Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Пермская государственная сельскохозяйственная академия

имени академика. Д.Н. Прянишникова»

Кафедра информационных систем

Козлов А.Н.

# **Интеллектуальные информационные системы**

МЕТОДИЧЕСКОЕ ПОСОБИЕ ПО курсовому проектированию

специальность 080801 «Прикладная информатика (в экономике)»

направления 080800 и 230700 «Прикладная информатика»

**2012**

УДК 004.491

ББК 32.973.018.2

# **Козлов, А.Н.** Интеллектуальные информационные системы. Методическое пособие по курсовому проектированию – Пермь: ФГБОУ ВПО Пермская ГСХА, 2012. – 25 с.

*Рецензент:*

**Беляков А.Ю.**, канд. техн. наук, доцент, доцент кафедры Информационных систем ФГБОУ ВПО Пермская ГСХА.

Учебное пособие содержит методические указания по выполнению курсового проекта в соответствии с программой дисциплины «Интеллектуальные информационные системы» для студентов, обучающихся по специальности 080801 «Прикладная информатика (в экономике)», а также студентов обучающихся по направлениям 080800 и 230700 «Прикладная информатика»».

Предназначено для студентов очной и заочной форм обучения.

УДК 004.491

ББК 32.973.0.18.2

© Козлов А.Н. 2012

Содержание

|  |  |
| --- | --- |
| Введение | 4 |
| 1. Структура курсового проекта | 5 |
| 2. Учебно-методический материал для выполнения практической части курсового проекта | 9 |
| 2.1. Постановка задачи. | 9 |
| 2.2. Разработка дерева решений | 10 |
| 2.3. **Преобразование дерева решений в правила** | 13 |
| 2.4. Структуры данных экспертной системы | 16 |
| **3. Рекомендации по выполнению и защита курсового проекта.** | 21 |
| Заключение | 23 |
| Библиографический список | 24 |
| **Приложение** | 25 |

Введение

Курсовое проектирование является необходимым и важным компонентом образовательной дисциплины, в ходе проектирования студенты не только систематизируют полученные знания, но и приобретают конкретные навыки и умения Это позволяет им подготовиться к более профессиональному подходу при выполнении предстоящих учебных и производственных задач.

Курсовое проектирование позволяет сформировать у студента следующие компетенции:

способность находить организационно-управленческие решения и готовность нести за них ответственность (ОК-4);

способность использовать нормативные правовые документы в профессиональной деятельности (ПК-1);

способность ставить и решать прикладные задачи с использованием современных информационно-коммуникационных технологий (ПК-4);

Данное учебное пособие содержит методические указания по выполнению курсового проекта в соответствии с программой дисциплины «Интеллектуальные информационные системы» для студентов, обучающихся по специальности 080801 «Прикладная информатика (в экономике)», а также студентов обучающихся по направлениям 080800 и 230700 «Прикладная информатика»».

Предназначено для студентов очной и заочной форм обучения.

1. Структура курсового проекта

Курсовой проект состоит из двух частей: теоретической и практической. Обе части оформляются в одной обложке

Теоретическая часть выполняется в виде реферата объемом 10-15 листов (в обложке) на листах бумаги формата А4. Текст должен быть выполнен с помощью компьютера. Страницы работы должны быть пронумерованы, поля сверху и снизу 2 см, справа 1,5 см слева 3 см, шрифт – 14 Times New Roman, межстрочный интервал - 1,2. В конце реферата приводится список использованной литературы, ставится подпись и дата выполнения.

Практическая часть выполняется на отдельных листах, которые подшиваются к реферату. Нумерация листов общая с рефератом. Программная реализация записывается на CD, который необходимо заклеить в конверт и подшить к реферату.

Структура курсового проекта:

Титульный лист (Приложение 1).

Содержание.

**Часть 1**. Реферат на тему « ...(*согласно варианта*) ... ».

Введение.

* 1. 1. ... *Основная часть реферата*
  2. 2. ... *(2-3 раздела с подразделами).*
  3. 3. ...

Заключение.

Список литературы. (*5-10 наименований)*

**Часть 2**. Разработка экспертной системы «...*название*...».

1. Постановка задачи.

2. Дерево решений, таблица переменных и база знаний.

3. Структуры данных.

