence. Every class has a unique machine number for identification and a list of rules for generation of the paradigm. A part of speech is a set of classes. Every set can be divided into subsets depending on criteria pertaining to this particular part of speech.

For example the set of nouns includes the classes with machine numbers 1-75. There are 4 subsets depending on the gender as follows: with machine numbers 1-40: masc., 41-53: fem., 54-73: neut., 74-75: only plural.

Figure 1 shows the inflectional type numbers for the parts of speech. The last 6 rows (noninflective POS and proper nouns) are added from the author in order to make the classification more complete.

| Inflectional types No | POS/Subsets | Descriptions | |
|---|---|---|---|
| 1..75 | Common nouns | N | $T_1$ |
| 1..40 | masculine | NM | $T_{11}$ |
| 41..53 | feminine | NF | $T_{12}$ |
| 54..73 | neuter | NN | $T_{13}$ |
| 74,75 | only plural | NP | $T_{14}$ |
| 76..89 | Adjectives | A | $T_2$ |
| ..... | ..... | ..... | ..... |
| 142..187 | Verbs | V | $T_5$ |
| 188 | Adverbs | Adv | $T_6$ |
| 189 | Particles | Part | $T_7$ |
| 190 | Prepositions | Prep | $T_8$ |
| 191 | Conjunctions | Conj | $T_9$ |
| 192 | Interjections | Intr | $T_{10}$ |
| 201..230 | Proper nouns | Prop | $T_{11}$ |
| 201-215, 221,222 | Male name based | | |
| 216..220 | Female name based | | |

Figure 1. Inflectional type numbers for the parts of speech.

Two words are in the same class if their paradigms are generated in the same way. The paradigm is described as a list of wordforms with specific grammatical features for each of them. Every wordform also has a number. Two wordforms with equal numbers have the same grammatical features.

For example in the paradigm of the adjectives, wordform num. 1 has grammatical features (masc., sing.); wordform num. 2 has grammatical features (pl.); etc. For all parts of speech wordform num. 1 is the base (citation) form.

A morphological processor for BL was built on the basis of common properties to which submit inflective morphologies [1]. It is a tool performing automatic morphological analysis and synthesis.

The purpose of the automatic morphological analysis is to perform automatically a morphological classification of an arbitrary wordform. This includes identifying the base form of the word, its grammatical features and to which inflectional type (part of speech) it belongs. In case of homonyms (when the wordform belongs to more than one inflectional types and has different grammatical features) all possible types must be found. Every wordform obtains 2 formal features:

1. An inflectional type number
2. A wordform number in the paradigm of that type.

The inflectional type number determines the part of speech the analysed word belongs to.

**3. A program realisation.** A program (WB macro) is realised which gives opportunities to the user to search words with definite grammatical features in a WinWord 7.0 document. This macro uses a dynamic link library LIBMORF.dll. The 32 bits dynamic library (working under Windows 95/NT) contains a morphological processor for BL and manipulate a morphological lexicon. Both are developed by the author and a team in the Department of Computer Science, Faculty of Mathematics and Informatics at the Paisii Hilendarski University of Plovdiv.

**3.1.External subroutines.** The function (from LIBMORF.dll) used in the macro is declared (Delphi 2.0) as follows:

```
function AnTnWn(s:pcharInput text):pchar;export;stdcall;
```

Input parameters

`s:pchar` – this is the consecutive word from the text which grammatical features of have to be determined.

The type of the function is `pchar` – a string of ordered pairs (inflectional type number, wordform number), separated by comma (','').

The definition of this function in the macro is `AnTnWn$(s As String)`

**Example:** The result of the function call with an input parameter the word *(system)*: `AnTnWn$(“”)` is “41,1,”. Where 41 is an inflectional type number (see Figure 1). The word belongs to the set of nouns because 41 is between 1 and 75. Looking at the subsets at the same Figure we realised this number belongs to the subset [41..53]: nouns-feminine. Finally the wordform number gives all grammatical features.

The result of a function call with an argument the word *(the form, the format as well as formats-count form):*

`AnTnWn$(“”)` is “41,2,7,2,7,6,”. These are three ordered pairs: <41,2>, <7,2> <7,6>.

**3.2. Main data structures.** A binary array **SearchAr** containing 0 and 1 values, in which the searching grammatical features is “ticked” by value 1. The values are obtained from a dialogue boxes constructed for every part of speech.

Every column belongs to a part of speech. The first one belongs to the nouns, the second – to the adjectives etc., the last one (11-th) belongs to the proper nouns (see Figure 1). The element with index 0 (from the row 0) is responsible for choosing the corresponding to the column part of speech. If it is equal to 0 – this part of speech has not been "ticked" for search. If it is equal to 1 – it has been "ticked" for search. In case of wordforms (inflective parts of speech – first 5 columns as well as the last one corresponding to nouns, adjectives, pronouns, numerals, verbs and proper nouns), besides the 0 – element is obligatory to "tick" another element in the corresponding columns, with the respective wordform number.

The column number 0 is auxiliary. It is used for nouns to "tick" the gender: masculine – first row, feminine – second row, neuter – third row as well as only plural – 4-th row.

For example the state of the array when nouns masculine, short and full definite article are "ticked" is as follows:

| Indexes | 0. | 1. | 2. | 3. | 4. | 5. | 6. | 7. | 8. | 9. | 10. | 11. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0. | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1. | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2. | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3. | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 55. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

An array `TnWn`, in which every column has the following purpose:

0 column – inflectional type number;

1-st column – POS number (1-11);

2-nd column – wordform number;

3-rd column – auxiliary, where a subset number for nouns (1-4) is being written.

In case of homographs (when the two or more wordforms have the same graphic representation) a number of rows (equal to the number of homographs) is being filled up. The element TnWn(0, 0) contains this number.

**3.3. Main subroutines.** The filling up of the array TnWn is performed in the procedure `ConvertToTabnWordn(AnalysResult$)`. The input parameter `AnalysResult$` is a string of ordered pairs <inflectional type number, wordform number>, separated by comma (','). This string is a result from the function `AnTnWn$`.

The function `SpecialMatch(w$)` looks up matching of the result of the analysis `w$` (after converting) in the array `SearchAr`.

Functions displaying the corresponding dialogue boxes has been realised. The main dialogue box "ticks" the requiring parts of speech. Push buttons are included next to every inflective part of speech for choosing definite grammatical features attached to the selected part of speech. After pushing it a dialogue box with these features is being displayed. The user could select desirable features. Afterwards the program finds the first word owned these features, selects it and displays a message box asking whether to continue searching. The searching finishes either in case of end of the document or by user cancellation.

**4. Outlook.** New functions would be implemented for WORD 7.0 ( as well as for WORD 97 which uses VISUAL BASIC) for linguistic processing like searching of all Bulgarian wordforms derived from a base form, corpora tagging, intelligent check spelling with an auxiliary dictionary with base forms only, statistical processing etc.

REFERENCES

[1] Hr. Krushkov. Modelling and building of machine dictionaries and morphological processors. Plovdiv, Ph.D. Thesis, 1997.
[2] B. Krustev. The Bulgarian Morphology in 187 type tables, Sofia, Nauka i Izkustvo, 1984.
[3] B. Stefanov. Bulgarian proofing tools in MS OFFICE 95 / 97, magazine Computer, issue 8, 1997.
[4] M. Wait, G. Arka. Familiarise with word processing, Sofia, Technica, 1988.

Hristo Dimitrov Krushkov
Department of Computer Science
Faculty of Mathematics and Informatics
The Paisii Hilendarski University of Plovdiv
24, Czar Assen St, 4000 Plovdiv, Bulgaria
e-mail: hdk@uni-plovdiv.bg
http://www.uni-plovdiv.bg/hdk/hdk.htm

## РАЗРАБОТВАНЕ НА ЛИНГВИСТИЧЕН СОФТУЕР ЗА WORD 7.0

**Христо Димитров Крушков**

Текстообработващата система WORD за WINDOWS е една от най-използуваните у нас. Напоследък бяха реализирани средства, интегрирани към нея [3], улесняващи работата на българския потребител (проверка на правописа, сричкопренасяне и др.). В настоящата статия са представени възможностите за разработване на допълнителен лингвистичен софтуер, който да подпомага потребителите лингвисти при работа с тази текстообработваща система. Описана е и реализацията на търсене на словоформи в документи на WORD, притежаващи конкретни граматични характеристики.