World Applied Sciences Journal 36 (2): 197-212, 2018
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2018.197.212

# Development of the Intelligent Electronic Educational Editions on the Subject "Algorithms and Data Structures" Based on the Technology OSTIS

*Aliya Kintonova, Bakkaisha Andassova, Madina Yermaganbetova, Elmira Maikibayeva, Zhanat Abdugulova, Kuralay Sadvakasova and Gabit Shuitenov*

Department of Informatics & Information Security,
Gumilyov Eurasian National University, Astana, Kazakhstan

**Abstract:** The article is characteristic of OSTIS technology and system design of electronic textbooks. The main part of the article describes the direct implementation of intelligent electronic educational publications on the subject "Algorithms and Data Structures" on the basis of OSTIS technology. The chapter describes the possibilities of OSTIS core technology - meta-IMS. OSTIS, SC-language step by step described implementation of intelligent electronic educational publication "Algorithms and Data Structures" and describes the structure of intelligent e-learning issues. Development of an electronic textbook, which promotes educational resource for e-learning universities and will be used in the educational process in the study of undergraduate students for a compulsory subject "Algorithms and Data Structures" and the preparation of students for the TDMA in the discipline.

**Key words:** Algorithms · Data · Intelligent electronic textbook · Sc-language · OSTIS technology · E-learning · Educational process · Electronic educational publications

## INTRODUCTION

The increasing popularity leads to a process of informatization of modern society. One of the most important and biggest trends is the use of computer technology in education.

Artificial intelligence is an interdisciplinary scientific discipline. This is due to its great potential, as strong results are born at the junction of scientific fields. But due to the same great difficulties since the development of artificial intelligence requires a deep understanding and cooperation of researchers with different style of thinking, a different approach to the subject of study, different mentality, different target installation and traditions.

The current stage of development of artificial intelligence is in urgent need of overcoming these difficulties. The most important task of artificial intelligence at the moment is the construction of the integrated intelligent systems theory, in which would combine a variety of areas of artificial intelligence [1].

Informatization of society - it is a social process being implemented globally. Information in a society provides high level of informational services leading to the availability of any human sources with a verified information and visualization of the received information with the significance of a data used.

In this paper, we study the most important part of the process of informatization of society and education - the creation of one of the essential forms of education using powerful arsenal of modern information technologies - intelligent electronic textbook on discipline "Algorithms and Data Structures" on the basis OSTIS technologies.

*The scientific novelty of the work,* reported in this paper is the development of intelligent electronic academic publications on the subject "Algorithms and data structures" on the basis OSTIS technologies - metasystem IMS.OSTIS, SC-language.

**Theoretical Significance:** We studied electronic educational publications and development technologies.

**Corresponding Author:** Prof. Aliya Kintonova, Department of Informatics & Information Security, Gumilyov Eurasian National University, Postal Code: 010008, Pushkin Str., 11, an educational building No 2, Astana, Kazakhstan.

Analyzed the design tools of knowledge representation based on OSTIS technologies. Possibilities metasystem IMS.OSTIS, SC-language.

**Practical Significance:** The developed intelligent electronic textbook on discipline "Algorithms and Data Structures" will be used in the educational process in the study of a compulsory subject "Algorithms and Data Structures" and the preparation of students for the TDMA in the discipline.

## MATERIALS AND METHODS

*The theoretical basis for the study* were OSTIS conference materials (2011-2016 years), "Design Tools" monograph "The representation and processing of knowledge in graphical dyanmic associative machines" [2], handbook on the subject "Design programs in intelligent systems" (BSUIR, Minsk ), "installation kernel OSTIS» (BSUIR, Minsk).

**Classification of Electronic Textbooks:** Information and computer technology as a result of the widespread availability of computer equipment, which deeply penetrates into all aspects of human life. Education is the main area for activity of the society, to keep pace with the times and rapidly implements informative systems for training in educational institutions of different levels.

**A New Generation of Intelligent Tutoring Systems:** Here are the basic qualities that combine to have a new generation of ITS (in this case for the sake of brevity, we shall call them simply IOS).

- IOS learns not only through the collection of information by the system itself, but also through dialogue with the student.
- The system is able to independently solve the problem, the solution of which it is to teach the student.

**Open Semantic Technologies of Intelligent Systems (OSTIS):** *OSTIS technology* is a set of tools and techniques for the design of intelligent systems, as well as to improve the technology. The very ideology of intelligent systems, maximizing the independence of the evolution of knowledge based on the evolution of problem solvers, allows us to consider intelligent systems as a class of computer systems with a high degree of openness [3].

The technology uses a technique of OSTIS component design, which is a factor in the maturity of any technology and is based on the ever-expanding library of reusable components.

OSTIS technology idea is a specific embodiment of knowledge represented in intelligent systems that do not depend on any virtual knowledge processing machine or on different platforms, the technical realization of the intellectual system. Presentation of data would depend only on the semantics provided by the subject area. The basis of this method of knowledge representation will be standardized in an abstract semantic network with the base set-theoretic interpretation [2].

## RESULTS AND DISCUSSION

Effective use of innovative technologies in educational process prepares students to various professions, to become competitive in the future specialists. Teachers, through the effective use of innovative technologies, can teach students to formulate and communicate their ideas, develop creative thinking to solve problems in business relations, effective use of all kinds of information [4].

Today, in educational institutions, various kinds of theoretical and practical innovation. This question engaged scholars such as A.Ya. Savelyev, V.M. Maksimov, G.K. Selevko. According to research of V.M. Maximov, there are following concepts offered: the structural logic technology, integration technology (interrelation of various disciplines), gaming technology, educational technology, information technology, interactive technology.

These concepts are widely used at present in pedagogy. Education in the educational process is a joint teaching process the teacher and the student, aimed at achieving a specific goal. And in order to achieve this particular goal due to the intensification of the educational process.

In the educational process necessary to make extensive use of educational, educational, developmental learning function. Each pedagogical technology creates favorable conditions, opportunities for personal self-development, enhancing its creative potential [5].

Electronic textbooks, regardless of type, can significantly help the teacher in the learning process and improve the quality of students' knowledge. Nevertheless, the development of scientific and technical progress requires a rational use of new teaching methods, technical manuals, in other words, new learning technologies.

Electronic textbook is program-methodical complex, located on a magnetic portable device, providing the possibility of mastering the training course by learners with the help of a computer, on their own or with the assistance of a teacher [6].

Electronic textbook should consist of three main components:

- The main information section - presentation;
- The practical section, responsible for consolidation of acquired knowledge -uprazhneniya;
- Section for the assessment of students' knowledge - testing.