4. Алгоритм и описание программной реализации.

Приложение: программная реализация экспертной системы.

Первая часть. Подготовить реферат по заданной теме. Номера тем определяются по табл.1 по предпоследней и последней цифрам номера зачетной книжки студента (шифра). Например, студент, у которого номер зачетной книжки оканчивается на 58, должен выбрать тему 9.

##### *Таблица 1*

##### Номера тем для выполнения первой части задания

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Предпоследняя цифра шифра | Последняя цифра шифра | | | | | | | | | |
| **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| **0** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **1** | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| **2** | 21 | 22 | 23 | 24 | 25 | 1 | 2 | 3 | 4 | 5 |
| **3** | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| **4** | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| **5** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| **6** | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| **7** | 21 | 22 | 23 | 24 | 25 | 1 | 2 | 3 | 4 | 5 |
| **8** | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| **9** | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

## Темы для выполнения первой части курсового проекта.

1. Интеллектуальные информационные системы: назначение, предметные области применения, особенности, классификация, примеры.
2. Система естественно- языкового интерфейса в ИИС.
3. Интеллектуальный интерфейс.
4. Методы рассуждения в ИИС
5. Этапы создания ИИС.
6. Понятие и структура фрейма.
7. Инструментальные средства создания ИИС.
8. Язык ПРОЛОГ: назначение, представление правил и фактов.
9. Экспертные системы: назначение и архитектура.
10. Машина вывода экспертной системы: назначение, структура, цикл работы.
11. Коллектив разработчиков ЭС.
12. Этапы проектирования экспертных систем.
13. Стратегии вывода: прямой вывод.
14. Стратегии вывода: обратный вывод.
15. Стратегии получения знаний.
16. Теоретические аспекты получения знаний: психологический.
17. Теоретические аспекты получения знаний: лингвистический.
18. Теоретические аспекты получения знаний: гносеологический.
19. Классификация методов извлечения знаний.
20. Коммуникативные методы извлечения знаний.
21. Текстологические методы извлечения знаний.
22. Машинное обучение: виды и их характеристика.
23. Понятие Интеллектуального анализа знаний (ИАД).
24. Нейронные сети.
25. Машинное обучение.

**Вторая часть (практическая).** Отработать порядок разработки экспертной системы для решения самостоятельно выбранной задачи (проблемы). Осуществить программную реализацию экспертной системы на любом языке программирования.

Результаты по второй части курсового проекта:

1. Сформулированная проблема (задача).
2. Дерево решений для выбранной проблемы (задачи).
3. Таблица переменных.
4. База знаний (правила 10-15 шт.).
5. Таблицы структур данных.
6. Блок схема алгоритма программной реализации.
7. Программная реализация ЭС.

Примерные темы курсового проекта:

1. Разработка экспертной системы «Выбор сотового телефона».
2. Разработка экспертной системы «Выбор квартиры».
3. Разработка экспертной системы «Выбор игрушек для девочек».
4. Разработка экспертной системы «Выбор персонального компьютера».
5. Разработка экспертной системы «Выбор домашнего животного».
6. Разработка экспертной системы «Выбор ноутбука».
7. Разработка экспертной системы «Выбор свадебного платья».
8. Разработка экспертной системы «Выбор оружия самозащиты».
9. Разработка экспертной системы «Выбор места отдыха».
10. Разработка экспертной системы «Место отдыха».
11. Разработка экспертной системы «Выбор специальности».
12. Разработка экспертной системы «Выбор принтера».
13. Разработка экспертной системы «Выбор мотоцикла».
14. Разработка экспертной системы «Выбор прически».
15. Разработка экспертной системы «Выбор компьютерной техники».
16. Разработка экспертной системы «Выбор страны отдыха».
17. Разработка экспертной системы «Выбор спортивной секции для ребенка».
18. Разработка экспертной системы «Выбор КПК».
19. Разработка экспертной системы «Выбор Меню».
20. Разработка экспертной системы «Выбор автомобиля».
21. Разработка экспертной системы «Выбор антивирусной программы».
22. Разработка экспертной системы «Выбор шампуня для волос».
23. Разработка экспертной системы «Выбор сабвуфера».
24. Разработка экспертной системы «Выбор квартиры».
25. Разработка экспертной системы «Выбор вида отдыха».
26. Разработка экспертной системы «формирование кадрового резерва».
27. Разработка экспертной системы «Брачное агентство».

**2. Учебно-методический материал для выполнения практической части курсового проекта**

**2.1. Постановка задачи.**

Прежде всего, поставим задачу, для решения которой будет разрабатываться экспертная система. Подходящей задачей, при решении которой можно использовать обратную цепочку рассуждений, может быть задача, вытекающая из следующей ситуации: к директору крупной технической фирмы пришел человек, желающий устроиться на работу. Директор располагает сведениями о потребностях фирмы в специалистах и общем положении дел в фирме. Ему нужно решить, какую должность в фирме может занять посетитель. Для этого необходимо задать посетителю такие вопросы, ответы на которые дадут возможность сделать правильный выбор должности.

На первый взгляд задача не очень сложная, но на решение директора влияет много факторов. Допустим, претендент работает в данной области недавно, но уже сделал важное открытие или он закончил учебное заведение с посредственными оценками, но несколько лет работал по специальности. В данной ситуации люди ведут себя по-разному, и хотя для того, чтобы получить работу необходимо, удовлетворять определенным критериям, в биографии претендента могут быть самые различные факты, анализ которых поможет подобрать для него соответствующую должность. Поскольку в задаче надо выбрать один из нескольких возможных вариантов (должностей), для её решения можно воспользоваться обратной цепочкой рассуждений.

Таким образом, необходимо разработать экспертную систему, которая определит подходящую должность посетителю. Экспертная система будет содержать экспертные знания директора и заменит его при решении описанной задачи. Такой системой может пользоваться как сам посетитель, так и менее квалифицированный (не эксперт - директор) сотрудник фирмы.

Итак, задача поставлена. Теперь нужно наглядно ее представить. Для описания подобных задач обычно используются диаграммы, которые называются деревьями решений. Деревья решений дают необходимую наглядность и позволяют проследить ход рассуждений.

**2.2. Разработка дерева решений.**

***Дерево решений*** *– это ориентированный граф, вершинами которого являются условия и выводы, а дугами результат выполнения (проверки) условий.*

Диаграммы называются деревьями решений потому, что, подобно настоящему дереву, имеют ветви. Ветви деревьев решений заканчиваются логическими выводами. Для рассматриваемого примера вывод заключается в том, предложит ли директор должность поступающему на работу, и если да, то какую. Многие задачи сложны, и их непросто представить (или для их решения не собираются использовать экспертную систему). Дерево решений помогает преодолеть эти трудности.

На рис.1 показано дерево решений для рассматриваемого примера приема на работу. Видно, что диаграмма состоит из кружков и прямоугольников, которые называются вершинами. Каждой вершине присваивается номер. На вершины можно ссылаться по этим номерам. Номера вершин можно выбрать произвольно, т.к они и служат только для удобства идентификации, за исключением первой вершины. Линии, соединяющие вершины, называются дугами. Совокупность вершин и дуг называется ветвями.

Кружки, содержащие вопросы, называются вершинами условий. Прямоугольники содержат логические выводы. Линии (стрелки) показывают направление диаграммы. Подписи возле линий это ответы на вопрос, содержащийся в вершине условия. Вершины условий могут иметь сразу по нескольку выходящих линий (стрелок), связывающих их с другими вершинами. В этом случае каждая линия (стрелка) должна быть четко определена. Не может быть две линии, у которых подписи одинаковые, например, подпись «Да». Выбор выходящей из вершины ветви определяется проверкой условия (вопроса), содержащегося в вершине. В программе под каждую вершину отводится переменная, а затем ей присваивается значение (ответ посетителя). Можно сказать, что вершины содержат переменные, а пути - это условия, в соответствии с которыми переменным присваиваются значения.

В дереве решений могут быть локальные (частные) выводы или цели. Для рассматриваемого примера локальным выводом может быть содержащийся в прямоугольнике 3 ответ на вопрос, будет ли посетителю предложена должность. Однако эта вершина имеет и исходящие ветви, и, следовательно, через неё может проходить путь к следующему логическому выводу. В последнем случае, поскольку исходящая ветвь не содержит условия и она только одна, говорят, что вершина содержит локальный вывод для другой цели. Локальный вывод - это также составляющая условной части (ЕСЛИ) правила.

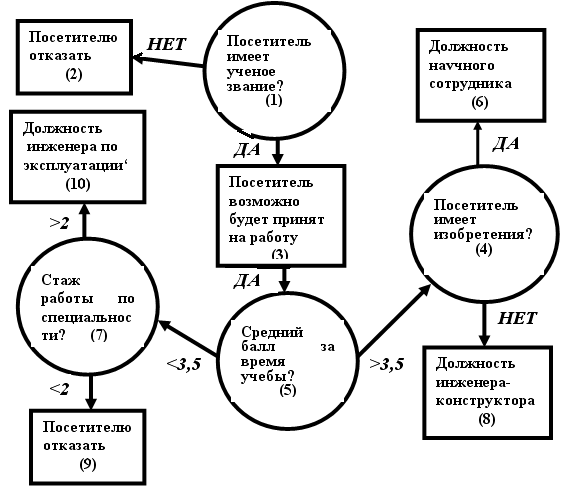


Рис.1. Дерево решений для выбора должности

**2.3**. **Преобразование дерева решений в правила**

Как уже говорилось, правило «ЕСЛИ-ТО» состоит из двух частей. Часть ЕСЛИ может включать несколько условий, которые связываются между собой логическими операторами И, ИЛИ и НЕ. Часть ТО правила включается в работу только в том случае, если истинны все условия в условной части. В дереве решений обеим частям правила соответствуют связанные между собой вершина(ы) логического условия(ий) (кружки) и вершина логического вывода (прямоугольник). Условная часть содержит все вершины условия, находящиеся на пути к логическому выводу, т.е. каждая вершина решения на пути к выводу - это одно условие части ЕСЛИ, например, вершины 1 и 4. Вывод же составляет часть ТО правила, в данном примере вершины 6, 8 и т.д.

Порядок формирования правил:

1. Выбрать из дерева решений вершину вывода (прямоугольник) и зафиксировать её.
2. В обратном направлении линии (стрелки) найти вершину условия (кружок) и зафиксировать её.
3. Повторять шаг 2 до тех пор, пока не будут исчерпаны все вершины условия, расположенные в обратном направлении стрелок от зафиксированной вершины вывода, или не встретится вершина локального вывода. Если встретилась вершина локального вывода, то её надо зафиксировать и прекратить выполнение шага 2.
4. Каждая вершина условия (кружок), составляющая путь, - это одна из переменных части ЕСЛИ правила. Эти вершины объединяются логическим оператором **И**.
5. Выбранный на шаге 1 логический вывод перенести в часть ТО правила.

**Пример создание правила.** В качестве примера рассмотрим путь 6, 4, 5, 3. Создание правила начинается с вывода (вершина 6) и дерево решения просматривается в обратную сторону. Просмотр данной ветви (пути) заканчивается на вершине 3, которая является локальным выводом. Если бы вершины 3 не было в дереве решений, то путь закончился бы на вершине 1.

Применив полученный путь, запишем правило в следующем виде:

**ЕСЛИ** посетитель, возможно, будет принят на работу = да

**И** средний балл за время учебы >=3,5

**И** посетитель имеет изобретения = да,

**ТО** предложенная должность = научный сотрудник.

Для каждой вершины логического вывода определяется путь и записывается правило. В рассматриваемом дереве решений имеется 6 вершин логического вывода (прямоугольников), в том числе и локальный вывод в вершине 3, значит будет составлено 6 правил.

Длинную фразу “посетитель, возможно, будет принят на работу ” можно заменить переменной, принимающей значения “да” или “нет”. Список имен переменных, текст, который они заменяют, и номера вершин пути сводят в таблицу, (табл.1). Использование переменных вместо полного текста упрощает формирование и запись правил.

*Таблица 6.1*

**Таблица имён переменных**

|  |  |  |
| --- | --- | --- |
| **Имя переменной** | **Условия** | **Вершина(ы)** |
| DEGREE | Посетитель имеет ученое звание? | 1 |
| QUALIFY | Посетитель, возможно, будет принят на работу | 3 |
| PATENT | Посетитель имеет изобретения? | 4 |
| EXPERIENCE | