

## Список литературы

1. Батырканов Ж.И. Обучающие экспертные системы// Известия КГТУ.-Бишкек. – 2011.-№ 21. – С.129-131.
2. Батырканов Ж.И. Модели представлений знаний на основе приближенного множества// Вестник науки Костанайского социально – технического университета им. академика Зулхарнай Алдамжар 3/2014-с.35-39.
3. Боскебеев К.Дж. Модель интеллектуальной обучающей системы на основе теории систем // Материалы международной научно-практической конференции, посвященной 70 – летию образования Волгоградского государственного аграрного университета. г. Волгоград 2014г. – с.331-336.
4. Боскебеев К.Дж. Информационная система управления на основе метода нечеткого множества // "Вестник Таджикского технического университета" – январь- март 2014г. – с.45-49.
5. Расим Д. Автоматизированная обучающая система// Известия КГТУ. –Бишкек. – 2013.- № 32 (часть I). – С.145-146.
6. Расим Д. Модель представления знаний организации// Вестник науки Костанайского социально – технического университета им. академика Зулхарнай Алдамжар 3/2014-с.64-68.
7. Питерсон Дж. Теория сетей Петри и моделирование систем. М.: Мир, 1984. 264 с.

## References

1. Batyrcanov Z.I. The educational expert systems // Proceedings of KSTU. Bishkek. - 2011.-№ 21. – page 129-131.
2. Batyrcanov Z.I. Model of knowledge representation based on the rough sets // Science Bulletin of the Kostanai Social - Technical University named after Academician Zulkharnai Aldamzhar 3/2014, page.35-39.
3. Boskebeev K.D. Intelligent tutoring system model based on the theory of systems // Proceedings of the international scientific-practical conference dedicated to the 70 - anniversary of Volgograd State Agrarian University. January 28 - January 28. Volgograd 2014. – page 331-336.
4. Boskebeev K.D. Management information system based on the method of fuzzy sets // "Bulletin of the Tajik Technical University" 1 (25) January-March 2014. – page 45-49.
5. Rasim D. automated training system .// Proceedings of KSTU. -Bishkek. - 2013. - № 32 (Part I). - page.145-146.
6. Rasim D. Knowledge Representation Model of Organization .// Science Bulletin of Kostanai Social - Technical University named after Academician Zulkharnai Aldamzhar 3/2014, page 64-68.
7. J. Peterson. The theory of Petri nets and modeling of the systems. M .: Mir, 1984. 264 page.

УДК:004.412.001.636004.891.004.822:61

### РАЗРАБОТКА ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ПРОЕКТИРОВАНИЯ ЭКСПЕРТНЫХ СИСТЕМ НА ОСНОВЕ СЕМАНТИЧЕСКИХ СЕТЕЙ В ОБЛАСТИ МЕДИЦИНЫ

**Уметалиев Азат.**, студент кафедры ИСЭ-1-12, КГТУ им.И.Раззакова, Кыргызской Республики (+996) 54-19-20, 720044, г.Бишкек, пр. Мира, 66, e-mail:Shotokan.Azat@mail.ru

**Ибраимов Чингиз Уланович.**, соискатель ,кафедры ИСЭ, КГТУ им.И.Раззакова, Кыргызской Республики (+996) 54-19-20, 720044, г.Бишкек, пр. Мира, 66, e-mail:ibraimov@turmedia.com

**Боскебеев Калычбек Джетмишбаевич**, доцент, к.т.н., зав. отделом науки и аспирантуры и докторантуры КГТУ им. И. Раззакова Кыргызской Республики (+996) 54-51-51. 720044, г.Бишкек, пр. Мира, 66, e-mail: [kboskebeev@mail.ru](mailto:kboskebeev@mail.ru)

В статье представлен подход к формированию базы знаний терапевта в медицине, к созданию экспертной системы для диагностики болезни пациента. Целью данной статьи является разработка программного продукта, задачей которого является хранения знаний специалиста. В результате разработке программы были разработаны инструментальные средства проектирования экспертных систем на основе семантических сетей, позволяющие создавать по определенным правилам базы знаний, используемые в данной экспертной системе, редактировать и пополнять уже существующие, а так же проводить консультацию. Можно просмотреть дерево заболеваний, а так же местоположение определенной болезни в нем. Разработанная система является продукционной. Системы продукций включают три основных компонента: базу правил, состоящую из набора продукций (правил вывода), базу данных, и интерпретатор для получения логического вывода. Так и в данной разработке можно выделить базу правил, которая описывается в программе в виде предиката rule (RuleNO, Category, Category, Conditions) - хранит правила вывода, база правил, описанная с помощью предиката cond (CondNO, STRING), интерпретатором в данном случае является машина логического вывода. Разработанный редактор позволяет осуществлять ряд манипуляций со знаниями.