Continuous modernization of information technology can provide optimization and intensification of the educational process. Therefore, the new technology is a major factor in the development and achievement of good results for each school [7].

**The Implementation of Intelligent Electronic Educational Publications on the Subject "Algorithms and Data Structures" on the Basis of OSTIS Technology:** Key Features of OSTIS technology. The basic idea is a graph of storage realized in the Internet space. The general principle of architecture: the relevant classes of network pages (Sc-graphs), shown through web- interface. Language semantic network implemented using Sc-code. A method of user information extraction: intelligent portal that can educate themselves (IMS.OSTIS). The means of data processing: the semantic web. Tool-artist: the inference engine. OSTIS pays semantically compatible abstract models of intelligent systems on all platforms. Skins for the creation of a knowledge base: Shell to create a knowledge base - a software product that has a means of implementing knowledge for specific subject areas. User task is not in the direct programming and in the accumulation of knowledge and input from the shell of functions.

The technology on which the machine is formed by knowledge processing metasystem IMS.OSTIS is part of the open semantic technology designed by OSTIS intelligent systems [3].

The user interface metasystem IMS. OSTIS SCG-based on the code. All objects are displayed in the main window and feature some SC-elements in the knowledge base.

SC-code is a universal core of open language semantic networks, which is the result of the integration of various languages semantic networks constructed on the basis of the SC-code and set.

- Fixed alphabet (SC-code alphabet);
- Constantly expanded (open) family of key components, which is composed of all key components of all integrable languages.

SC-code represents the unity of language and meta-language. For example, in SC-structures you can describe the syntax, semantics and ontology of SC-code. From a formal point of view, SC-code can be interpreted as a meta-language of semantic base specification of SC-elements using the special set of key nodes SC-code.

Semantic knowledge base every intelligent system structure is interpreted as a reflection of the hierarchical system of interconnected with each other subject areas represented in the knowledge base. This implies clarifying the concept of the subject area; the development of linguistic means to describe the structure of the subject areas using standardized semantic networks; the development of linguistic resources to describe the typology of subject areas and different types of connections therebetween.

Structuring the knowledge base, the allocation to it of various interconnected substructures needed for a variety of reasons. In particular, it is necessary for didactic purposes (man to assimilate some knowledge, it is desirable to have some sort of table of contents or "map" of this knowledge that allows you to plan and consider their assimilation them with varying degrees of detail), as well as for the organization of the distribution of work on the design of knowledge bases (when different artists entrusted with the development of various fragments of knowledge, with enough clear boundaries) [1, 9, 10].

**Logical-semantic Model of User Interface:** Like any other system built with the use of OSTIS technology, user interface and a metasystem is built using the component method. We distinguish the following classes of user interface components:

*Translation components:* The components of this class provide a translation of SC-code on an external language and vice versa;

*Imaging components:* This class component provides an output data, which is presented in a foreign language;

*Component editing:* This class component provides input user information to the external language.

Each component consists of a knowledge base and a set of SC-fragment agents. Because broadcast metasystem components in PI have the following components: - a translator of SC-SCsJson code format (the format is similar in structure to the SCs-code level 1, adapted for Web);

Translator of SCsJson format in SC-code;
Scs-compiler code SC-code;
Translator SCg-code SC-code;
The translator of the SC-code-code in the SCG.

Abstract knowledge processing machine consists of the following agents SC-(SC-additives, which are part of the broadcast and display components):

Sc-generating agent command at an instance is a general description.

All user interface commands are described in the knowledge base and contain any layout (template), on the basis of which the inner -agent creates a copy of this command with replaceable BEE arguments;

Sc-transfer agent runs responses to user requests. This SC-agent waits for a response to a user request and creates a copy of the team, which will start broadcasting a response to the external language.

Sc-agent garbage collection. This SC-agent performs a search and deletes information that has been copied or used as a result of the changes. This information may already be completed by a broadcast team and so Dale, etc .;

Sc collection agent IDs. This SC-agent collects all detected SC-elements in order to provide a quick search for items on the ID [11, 12].

- It is implemented on the basis of OSTIS technology;
- It provides comprehensive support for the development of intelligent systems in accordance with OSTIS technology;
- It contains all the necessary skills for this (knowledge of the available models, methods and design tools);
- It is a form of realization of material technology OSTIS;
- It manages the automation project aimed at continuous improvement of them.

**Stage 1: Implementation IEUI "Algorithms and Data Structures" with SC Metasystem IMS.OSTIS Language:** SC-code is the abstract language that can therefore be realized on graphics verifying SC-code (for example, SCg-code), numerous options image sc-constructs in the form of character strings (e.g., SCs-code), numerous variants machine representation sc-constructs in standard computer memory address and in particular, structurally reconfigurable associative memory for future computers designed to process known databases. SC code constitutes not only coded sophisticated computer networks, which do not appear "alien" in semantic network representation and their description is also in the form of semantic networks, but allows more simple forms - with the minimum alphabet.

Scg-language - one of the known methods of visual presentation sc-texts or sc.g-graphs. Most often SCg-code is used in the fourth level. To make domain-specific language you need to consider all levels of the structure in detail, in order to specify entities and relationships that is in the SCG-described code.

SC-language may be used in meta-language for the registration of any external language, ie, languages, which shall not have sc-designs.

Analyzing metasystem «ims.ostis.net», we can identify the following components of the SCG-code:

Sc-design is the semantic web, described in SC-code, the foundation of the knowledge base.

Sc-member - a label or key node in sc-design, can be a meaningful representation of the object, structure and control.

Sc-constant - the key concept and character sets.
Sc-variable - semantically simple piece of text.
Sc-alphabet - is a set of types of sc-elements.

We describe the structure of the Sc-alphabet: Sc - Alphabet described label depicting one or another concept, which is called sc-site assembly. Sc - the essence of which is a set of semantic components alfavita.Sc site is an abstract class that inherits from the following entities:

Structure - class elements are presented below in accordance with Figure 2.

Fig. 2. Structure of the concept – it shows the range of the node (the set), an attribute "ID" means that the node is represented by an abstract class in accordance with Figure 3.

In this model, primitive formultions, identifiers and regular group of objects are seen. Conference Information will be represented in the knowledge base as a system identifier that contains the mapped information about it. All system IDs are required in order for the system to merge them identical nodes. System IDs are unique. In other words, there cannot be two identical node with the same identifier. If such sites are present in SCg-text, it means that they bound to the same object.

Fig. 2: Structure

Fig. 3: Concept

Fig. 4: Material object

Fig. 5: Non-binary bundle

Fig. 6: Many

Fig. 7: Neurological interactions

These nodes will be presented as data sets:

- Topics (theme);
- Authors (Authors).

material object. It is depicted in accordance with Figure 4.

Non-binary bundle - label, with which creates a link between the essence and the object, which is the identifier in this hereditary group of objects in accordance with Figure 5.

Set - a set, against which all other subsets of normalized set is in accordance with Figure 6.

Constant node for processing connections and types of bonds in accordance with Figure 7.

SC-links - a sc-element identifying the specific file that has the ability to view or some computer system with which it is possible to interact. Here in accordance with Figure 8 demonstrated the lecture structure [1]1, made using Sc Code.

Scs-code - the set of sc.s-texts. Each sc.s-text is a sequence of sc.s-offerings, each with a known delimiter; (Double the semicolon). Many sc.s- are divided on many simple and many complex sc.s-proposals. Each complex sc.s- proposal is as part of the integrated proposals that ends as a limiter (* ... *).

As part sc.s-text of any level we can use comments given in such a type:

```
// A single line comment
 / * Multi line comment * /
```

At the beginning of the file describing sc.s-text it is strongly recommended to describe the level and version of the SCs-Code. For this, the comments are given the following form:

```
/ * SCs_code < "version"> .Level < "level number">
* /
```

For example:

```
 / * SCs_code0.1.0.Level 6 * /
```

Universal design of building units:

```
System sc-ID
=> Nrel_main_idtf:
[Main sc-id]
(* <- Lang_ru ;; *);
<- Rrel_key_sc_element:
(*
<- Sc_explanation ;;
<= Nrel_sc_text_translation:
...
(*
-> Rrel_example:
"File"
(* <- Lang_ru ;; *) ;;
*) ;;
*) ;;
```

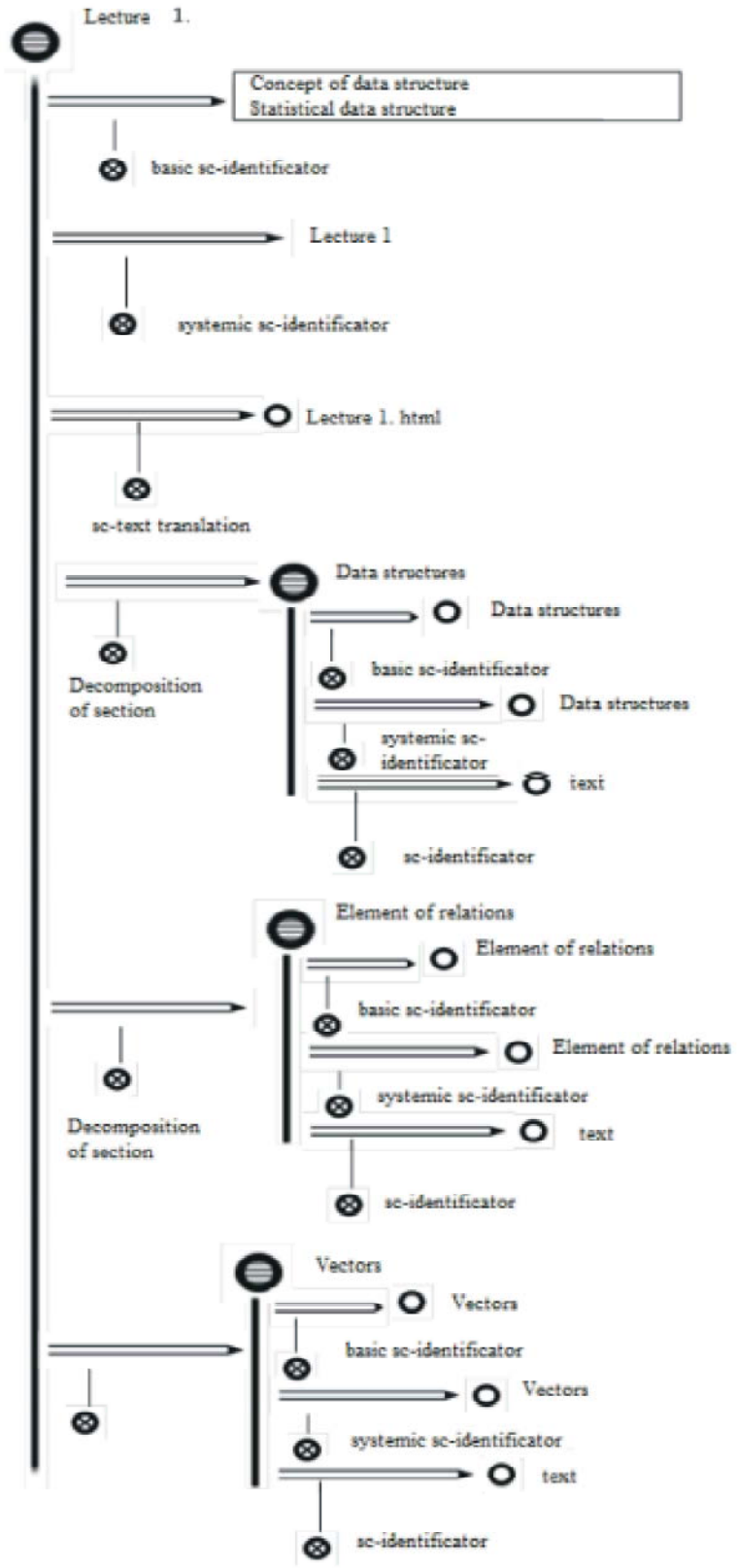


Fig. 8: Structure of the lecture made by a SC code

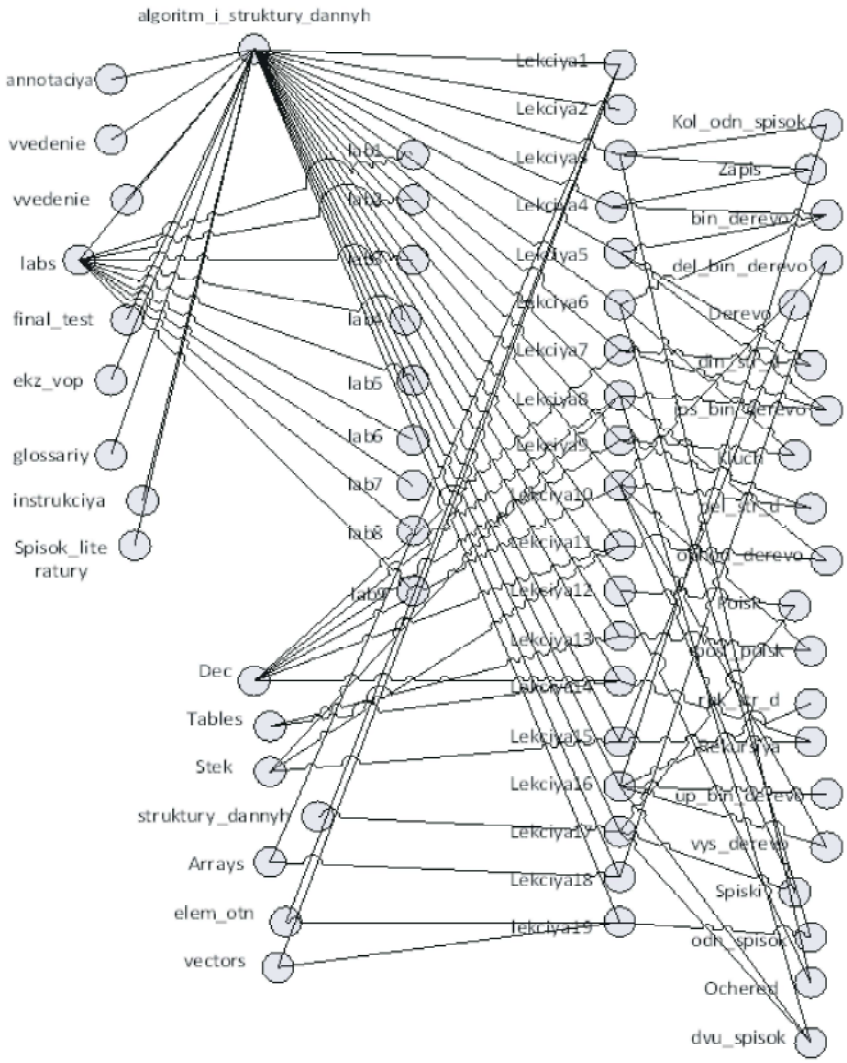


Fig. 9: Scheme of links between the nodes of the training manual

Universal design of building is secondary:
System sc-ID
=> Nrel_main_idtf:
[Main sc-id]
(* <- Lang_ru ;; *);
=> Nrel_idtf:
[Text]
(* <- Lang_ru ;; *) ;;

The data for the creation of the final textbook sites listed in the B-D applications.

**Stage 2: Implementation. Creating Links Between the Nodes:** After creating units to merge them into a single semantic network, that is, to set up links between the elements textbook content (lectures, terms, etc.).

The scheme links between nodes textbook in accordance with Figure 10.

To create connections between core nodes in OSTIS, decomposition section is used. Decomposition is carried by the structure of the form:

<= Nrel_section_decomposition:

{
Identifier1
(*
=>nrel_section_base_order:

Identifier2;;
*);
Identifier2};;

Where nrel_section_decomposition - creates a decomposition section of these nodes, nrel_section_base_order - indicates the order of arrangement of nodes.

Host main section with customized connections (decomposition) as follows:

```
doc_technology_ostis
=> nrel_main_idtf:
[Algorithms and Data Structures]
(* <- Lang_ru ;; *);
<= Nrel_section_decomposition:
{
annotation
(*
=> Nrel_section_base_order:
vision;;
*);
vision
(*
=> Nrel_section_base_order:
lecture1 ;;
*);
Lecture 1
(*
 =>nrel_section_base_order:
 Lecture 2;;
 *);
 Lecture 2
 (*
 =>nrel_section_base_order:
 Lecture 3;;
 *);
 Lecture 3
 (*
 =>nrel_section_base_order:
 Lecture 4;;
 *);
 Lecture 4
 (*
=>nrel_section_base_order:
 Lecture 5;;
 *);
 Lecture 5
 (*
 =>nrel_section_base_order:
 Lecture 6;;
 *);
 Lecture 6
 (*
 =>nrel_section_base_order:
Lecture 7;;
 *);
 Lecture 7
 (*
 =>nrel_section_base_order:
 Lecture 8;;
 *);
 Lecture 8
 (*
 =>nrel_section_base_order:
 Lecture 9;;
 *);
 Lecture 9
 (*
 =>nrel_section_base_order:
 Lecture 10;;
 *);
 Lecture 10
(*=>nrel_section_base_order:
 Lecture 11;;*);
  Lecture 11
  (*
  =>nrel_section_base_order:
  Lecture 12;;
*);
 Lecture 12
 (*
 =>nrel_section_base_order:
 Lecture 13;;
 *);
 Lecture 13
 (*
 =>nrel_section_base_order:
 Lecture 14;;
 *);
 Lecture 14
 (*
 =>nrel_section_base_order:
 Lecture 15;;
 *);
 Lecture 15
 (*
 =>nrel_section_base_order:
 Lecture 16;;
 *);
 Lecture 16
 (*
 =>nrel_section_base_order:
 Lecture 17;;
 *);
 Lecture 17
```

```
(*
 =>nrel_section_base_order:
 Lecture 18;;
 *);
 Lecture 18
 (*
 =>nrel_section_base_order:
 Lecture 19;;
 *);
 Lecture 19
 (*
 =>nrel_section_base_order:
 labs;;
 *);
 labs
 (*
 =>nrel_section_base_order:
 final_test;;
 *);
 final_test
 (*
 =>nrel_section_base_order:
 ekz_vop;;
 *);
 ekz_vop
 (*
 =>nrel_section_base_order:
 glossary;;
 *);
glossary
 (*
 =>nrel_section_base_order:
 instruction;;
 *);
 instruction
 (*
 =>nrel_section_base_order:
 references;;
 *);
 references
 };;
```

**Stage 3: Implementation Phase. Creating EOS Content in HTML Format:** To optimize the display of content and eliminating the need of using third-party software in the tutorial files using HTML format. Use any text editor to create data files and a standard html text markup language. To do this, you want to create the frame type of the page:

```
<DOCTYPE HTML PUBLIC! "- // W3C // DTD HTML 4.01 Transitional // EN"
"http://www.w3.org/TR/html4/loose.dtd">
<Html>
 <Head>
 <Meta http-equiv = "Content-Type" content = "text / html; charset = utf-8">
 </ Head>
 <Body>
 </ Body>
 </ Html>
```

In this frame between the body tags you want to place the content corresponding to the site and place it in a suitable display for the user by using the tags <p>, <h1>, <img>, etc. On completion of the html file corresponding nodes, OSTIS core system at startup check each site for errors and a positive result at this stage begin to correctly display the textbook in accordance with Figure 10.

**Implementation Stage 4. Creating Cross-references to the Secondary Nodes:** To create a cross-reference to the components of the system in html pages of a textbook, we can use a specific system type tag:

```
< Sc_elementsys_idtf = "Sistemnyy_identifikator_uzla"> text </ sc_element>
```

This tag is automatically read by the system and booting nodes OSTIS textbook provides browser in html code have a link in the form of

```
<a href="#" class="sc-element" sc_addr="number_equivalent_uzla"> text </a>
```

The numeric equivalent of a node is automatically assigned to each node sistemy.This design can be used anywhere in the html text node. For example:

```
<P> The vector which consists of a completed number is strictly defined. </ P>
<Palign = "center"> <b> <i> <sc_elements_idtf = "arrays"> Array </ sc_element> </ b> </ i> </ p> <p>
```

In general, the member <b> <i> <sc_elements_idtf = "arrays"> Array </ sc_element> </ b> </ i> is an element of <b> <i> <sc_elements_idtf = "vectors"> vector </ sc_element> </ b> </ i>, which itself is an elementary structure.

In this passage highlighted html code word array and vector. Now the user can enter data on the terms for a direct link, in accordance with the Figure 11.

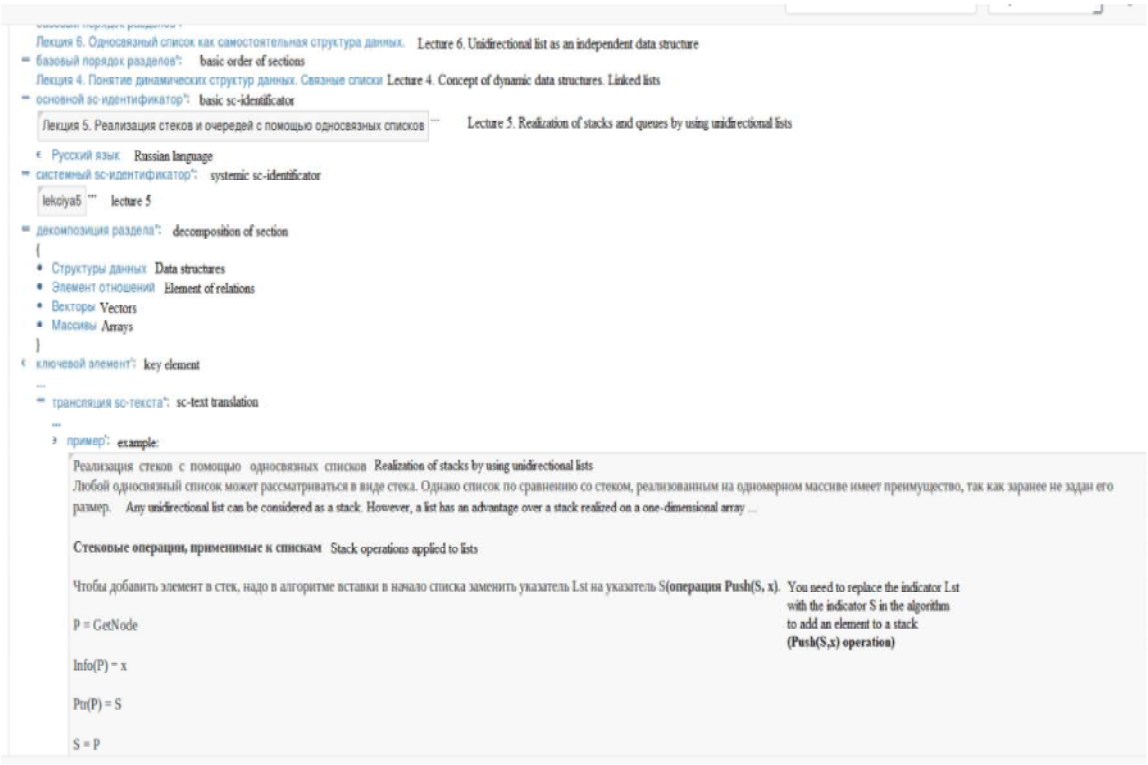


Fig. 10: IEUI interface on the subject "Algorithms and Data Structures"

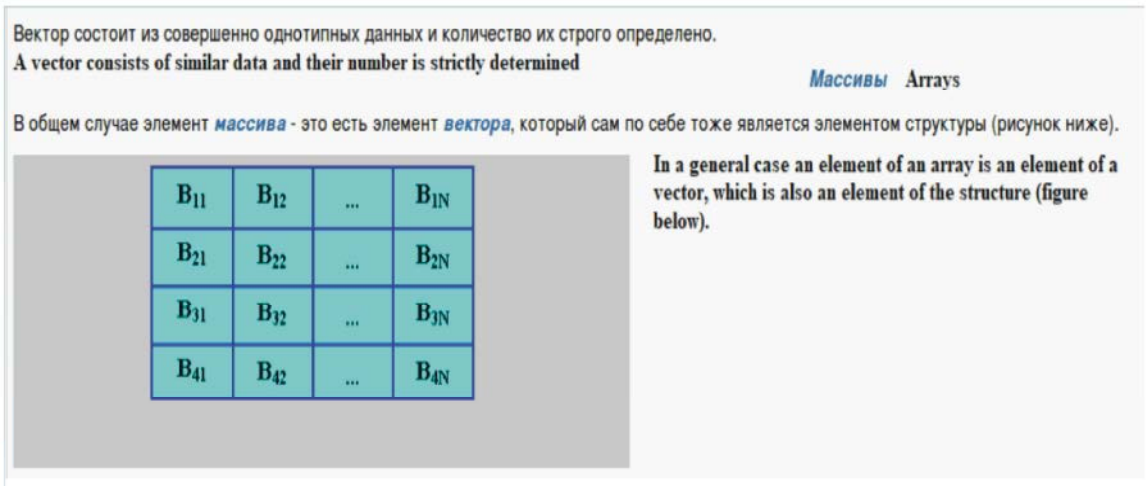


Fig. 11: Example of references to the basic concepts and terminology

**5th Implementation Phase. Creation of Test Elements Using jQuery:**

```
<li class="ans">Îòâåò 3</li>
<li class="ans">Îòâåò 4</li>
</ul>
</div>
</div>
```

The number of units with the class quest next step is the creation of a framework for test questions. To do this, you want to create a block for example in the tutorial uses an HTML page, then you can extend their functionality with jQuery and JavaScript. To do this, you need to connect to the file containing the page, using jQuery library type of construction:

```
<Script type = "text / javascript" src = "jquery-2.1.1.min.js"> </ script>
```

And to put the library itself in the same directory where is required by us page.

In order to put to all the questions, a separate block for each test question, headers, which contain a question and lists that contain variants of the answer. For each of these classes ask for further processing using CSS.

The entire structure is as follows:

```
<Divid = "questions">
<H2 align = "center"> Test </ h2>
<Div class = "quest">
<H3> Question </ h3>
<Ul>
<Li class = "ans"> A 1 </ li>
<Li class = "ans"> Answer 2 </ li>
```

It corresponds to the number of test questions. Now using jQuery and JavaScript which is required to process a click on an answer option in order to find out the correct answer or not, count the number of correct answers, to hide the block to the current issue and the next show. At the end you want to show the user the number of correct answers and recommendations for further action.

This requires you to have two variable- bally = 0 = 0 and the nom. Partly bally - counter, which is increased by one if the answer is correct, zero if incorrect. Partly nom indicates the number of current issues and increases when you click on the answer.

Since you only need to display the current question, initially all the blocks with the questions will be hidden. For these purposes, use CSS, to display function will be used jQueryFadeIn and FadeOut.

CSS code:

```
<Style type = "text / css">
.quest {
display: none;
}
#questions {
width: 100%;
height: 300px;
background-color: #CCCCCC;
padding: 20px;
}
.ans: hover {
color: green;
}
.ans {
padding: 10px;
cursor: pointer;
}
</ Style>
#questions
```

Selector determines the overall look of the test unit (gray background, full width, spacing, height). Selectors .ans: hover and .ans define the appearance of possible answers (green hover, black in the standard state, the hand cursor). Selector. Quest hides all questions.

The JavaScript code

```
<Scripttype = "text / javascript">
var bally = 0;
var nom = 0;
$ (Document) .ready (function () {
$ ( "Quest.") Eq (nom) .show ().;
})functionans(b){
bally=bally+b;
$(".quest").eq(nom).fadeOut(300);
nom=nom+1;
$(".quest").eq(nom).fadeIn(300);
}
</script>
```

The function $ (document) .ready (function () when the page loads selects the first block of all hidden blocks quest class and displays it. Ans (b) function takes the parameter b and adds it to the points, hides the current issue, increase the current number question by one and shows it to fire a function necessary to tie it to the event click on the element with class ans (answers) to the correct answer feature will send 1 for wrong - 0.Primer..:

```
<Li class = "ans" onClick = "ans (0);"> odinindeks </ li>
<Li class = "ans" onClick = "ans (1);"> paryindeksov </ li>
<Li class = "ans" onClick = "ans (0);"> trehindeksov </ li>
<Liclass = "ans" onClick = "ans (0);"> four members </ li>
```

It remains to process test results and display them to the user. To do this, create two sets of responses to the user. In one of them displays a message when passing a threshold of points, in the second output messages when an unsatisfactory result of testing.

```
<div id="test_end1">
You scored <span id = "bl1"> </ span> out of 5 <br>

We encourage you again to read the subject <b> <i>
<sc_elements_idtf = 'current lecture'> current lecture
</ sc_element> </ b> </ i>
</ Div>
<Div id = "test_end2">
You scored <spanid = "bl2"> </ span> out of 5 <br>
You can study the following subject <b> <i>
<sc_elements_idtf = 'following the lecture'> next
lecture </ sc_element> </ b> </ i>
</ Div>
if(bally<3) $("#test_end1").fadeIn(300);
if(bally>2) $("#test_end2").fadeIn(300);
document.getElementById
```

(In the first block, the message is displayed when the test is failed. The elements <spanid = "bl1"> </ span> and <spanid = "bl2"> </ span> shows the number of points. Below is a link to the recommended material for the study. To display the number of points and recommendations on, the end of the test, you must add a line to the script:

```
if(nom==5){
'bl1').innerHTML=bally;
document.getElementById('bl2').innerHTML=bally;
}
}
```

And hide the units with the recommendations of using CSS:

```
#test_end1{
display:none;
}
#test_end2{
display:none;
}
```

Sample test questions in accordance with Figure 12.

**6th Implementation Phase. Section "Glossary" Assembly:** Glossary is the node containing all the terms included in the lecture. To do this, all the secondary nodes, the existing system should be added to the node by the structure.

```
<= Nrel_section_decomposition.
```

The interface of the glossary is shown below in accordance with Figure 13.

**7th Implementation Phase. Under "Final Test" Assembly:** Stage is similar to step 5. Differences in the pattern of building blocks (for the convenience of using block elements instead of headers and lists).

```
<Div class = "quest"> Question1
 <Div class = "ans" onClick = "ans (0);"> A 1 </ div>
 <Div class = "ans" onClick = "ans (1);"> A 2 </ div>
 <Div class = "ans" onClick = "ans (0);"> A 3 </ div>
 <Div class = "ans" onClick = "ans (0);"> A 4 </ div>
 </ Div>
```

Instead, the recommendations in the study displayed a single block with the number of points:

```
<Div id = "test_end1">
Vynabrali <span id = "bl1"> </ span> 65 balloviz <br>
</ Div>
JavaScript code:
<Script type = "text / javascript">
var bally = 0;
var nom = 0;
$ (Document) .ready (function () {
$ ( "Quest.") Eq (nom) .show ().;
})
functionans (b) {
bally = bally + b;
$ ( "Quest.") Eq (nom) .fadeOut (300).;
nom = nom + 1;
setTimeout (function () {
$ ( "Quest.") Eq (nom) .fadeIn (300).;
if (nom == 65) {
$ ( "# Test_end1") fadeIn (300).;
document.getElementById ( 'bl1') innerHTML =
bally.;
}
}, 400);
}
</ Script>
```

Set timeout function performs the delay before displaying the next issue to avoid "spikes" of the text during the change of subject.

Navigating through the Tutorial: Navigating the electronic textbook realized by means of links, the top menu and sidebar stories queries.

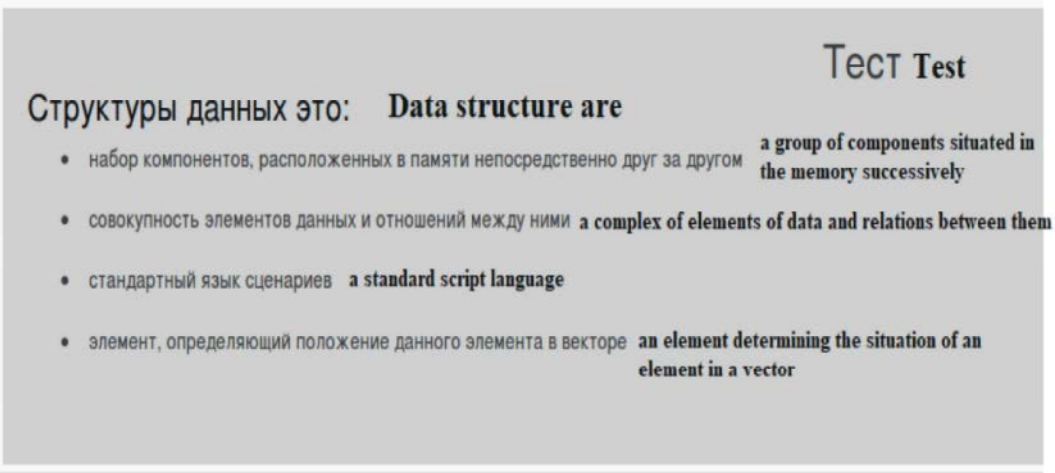


Fig. 12: Sample test questions


Fig. 13: Glossary Interface

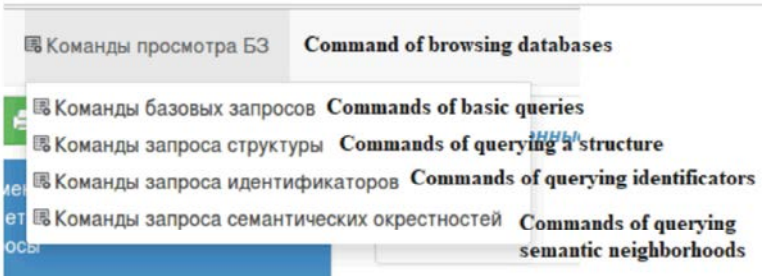


Fig. 14: Example of finding a lecture on the parent node

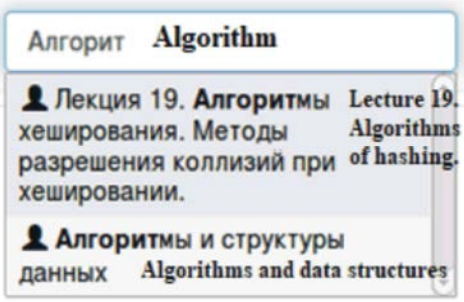


Fig. 15: Navigational search engine

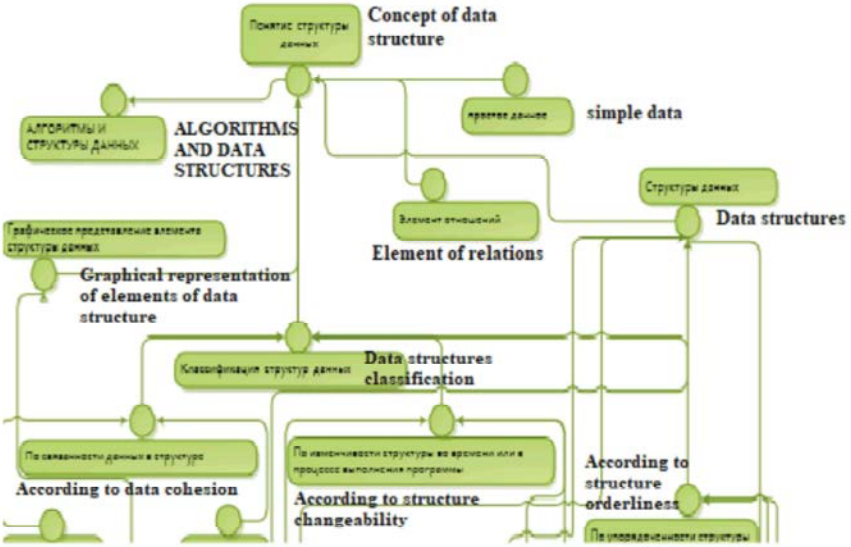


Fig. 16: Presentation of the educational benefits in the form of a connected graph

All these sections - contain a set of data referring to each other (linked nodes), so you can easily make the transition from section to section, search terms and data.

The top menu is a menu of OSTIS core functionality. Since all of our forums, terms and other elements of a textbook represent nodes, we can use the core functionality to monitor the communication nodes. For example, find the lecture which contains the desired term (the parent node) with accordance to Figure 14.

To navigate through and also search in accordance with Figure 15. One advantage OSTIS core is accelerated processing search requests. As a result, we get a live search of the necessary data to us.

On the basis of textbook structure, we will develop a semantic model of knowledge representation in accordance with Figure 16. We divide each part of the training manual on units and present it as a connected graph.

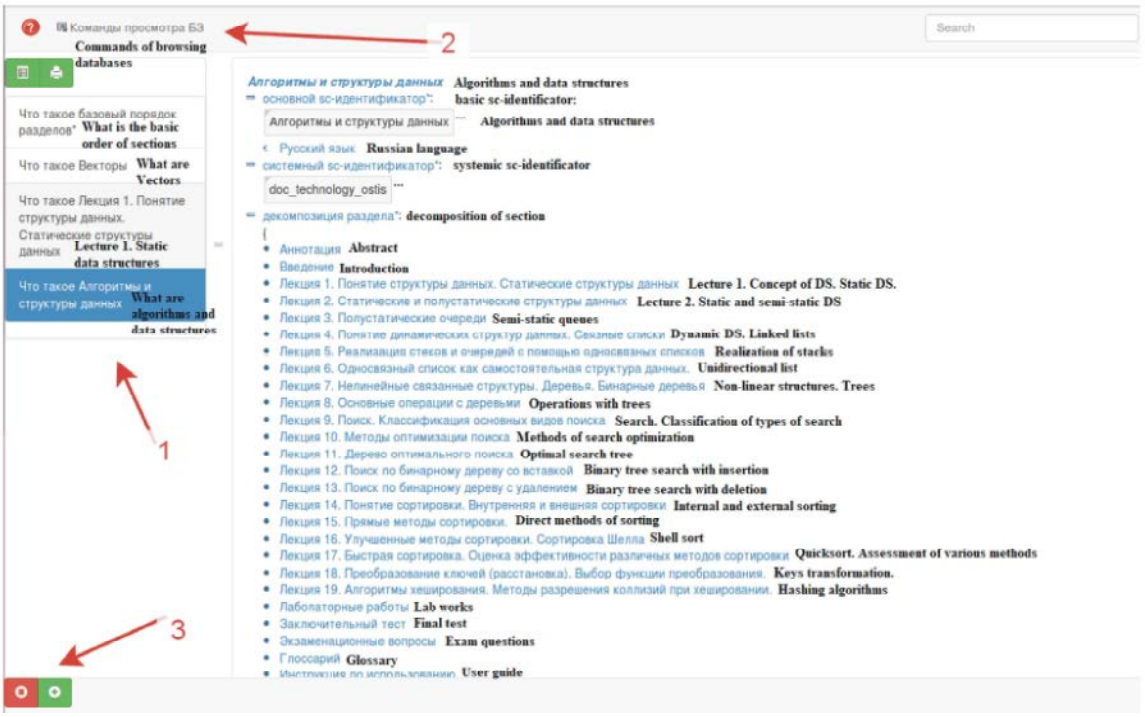


Fig. 17: The navigation controls in the core OSTIS
1. Sidebar; 2. Top menu; 3. Lower secondary menu

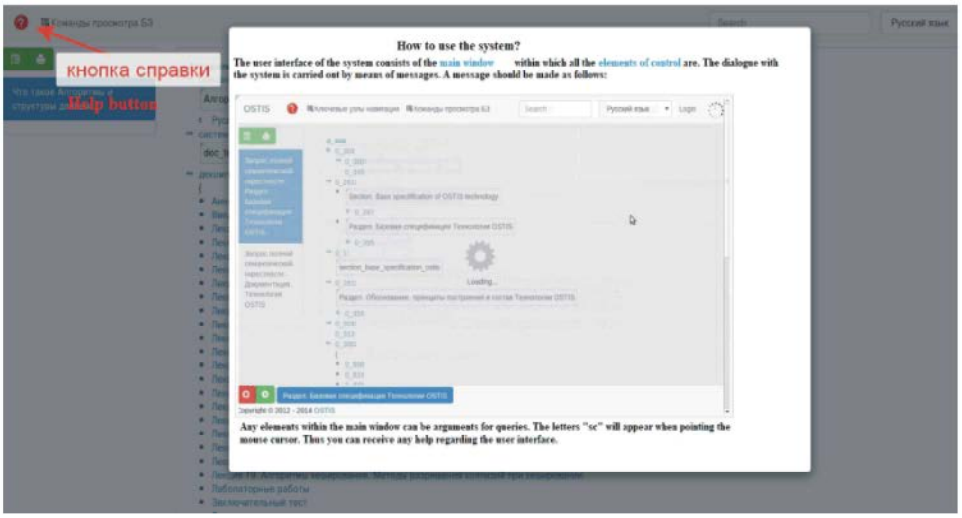


Fig. 18: Help Window to OSTIS system

To fast transitioning and finding more information in OSTIS kernel has three navigation element in accordance with Figure 17.

OSTIS system contains help with animated instructions on how to use the system in accordance with Figure 18. Reference is caused at the first visit or click on the Help button.

## CONCLUSION

One of the most important trends is the use of computer technology in education. This article reflects the implementation of intelligent electronic educational publications on the subject "Algorithms and Data Structures" on the basis of OSTIS technologies. In the

development of intelligent electronic educational publications we used active development at present with a comprehensive open (OSTIS - Open Semantic Technology for Intelligent System) technology to design intelligent systems. At the heart of OSTIS technology, it uses as a way of describing the unified semantic knowledge of networks with the basic set-theoretic interpretation of their elements. This method of knowledge representation is called SC-code (Semantic Code) and semantic networks, presented in SC-code named sc- graphs.

Electronic textbooks provide the opportunity to see inaccessible to the usual eye and touch the mysteries of nature, the results of various physical, chemical and biological experiments.

Thus, innovative technologies contribute to the development of educational, scientific and professional competence of students.

Effective teachers use innovative technologies in the educational process is aimed at improving the educational level of young people - the future of highly competitive professionals.

The selected technology, given a set of models, tools and techniques implemented in the form of intellectual metasystem is formed by the same proposed technology design of intelligent systems and following from this so-called metasystem IMS.OSTIS (Intelligent Metasystem of Open Semantic Technology for Intelligent System). This means that the range of models, tools and techniques used for the continuous updating and improvement of the proposed technology is none other than a set of models, means and methods of continuous updating and improvement of the said intellectual metasystem, which can be used not only for the development and maintenance of this metasystem, but also any other intelligent systems formed on the basis of the technology used.

## REFERENCES


1. Golenkov, V.V., N.A. Gulyakin, I.T. Davydenko, D.N. Koronchik and D.V. Shunkevich, 2014. Semantic Technology for Intelligent Systems Design. Information systems and technology. Mathematical modeling. Bulletin KrNU named after Mykhailo Ostrohradskyi, 5(88).
2. Golenkov, V.V., V.V. Emelyanov and V.B. Tarasov, 2001. Virtual department and intelligent tutoring systems. News of Artificial Intelligence, 4: 3-13.
3. Belyaev, M.I., V.V. Grinshkun and G.A. Krasnov, 2006. Technology for creating e-learning. Development of the Institute of Distance Education of the Russian University of Friendship of Peoples, pp: 25.
4. Turabaeva, L.K., 2013. Kazakh language. The computer program. SKSU named after M. Auezov, Center of Electronic Educational Publications, pp: 110.
5. Turabaeva, L.K., 2012. Professional Kazakh language. Ebook, pp: 108.
6. Abubaeva, A., 2006. Use of electronic textbooks. Fundamentals of Computer Science, 4: 98.
7. Turabaeva, L.K., G.A. Abdikerimova, G.U. Tanabaeva and L.A. Irsimbetova, 2015. E-books and their application. Education. Pedagogy. International Journal of Experimental Education, pp: 7.
8. Krivosheev, A.O., 1996. The development and use of computer training programs. Information Technology, 2: 14-18.
9. Shunkevich, D.V., 2014. Principles of interpretation of the language program, based on the processing of knowledge bases. NB: Cybernetics and programming, 5: 1-13.
10. Shunkevich, D.V., 2014. The machine processing of knowledge of intellectual metasystem support design of intelligent systems. Open semantic technologies of intelligent systems (OSTIS-2014). In the Proceedings of the International educ.-tech. Conference. Minsk, pp: 93-96.
11. Litvintseva, L.V., 1996. Virtual worlds in education systems. In the Proceedings of the V National Conference with international participation "Artificial Intelligence - 96". Kazan, pp: 183-186.
12. Brusilovskiy, P.L., 1990. Intelligent tutoring systems. Computer Ser. Information Technology. Facilities and Systems, 2: 3-23.