

---

Международный электронный научно-практический журнал  
**«Вопросы филологии и межкультурной коммуникации: научные  
исследования и практические решения»**

International Scientific and Practical Journal  
"Studies in Philology and Cross-cultural Communication"

График выхода: 4 раза в год

Языки: русский, английский

Свидетельство о регистрации: ЭЛ № ФС 77-68838 от 28 февраля 2017 года

Учредитель/издатель: Автономная некоммерческая организация "Международный центр продвижения русского языка и культуры "ЦЕНТРУС"

Официальный сайт: [www.vfmk.ru](http://www.vfmk.ru)

Прием статей по эл.адресу: [vfmk@centruscom.ru](mailto:vfmk@centruscom.ru)

Место издания: Российская Федерация, г. Волгоград

---

**Цель издания** – поддержка молодых ученых, популяризация и знакомство широкой общественности с новейшими научными исследованиями в области филологии и межкультурной коммуникации, создание единой площадки по обмену научным опытом между российскими учеными и их зарубежными коллегами, занимающимися проблемами межкультурной коммуникации и развития русского языка.

---

Куприева И.А., Бойко А.И., Чернышова А.И. Разработка виртуального словаря лексических единиц, номинирующих процесс познания [Электронный ресурс] // Вопросы филологии и межкультурной коммуникации: научные исследования и практические решения: электрон. научн.-практич. журн. 2017. N 1 (1). URL: <http://vfmk.ru/journal/01/017.html> (дата обращения: 10.07.2017).

УДК 81.33+004.42

**Куприева Ирина Анатольевна,**

д-р филол. наук, профессор,  
Белгородский государственный национальный  
исследовательский университет,  
Белгород, Россия

**Бойко Анатолий Игоревич,**

студент,  
Белгородский государственный национальный  
исследовательский университет,  
Белгород, Россия

**Чернышова Алёна Игоревна,**

студентка,  
Белгородский государственный национальный  
исследовательский университет,  
Белгород, Россия

E-mail: chernyshovaalyona@mail.ru

## **РАЗРАБОТКА ВИРТУАЛЬНОГО СЛОВАРЯ ЛЕКСИЧЕСКИХ ЕДИНИЦ, НОМИНИРУЮЩИХ ПРОЦЕСС ПОЗНАНИЯ**

**Аннотация:** Данная статья описывает процесс разработки программы для ЭВМ, которая предназначена для поиска информации о лексических единицах, номинирующих процесс познания в английском, русском, украинском и французском языках, а именно результатов их лексикографического, контекстуального и концептуального анализов в соответствии с установленными пользователем параметрами фильтрации.

**Ключевые слова:** виртуальный словарь, лексическая единица, лексикографический анализ, контекстуальный анализ, концептуальный анализ, форма программы.

**Kuprieva Irina Anatolyevna,**

Doctor of Philology, Professor,  
Belgorod State National Research University,  
Belgorod, Russian Federation

**Boiko Anatoly Igorevich,**

student,  
Belgorod State National Research University,  
Belgorod, Russian Federation

**Chernyshova Alyona Igorevna,**  
student,  
Belgorod State National Research University,  
Belgorod, Russian Federation  
E-mail: chernyshovaalyona@mail.ru

## **VIRTUAL DICTIONARY DEVELOPMENT OF LEXICAL ITEMS NOMINATING COGNITIVE PROCESS**

**Abstract:** This article deals with process of program development for ECM, which is designed for search of information about lexical items nominating cognitive process in English, Russian, Ukraine and French as follows: lexicographical, contextual and conceptual analysis results in accordance with filtering parameters chosen by user.

**Keywords:** virtual dictionary, lexical item, lexicographical analysis, contextual analysis, conceptual analysis, program form.

В настоящее время наблюдается тенденция увеличения использования компьютерной обработки лексикографических данных вместо традиционных методов. Использование компьютера позволяет не только экономить ресурсы и время, но и благоприятно сказывается на качестве получаемой информации, а также отсеивать большую часть ошибок.

Процесс компьютеризации лексикографической деятельности заключается в первую очередь в создании баз данных, в которых материал сгруппирован в соответствии с определенными требованиями. Помимо этого, в данный процесс входят такие составляющие как разработка программных средств, позволяющих обрабатывать полученные данные, а также отвечающих за визуальное представление информации в группированном виде.

Благодаря использованию компьютера значительно упрощается процесс поиска требуемой лексической единицы в базе данных, помимо этого, создание новых программных продуктов, реализующих поиск, отбор и отображение данных из баз данных, вместо использования уже готовых решений, позволяет увеличить качество отбора информации посредством узкой направленности каждого разрабатываемого модуля.

Таким образом, актуальность данной работы заключается в необходимости создания программного продукта, который позволит обрабатывать лексемы, собранные в базе данных и будет отвечать за представление этой информации для пользователя в наиболее удобном формате и в соответствии с заданными пользователем условиями отбора лексем.

Основной целью представленного исследования является освещение процесса разработки программных продуктов, отбора лексических единиц, а также непосредственно создание программного продукта, который, работая с занесенными в базу данных лексемами, позволит пользователю в как можно более короткие сроки получить как можно более полную информацию по каждой из лексем, с учетом его предпочтений, задаваемых при помощи параметров фильтрации. Таким образом, будет создан виртуальный словарь на основе базы данных.

Настоящее исследование позволит получить инновационные сведения о семантике и особенностях функционирования в предложении лексем, номинирующих процесс познания в английском, русском, украинском и французском языках.

Языковой материал, представленный в данном исследовании в качестве фактического, был отобран посредством сплошной выборки из соответствующих аутентичных источников. В частности были рассмотрены доступные интернет ресурсы, толковые словари и художественные произведения.

Отбор релевантных исследованию единиц проводился первично, т.е. посредством сплошной выборки, согласно способности единицы функционально или системно обозначать ситуацию познания в *синкретизме* (единство процесса познания и его результата) и/или *дискретности* (каждый из процессов получения знания в отдельности).

Однако формирование исследовательского тезауруса на следующем этапе проводилось по принципу уточнения его содержательного показателя. С этой целью, то есть с целью верификации полученных данных, проводится лексикографический анализ на предмет способности лексемы системно описывать рассматриваемый референт. Для этого выявлялись все возможные

дефиниции единицы, и отслеживалась прямая способность номинации ситуации когнициии. Иными словами, лексема проверяется на соответствие критерию «способность номинировать когнициию на системном уровне». Дальнейшая верификация происходила в корреляции с выявленным ранее концептуальным содержанием искомой ментальной структуры. Это дословно означает, что все с учетом вышперечисленных параметров та или иная лексема должна на системном и/или функциональном уровне актуализировать такие концептуальные составляющие как «внимание», «перцепция», «ментальная деятельность», «память», «воображение».

В качестве среды разработки для создания данного программного продукта была выбрана Borland C++ Builder, которая представляет собой совершенную интерактивную среду программирования на языке C++. Система обеспечивает высокую продуктивность и производительность, удовлетворяя современным требованиям в разработке приложений под Windows.

Следующим шагом после выбора языка программирования и среды разработки стала разработка алгоритма реализации виртуального словаря на основе баз данных:

- 1) запуск программы;
- 2) открытие формы поиска;
- 3) выбор необходимого языка и параметров фильтрации;
- 4) ввод искомой лексемы в строку поиска или ее выбор из списка;
- 5) просмотр информации о запрашиваемой единице в соответствии с параметрами фильтрации.

При разработке алгоритма необходимо не только описать функции, которые будет выполнять программа, но и рассчитать необходимое количество форм, используемых для отображения компонентов управления программой, и повышающих простоту её использования.

Разрабатываемая программа должна отвечать следующим требованиям:

- 1) обеспечивать вывод необходимой информации по искомой лексеме;
- 2) обеспечивать пользователя возможностью настраивать параметры фильтрации;

- 3) гарантировать невозможность изменения пользователем записей в базе данных;
- 4) исключать отображение излишней информации, используемой исключительно для межтабличных связей и фильтрации данных;
- 5) иметь справочный раздел, содержащий информацию о работе с программным продуктом.

В связи с данными условиями, было принято решение в необходимости создания трех форм, каждая из которых будет отвечать определенным требованиям, реализуя часть задуманного функционала.

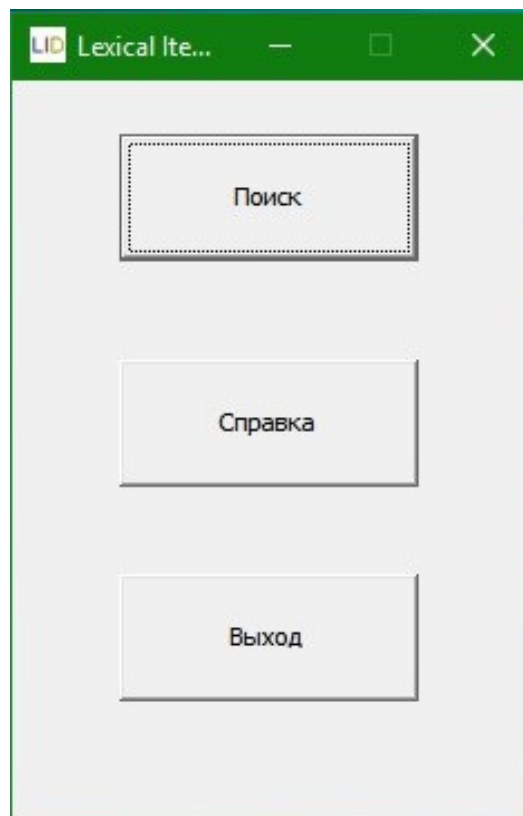


Рис. 1. Форма главного меню

На первой форме (Рис. 1) были размещены три компонента Button, которые в дальнейшем будут отвечать за переход на соответствующие формы поиска и справки или выход из приложения.

На второй форме (Рис. 2) были размещены следующие компоненты:

- DBGrid – таблица, отображающая информацию содержащуюся в определенной таблице в базе данных;

- четыре компонента Label, предназначенных для вывода на форму текста, который выполняет роль подписи для блоков переключателей;
- четыре компонента RadioButton обеспечивающих выбор одной опции из определенного набора;
- четыре компонента CheckBox предназначенных для задания пользователем параметров фильтрации выводимой программой информации;
- компонент Edit, который используется для ввода пользователем какого-либо текста с клавиатуры;
- компонент RichEdit, выводящий на экран текст, который содержит информацию по искомой лексеме на основе установленных пользователем параметров поиска.

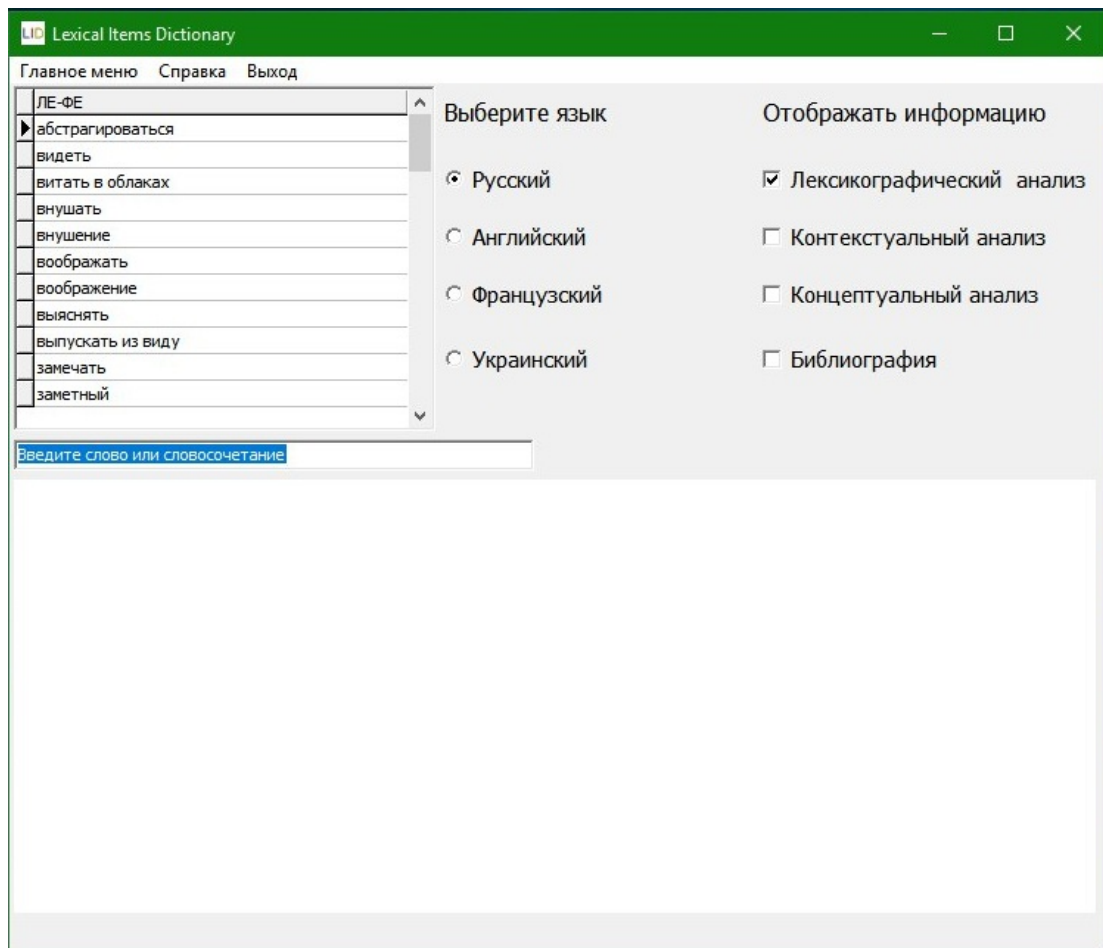


Рис. 2. Форма поиска

Переход на данную форму осуществляется при нажатии соответствующей кнопки в главном меню программы.

Для упрощения работы пользователя с большим объемом данных, содержащихся в таблице, было решено использовать динамическую фильтрацию. За реализацию данной функции отвечает блок программного кода, приведенный ниже (Листинг 1).

```
void __fastcall TForm2::Edit1Change(TObject *Sender)
{
    String text = "";
    text = Edit1->Text;

    if (RadioButtonRu->Checked) {

        if (text == "") {
            DataModule6->TableRussianLEFE->Filtered = false;
        }
        else{
            DataModule6->TableRussianLEFE->Filtered = true;
            DataModule6->TableRussianLEFE->Filter = "(ЛЕ-ФЕ like " +
QuotedStr(text+'*') + ")";
        }
    }
}
```

Листинг 1. Программный код, отвечающий за функцию динамической фильтрации лексем

Данный код вызывается при выполнении события OnChange, что соответствует концепции динамической фильтрации. Создается строковая переменная text в которую далее поступает введенный в поле Edit набор символов, извлекаемый из него при помощи свойства Text. Условный оператор if проверяет истинность утверждения RadioButtonRu->Checked, и используется для того, чтобы использовать подходящую таблицу. В случае если данное утверждение окажется ложным, будет проверяться истинность утверждения RadioButtonEn->Checked, которое имеет схожий с приведенным ниже программный код и отличается только целевой для фильтрации таблицей. В случае истинности первого утверждения, выполняется проверка соответствия переменной text пустой строке, в случае если значение данной переменной является пустым, свойству Filtered компонента TableRussianLEFE присваивается значение false, что отключает любую фильтрацию данной таблицы и обеспечивает вывод полной таблицы. В случае если в поле Edit был введен



символ, свойству Filtered присваивается значение true, после чего свойству Filter, присваивается строка содержащая имя колонки, в данном случае ЛЕ-ФЕ, оператор like, реализующий поиск по заданному шаблону, и переменная text помещенная в функцию QuotedStr, отвечающую за правильное взаимодействие с базами данных и помещающее значение переменной text в одинарные кавычки. Данная функция необходима в виду определенных требований к оформлению запросов, поступающих в свойство Filter. Астериск в данном случае используется для описания поискового запроса, допускающего любое количество символов после введенного пользователем. Благодаря этому у пользователя появляется возможность не вводить искомую лексему полностью, а выбрать её из списка при достаточном сужении области поиска.

Далее рассмотрим алгоритм, реализующий фильтрацию данных на основе указанных пользователем параметров и их последующий вывод в поле RichEdit.

```
void __fastcall TForm2::DBGrid1CellClick(TColumn *Column)
{
    RichEdit1->Clear();

    if (RadioButtonRu->Checked) {

        //Запоминаем код лексемы
        LECode = DataModule6->TableRussianLEFE_->AsString;
```

## Листинг 2. Получение кода лексемы

Как видно из приведенного выше фрагмента программного кода (Листинг 2), процесс вывода на экран требуемой лексемы происходит при условии выполнения события OnCellClick, компонента DBGrid. Использование метода Clear, делает просмотр информации по нескольким лексемам более простым, поскольку очищает компонент RichEdit от результатов, полученных при выборе предыдущей лексемы. После чего выполняется проверка истинности высказывания RadioButtonRu->Checked, от результатов которой зависит набор таблиц используемых для поиска информации. Если утверждение истинно, то объявленной в заголовочном файле переменной LECode присваивается значение

поля, скрытого от пользователя и содержащего код лексемы. Далее, как видно ниже (Листинг 3), условным оператором `if` выполняется проверка истинности утверждения `LekBox->Checked`, по умолчанию имеющего значение `true`. После чего выполняется фильтрация таблицы, содержащей результаты лексикографического анализа, в соответствии с полученным ранее кодом лексемы, выполняющей в таблице роль ключевого поля. Сам процесс фильтрации, как и программный код, отвечающий за его реализацию, был описан ранее и является модификацией алгоритма реализующего динамическую фильтрацию лексем при их вводе.

```
if (LekBox->Checked) {
    DataModule6->TableRussianLEK->Filtered = true;
    DataModule6->TableRussianLEK->Filter = "(Код_ЛЕ like " +
QuotedStr(LECode) + ")";
    RichEdit1->SelAttributes->Color = clHotLight;
    RichEdit1->SelAttributes->Size = 16;
    RichEdit1->Lines->Add("Лексикографический анализ");
    while (!DataModule6->TableRussianLEK->Eof)
    {
        RichEdit1->Lines->Add
            (DataModule6->TableRussianLEK->Fields-
>FieldByName("Результат")->AsString);
        DataModule6->TableRussianLEK->Next();
    }
}
```

Листинг 3. Фильтрация и вывод результата лексикографического анализа

При помощи свойства `SelAttributes` и его свойств `Color` и `Size` выводимая программой информация приобрела более удобный для чтения формат, поскольку различные виды анализов отделены от остального текста путем увеличения размера шрифта и изменения цвета текста. В связи с тем, что некоторые лексические единицы имеют несколько значений, возникла необходимость в использовании цикла `while` и свойства `Eof` компонента `ADOTable`. Данное свойство возвращает значение `true`, когда достигает конца списка, описанный внутри цикла алгоритм отвечает за вывод результатов лексикографического анализа. При помощи функции `Add` компонента `RichEdit`, на него выводятся значения полей из колонки «Результат», в формате `AnsiString`, после чего, при помощи функции `Next` осуществляется переход к следующей

записи в таблице. Благодаря использованию функции FieldByName, и указанием в ней имени колонки, выводятся только значения поля «Результат».

Поскольку программный код, отвечающий за фильтрацию и вывод результатов других видов анализа, аналогичен данному, за исключением названия таблиц, участвующих в выборке, его детальный разбор не приводится.

На следующей форме (Рис. 3) размещен один компонент – RichEdit – отображающий текстовый файл, содержащий в себе пояснения касаяемо данного программного продукта в целом. Данный компонент позволяет выводить текст, учитывая его особенности форматирования.

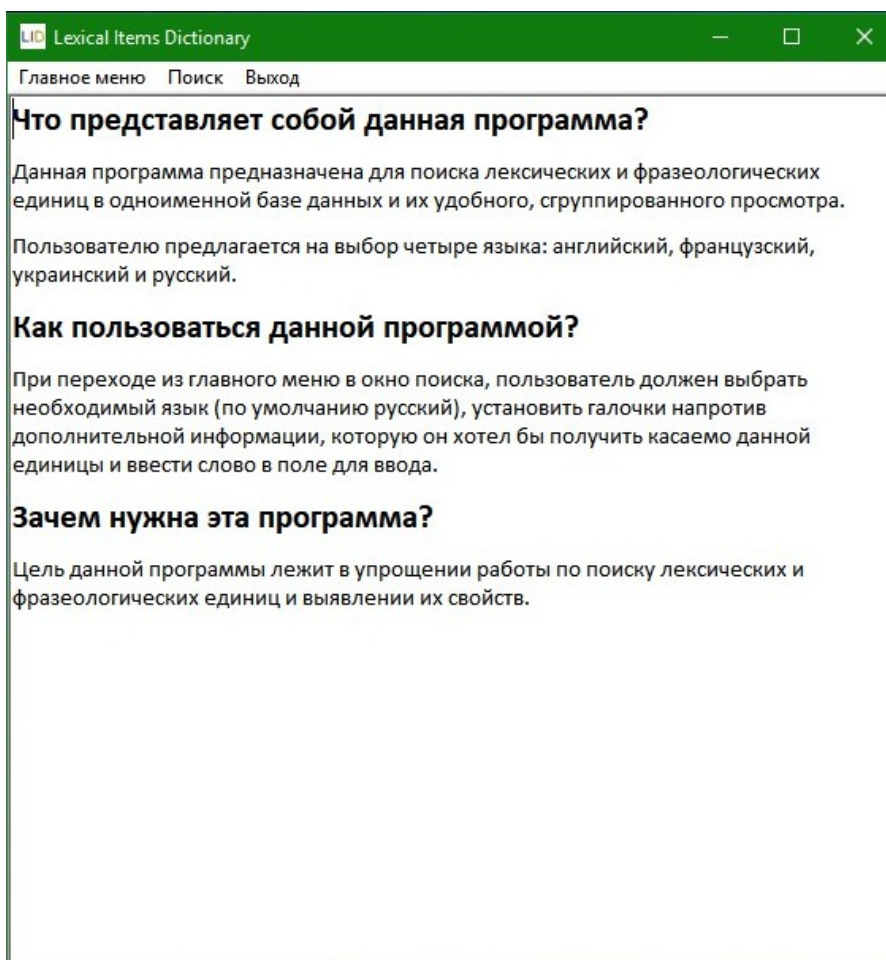


Рис. 3. Форма справки

После завершения процесса разработки можно приступать к апробации готового программного продукта.

В ходе тестирования было выделено несколько недостатков, которые были в дальнейшем устранены. Финальное тестирование показало, что программа полностью отвечает установленным требованиям и готова к использованию.

Далее разработанное приложение было упаковано в установочный файл, для повышения удобства его установки.

В результате вышеперечисленных действий был создан удобный в использовании программный продукт – виртуальный словарь на основе баз данных, позволяющий пользователю получать необходимую информацию о лексеме, не тратя при этом время на создание запросов и сопоставление значений в различных таблицах, используя такие СУБД как Microsoft Access.

#### **Список литературы:**

1. Послед Б. С. Borland C++ Builder 6. Разработка приложений баз данных. – DiaSoft, 2003.
2. Куприева И. А. Механизмы вербализации ментальных структур лексемами со значением «психические процессы» в англоязычном дискурсе //Дис.... док. филолог. наук. Белгород. – 2014.

*Дата публикации: 10.07.2017*