**Ключевые слова:** медицинская диагностика, экспертная система, база знаний, модель представления знаний, продукционные правила, извлечение знаний из баз данных, ассоциативные правила.

## DEVELOPMENT TOOLS OF DESIGN EXPERT SYSTEMS BASED ON SEMANTIC NETWORKS IN FIELD MEDICINE

**Umetaliyev Azat**, student of the Department "ISE", KSTU named after I. Razzakov, 66 Mir avenue, 720044, Bishkek, Kyrgyzstan, e-mail: [Shotokan.Azat@mail.ru](mailto:Shotokan.Azat@mail.ru)

**Ibragimov Ch. U.**, the applicant of the Department "ISE", KSTU named after I. Razzakov, 66 Mir avenue, 720044, Bishkek, Kyrgyzstan, e-mail: [ibrainov@turmedia.com](mailto:ibrainov@turmedia.com)

**Boskebeev K.D.**, Ph.D., associate professor, Head of Department of Science and postgraduate and doctoral studies, KSTU named after I. Razzakov, 66 Mir avenue, 720044, Bishkek, Kyrgyzstan, e-mail: [kboskebeev@mail.ru](mailto:kboskebeev@mail.ru), tell: (996) 312 54-51-51

In this paper performance an approach to building the knowledge base in medicine therapist to create an expert system for the diagnosis of the patient. The purpose of this article is to develop a software product, which aims to storage specialist knowledge. As a result, program development tools have been designed designing expert systems based on semantic networks, allowing to create specific rules for the knowledge base used in this expert system, edit and add to existing ones, as well as consulted. You can see the tree diseases, as well as the location of a particular disease therein. The developed system is a production. Production systems consist of three main components: the rule base, consisting of a set of productions (rules of inference), a database, and an interpreter for inference. And in the development of the rule base can be identified, which is described in the program as a predicate rule (RuleNO, Category, Category, Conditions) - stores the rules of inference rule base, described by a predicate cond (CondNO, STRING), the interpreter in this case an inference machine. Designed editor allows realize manipulations of knowledge.

**Keywords:** medical diagnosis expert system, knowledge base, knowledge representation model, production rules, extracting knowledge from databases, association rules.

Следует разработать экспертную систему на основе продукционных правил. Программа должна предоставлять ряд сервисных услуг: консультация по выбранной базе знаний; пополнение выбранной базы знаний; построение дерева знаний; редактирование выбранного файла. Для эффективной системы оказания медицинской помощи населению могут использоваться различные методы, связанные с информационными технологиями [6]. Одним из них является применение новых инновационных медицинских экспертных систем для диагностики заболеваний преимущественно в тех случаях, когда врач вынужден делать заключение на основе информации о симптоматике заболеваний и не может оперативно получить результаты дополнительных исследований с использованием инструментальных средств. Информация о симптоматике может быть получена врачом, который ставит диагноз, в основном следующими способами: путем личного общения с пациентом, в том числе и с применением программных средств типа Skype, специализированных телемедицинских систем [8]; на основании информации, предоставленной пациентом; за счет изучения данных, внесенных пациентами в персональные медицинские информационные системы, размещенные на их домашних ПЭВМ или серверах медицинских учреждений [5].

Наиболее актуальными на сегодняшний день являются системы приобретения знаний, основанные на интеграции технологий, использующих знания экспертов и методы извлечения знаний из баз данных [1,2,3,4,7]. Существует несколько проблем реализации данного подхода: отсутствуют эффективные методики интеграции знаний, полученных из разнотипных источников: существующие методы приобретения знаний, как правило, обладают низкой адаптивностью и не подразумевают использование механизмов обратной связи для установления новых зависимостей и корректировки правил в базе знаний. Поэтому целью настоящей статьи была разработка программы на преодоление указанных трудностей.

**Разработка программного продукта, реализующего описанные способы.** Перспективным способом разработки программных средств для извлечения знаний специалиста в области медицины является объектно-ориентированный метод, который позволяет создать мощную интерактивную систему, способную к динамическим изменениям.

Для разработки экспертной системы были использованы следующие программные средства: язык программирования PROLOG; среда разработки Visual PROLOG 5. Подсистема была создана для работы под управлением операционной системы Windows XP.

Минимальные требования к аппаратным средствам выглядят следующим образом: процессор Pentium 2Гц; оперативная память – 2Гб; место на жестком диске - 40 Гб; CD-ROM 32-х; монитор - 15"; Принтер - HP LJ 6L или аналогичный ему по характеристикам.

Собственно программу авторы разработали в среде Visual PROLOG 5. Ее основная форма приведена на рис.1.

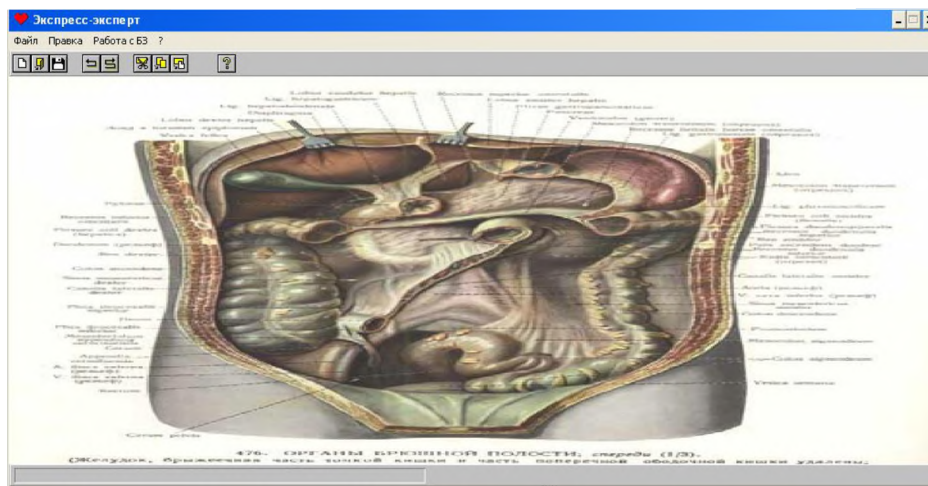


Рис. 1. Главная форма программы экспертной системы

В системе Prolog решение той или иной задачи получается путем логического вывода необходимых сведений из уже известных. При этом программа на Prolog не является последовательностью операторов, предписывающих выполнение определенных действий. Программа на Prolog представляет собой совокупность определенных фактов относительно рассматриваемой предметной области и правил для вывода логических заключений из этих фактов. Поэтому Prolog можно охарактеризовать как декларативный язык. Prolog использует модифицированный вариант синтаксиса языка логики предикатов. Необходимость такой модификации обусловлена желанием приблизить язык Prolog к естественному языку человека.

Экспертная система отсроченной консультации реализует следующие функции (см. рис.2): консультация; редактирование базы знаний, обмен сообщениями между инженером и врачом; дерево знаний; пополнение базы знаний.

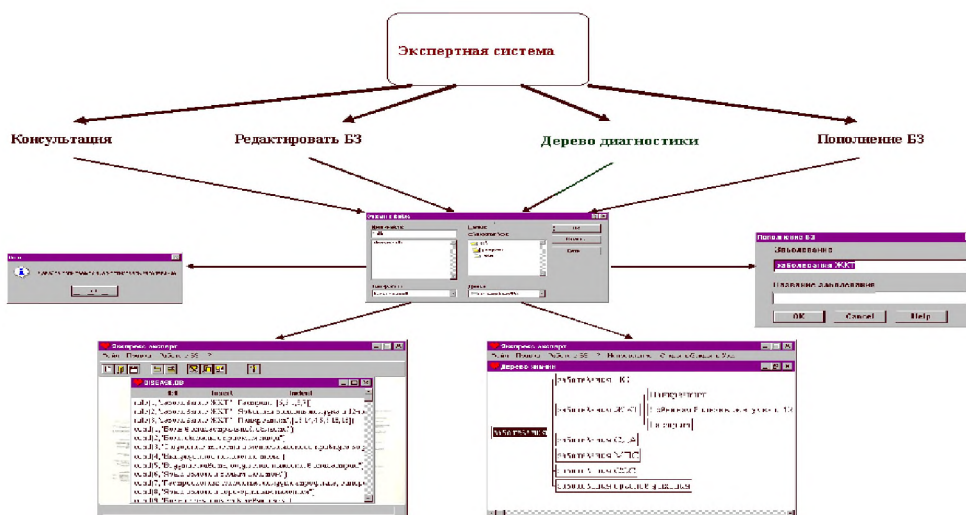


Рис. 2. Структура экспертной системы

При консультации используется файл базы знаний, в котором представлены факты и правила вывода тех или иных заключений. Взаимодействие с пользователем осуществляется с помощью диалоговых окон, посредством которых задаются вопросы. Выводы на основе ответов строятся с помощью логической машины вывода, используя свойство Visual Prolog поиск с возвратом (backtracking) см.рис.3:

**Машина логического вывода**

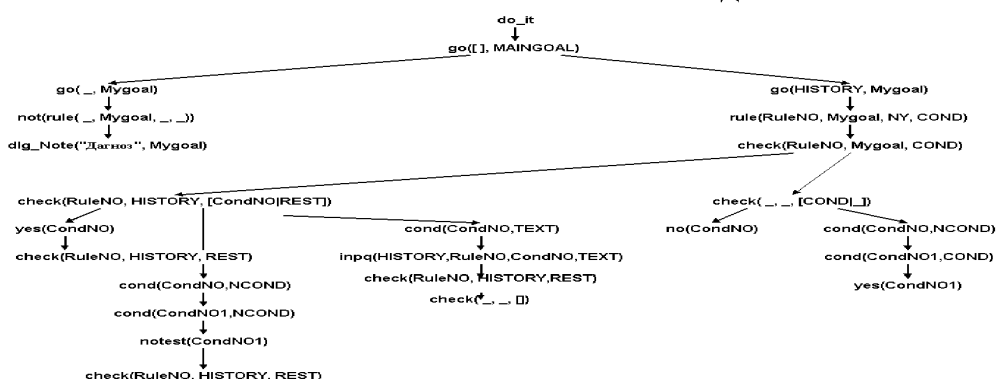


Рис. 3. Машина логического вывода

На рис.3. приведено взаимодействие множества подсистем в виде графа если все подсистемы будут истинны, тогда будет истинной сама система. На основе вышесказанного приводится фрагмент программ.

```

go( _, Mygoal ):-
    not(rule(_ ,Mygoal,_,_)),!,%Постановка окончательного диагноза
    concat("Я думаю, что это: ",Mygoal,Resault),
    dlg_Note("Диагноз",Resault),
    Answer = dlg_Ask("Вопрос","Я прав, не так ли?","&Да","&Нет"),
    evalans(Answer),!.
    
```

Вкратце опишем приведенный код программы:

(:-) - правило; (,) – логическая операция конъюнкция “AND”; (!) – отсечение (выход из подсистемы); (%) – комментарий; (.) – завершения подсистем.

Система -go( \_ , Mygoal ):- подсистема - not(rule(\_ ,Mygoal,\_,\_)),!, подсистема - concat("Я думаю, что это: ",Mygoal,Resault), подсистема – dlg\_Note("Диагноз",Resault)), подсистема - Answer = dlg\_Ask("Вопрос", "Я прав, не так ли?","&Да", "&Нет")),

В результате, если все подсистемы будут истинны, тогда будет истинной сама система - go( \_ , Mygoal ):-, то есть выполняется правило. Далее описаны другие системы.

```

go( HISTORY, Mygoal ):-
    rule (RuleNO,Mygoal,NY,COND), %Взять правило
    check (RuleNO,HISTORY, COND), %Проверить для нсго условия
    go ([RuleNO|HISTORY],NY) %Просмотреть другое правило
%Проверка условий для правила
check( RuleNO, HISTORY, [CondNO|REST] ):-
    yes(CondNO), !, %Ответ «Да», занести во временную базу данных
    check(RuleNO, HISTORY, REST).
check( _ , _ , [CondNO|_] ):-
    no(CondNO),!, fail. %Ответ «Нет», занести во временную базу данных
check( RuleNO, HISTORY, [CondNO|REST] ):-
    cond(CondNO,NCOND),
    fronttoken(NCOND,"не",_COND),
    frontchar(_COND,_COND),
    cond(CondNO1,COND), %Взять условие
    notest(CondNO1), !,
check(RuleNO, HISTORY, REST).%Проверить правило
check(_ , _ , [CondNO|_] ):-
    cond(CondNO,NCOND), %Взять условие
    fronttoken(NCOND,"не",_COND),
    frontchar(_COND,_COND),
    cond(CondNO1,COND), %Проверить правило
    yes(CondNO1), !, fail. %Ответ «Да» занести во временную базу данных хранения ответов пользователя
check( RuleNO, HISTORY, [CondNO|REST] ):-
    cond(CondNO,TEXT),!,
    inpq(HISTORY,RuleNO,CondNO,TEXT),
    check(RuleNO, HISTORY, REST).
    
```

```
check( _, _, [] ).
```

Для построения **дерева заболеваний** используется файл, содержащий набор правил построения. Далее эти правила используются при построении. Построение дерева описано с помощью определенных предикатов рекурсивных методов, а также описанных правил.

```
%Обнаружение нового листа дерева rev_treelist([],L,L).
```

```
rev_treelist([H|L1],L2,L3):-
    rev_treelist(L1,[H|L2],L3).
```

```
%Зайти в дерево
```

```
collect_tree(NODE,_,tree(NODE,unmarked,[],0)):-
    seen_node(NODE),!.%Обнаружение видимого узла
```

```
collect_tree(NODE,1,tree(NODE,unmarked,[],0)):-!.
```

```
collect_tree(NODE,N,tree(NODE,unmarked,RevList,0)):-
    findall(T,collect_tree1(NODE,N,T),TREEL),
```

```
%Предикат findall собирает список всех решений, возвращенных недетерминистическими предкатами
    rev_treelist(TREEL,[],RevList).
```

```
collect_tree1(NODE,N,TREE):-
```

```
    N1=N-1,
```

```
    includes(NODE,NODE1), %Факт, что лист дерева NODE содержит лист NODE1
```

```
    register_seen_node(NODE), %Регистрация найденного узла
```

```
    collect_tree(NODE1,N1,TREE).
```

```
collect_tree1(NODE,N,TREE):-
```

```
    N1=N-1,
```

```
    gen_from(NODE,NODE1), %Корень дерева NODE содержит ответвление NODE1
```

```
    register_seen_node(NODE), %Регистрация найденного узла
```

```
    collect_tree(NODE1,N1,TREE).
```

```
%Перечень видимых узлов
```

```
register_seen_node(NODE):-
```

```
    seen_node(NODE),!.
```

```
register_seen_node(NODE):-
```

```
    assert(seen_node(NODE)). %Занести во временную базу данных найденный узел.
```

**Пополнение базы знаний** о болезнях происходит в диалоговом режиме. Эксперту предлагается для болезней определенного класса добавить заболевание, а затем и записать для данного заболевания ряд характерных признаков. Обновление базы знаний происходит в два этапа. Сначала факты, описанные экспертом, добавляются в базу, которая хранится первоначально в оперативной памяти. Затем, если эксперт пожелает, то добавленные факты записываются в файл.

```
%Добавление фактов в базу данных
```

```
getnr(N,N):- not(rule(N,_,_)),!.%Если нет такого правила
```

```
getnr(N,N1):- H = N + 1,getnr(H,N1). %Увеличить номер для правила
```

```
readcondl([CondNO|R]):-
```

```
    Msg="Наберите условие, по которому произойдет переход от заболевания ЖКТ к его подвиду",
```

```
    InitStr="",
```

```
    Title="Симптом: ",
```

```
    CONDITIONS=dlg_GetStr(Title,Msg,InitStr),
```

```
    CONDITIONS >< "",!.
```

```
    getcond(CondNO,CONDITIONS),readcondl(R).
```

```
    readcondl([]).
```

```
getcond(CondNO,COND):-cond(CondNO,COND),!.%Формирование факта
```

```
getcond(CondNO,COND):- getbnr(1,CondNO),asserta(cond(CondNO,COND)). %Вставить факт
```

```
getbnr(N,N):- not(cond(N,_)),!.
```

```
getbnr(N,N1):- H = N + 1,getbnr(H,N1).
```

Редактирование файла осуществляется путем применения в программе как стандартных предикатов, так и набором предикатов, определенных автором проекта. Редактор позволяет выполнять набор стандартных действий:

```
- создать новый файл
```

```
    next_filename (FileName), %Создать файл с именем по умолчанию
```

```
    open_file (FileName,b_false), %Открыть этот файл
```

```
- открыть уже существующий
```

```
FileName = dlg_GetFileName("*. *",["*. *", "*. *"], "Открыть файл",[dlgfn_MultiSel], "", _OutListFiles ),
%Стандартное диалоговое окно открытия файла
FileName<>"",!, %Если файл выбран
open_file (FileName,b_true), %то открыть его
```

- открытый файл можно изменять и сохранять измененный файл под тем же именем

```
EW = win_GetFocus(), % Получить указатель окна редактора
edit_window (EW,FileName,OldText), %В этом окне в файле получить старый текст
Text = edit_GetText(EW), %Из окна с указателем EW вернуть в переменную Text содержание
OldText <> Text, % Если старый текст не соответствует новому
trap(file_str (FileName,Text),C,error_handler("Failed writing",C)), %Обработать возможные ошибки
retractall(edit_window(EW,_,_)), % Очистить факты из базы данных
assert(edit_window(EW,FileName,Text)), % В базу данных поместить новые факты
```

- сохранить под новым именем

```
NewName = dlg_GetFileName("*.lst", ["All files", "*. *"], "Save file",[dlgfn_Save], "", _OutListFiles ),
%Задать новое имя
EW = win_GetFocus(),%Получить указатель окна редактора
edit_window(EW,OldName,_) , %В этом окне получить старое имя
OldName <> NewName,!, %Если старое имя не соответствует новому,
save_as(EW,NewName) % изменить имя
; %иначе
EW = win_GetFocus(), % Получить указатель окна редактора
edit_window(EW,Name,OldText), %Заполнить факт из базы данных
Text = edit_GetText(EW), % Из окна с указателем EW вернуть в переменную Text содержание
OldText <> Text,!, %Если были изменения
trap(file_str(Name,Text),C,error_handler("Ошибка записи",C)), %Обработать ошибки
retractall(edit_window(EW,_,_)), %Очистить факты из базы данных
assert(edit_window(EW,Name,Text)) %Связать новые факты
```

- при выходе из редактора, если были изменения в файле, то система запросит о сохранении, если не было, то закроет файл.

Помимо всего прочего редактор предоставляет пользователю сервисные услуги по изменению текста: отмена последнего действия

```
edit_window(_ Window,_,_) , %В окне редактора
edit_PossibleUndoRedo(_ Window,Undo, _Redo), %Если отмена возможна
Undo=b_true,!,
edit_Undo(_ Window), %Произвести отмену последнего действия
```

повтор отмененного действия

```
edit_window(_ Window,_,_) , %В окне редактора
edit_PossibleUndoRedo(_ Window, _Undo,Redo), %Если повтор возможен
Redo=b_true,!,
edit_Redo(_ Window), %Произвести повтор отмененного действия
```

работа с буфером, вырезать в буфер обмена выделенный текст

```
edit_window(_ Window,_,_) , %В окне редактора
edit_Cut(_ Window), %Выделенный текст вырезать в буфер
```

копировать выделенный текст в буфер обмена

```
edit_window(_ Window,_,_) , %В окне редактора
edit_Copy(_ Window), %Копировать текст в буфер
```

вставить текст из буфера

```
edit_window(_ Window,_,_) , %В окне редактора
edit_Paste(_ Window), %Вставить содержимое буфера
```

удалить выделенный текст

```
edit_window(_ Window,_,_) , %В окне редактора
edit_Delete(_ Window), %Удалить выделенный текст
```

Обеспечение этих действий возможно благодаря применению как стандартных предикатов, так и определенных в программе. Таким образом, в перспективе рассмотренный прототип может послужить базой для создания

медицинских экспертных систем, предназначенных для предварительной постановки диагноза на основе симптоматики; оценки динамики состояния пациентов и дистанционных консультаций. При этом набор симптомов должен быть выбран таким образом, чтобы обеспечивалась достаточная дифференцирующая способность в отношении диагностируемых заболеваний.

#### Выводы

- Предложены инструментальные средства проектирования экспертных систем на основе семантических сетей, позволяющие создавать по определенным правилам базы знаний, используемые в данной экспертной системе, редактировать и пополнять уже существующие, а также проводить консультации.
- Можно просмотреть дерево заболеваний, а также местоположение определенной болезни в нем. Разработанная система является продукционной. Системы продукции включают три основных компонента: базу правил, состоящую из набора продукции (правил вывода), базу данных и интерпретатор для получения логического вывода.

#### Conclusions

- Proposed tools design expert system based on semantic networks, allowing to create specific rules for the knowledge base used in this expert system, edit and add to existing ones, as well as consulted.
- diseases can browse the tree, as well as the location of a particular disease therein. The developed system is a production. Production systems consist of three main components: the rule base, consisting of a set of productions (rules of inference), a database, and an interpreter for inference.
- Telemedicine consultative and educational center. - Mode of access: <http://bonum.info/telemedicine-center> (date accessed 11/06/2014), free. - Title from the screen. - Jaz. rus.

#### Список литературы

1. Батырканов Ж.И. Обучающие экспертные системы// Известия КГТУ. - Бишкек – 2011.-№21. – С.129-131.
2. Батырканов Ж.И. Модели представлений знаний на основе приближенного множества// Вестник науки Костанайского социально – технического университета им. академика Зулхарнай Алдамжар 3/2014-с.35-39.
3. Боскебеев К.Дж. Модель интеллектуальной обучающей системы на основе теории систем // Материалы международной научно-практической конференции, посвященной 70 – летию образования Волгоградского государственного аграрного университета. Волгоград, 2014г. – с.331-336.
4. Боскебеев К.Дж. Информационная система управления на основе метода нечеткого множества // "Вестник Таджикского технического университета" – январь - март 2014г. – с.45-49.
5. Комплексная медицинская информационная система (КМИС). – Режим доступа:[http://vs.ramec.ru/computer/ram\\_med/kmis&printable](http://vs.ramec.ru/computer/ram_med/kmis&printable) (дата обращения 06.11.2014), свободный. – Заглавие с экрана. – Яз. рус.
6. Митюшкин Ю. И. Soft-Computing: Идентификация закономерностей нечеткими базами знаний / Ю. И. Митюшкин, Б. И. Мокин, А. П. Ротштейн. – Винница: УНІВЕРСУМ-Вінниця, 2002. – 145 с.
7. Нестерова Е. В. Проблемы извлечения знаний из баз данных / Е. В. Нестерова, И. Г. Жукова // VII Регион. конф. молодых исследователей Волгоградской области. Волгоград, 12–15 ноября 2002 / ВолгГТУ [и др.] – Волгоград, 2003. – С. 208–210.
8. Телемедицинский консультативно-образовательный центр. – Режим доступа: <http://bonum.info/telemedicine-center> (дата обращения 06.11.2014), свободный. – Заглавие с экрана. – Яз. рус.

#### References

1. Batyrkanov Z.I. Educational expert systems // Proceedings KGTU. Bishkek. - 2011.-№ 21. - page.129-131.
2. Batyrkanov Z.I. Model of knowledge representation based on rough sets // Bulletin of Science Kostanai Social - Technical University. Academician Zulkharnai Aldamzhar 3/2014, page.35-39.
3. Boskebeev K.D. Intelligent tutoring system model based on the theory of systems // Proceedings of the international scientific-practical conference dedicated to the 70 - anniversary of Volgograd State Agrarian University. January 28 - January 28. Volgograd 2014. - page.331-336.
4. Boskebeev K.D. Management information system based on the method of fuzzy sets // "Bulletin of the Tajik Technical University" 1 (25) January-March 2014. - page.45-49.
5. Integrated Medical Information System (CMIS). - Mode of access: [http://vs.ramec.ru/computer/ram\\_med/kmis&printable](http://vs.ramec.ru/computer/ram_med/kmis&printable) (date accessed 06/11/2014), free. - Title from the screen. - Jaz. rus.
6. Mityushkin Y.I. Soft-Computing: Identifying patterns of fuzzy knowledge bases / YI Mityushkin, BI Mokin, AP Rothstein. - Athens: UNIVERSUM-Vinnitsya, 2002. - 145 p.
7. Nesterov E.V. The problem of extracting knowledge from databases / EV Nesterov, And. Zhukov // VII Region. Conf. young researchers of the Volgograd region, Volgograd, 12-15 November 2002 / VSTU [et al.] - Volgograd, 2003. - page. 208-210.
8. Telemedicine consultative and educational center. - Mode of access: <http://bonum.info/telemedicine-center> (date accessed 11/06/2014), free. - Title from the screen. - Jaz. rus.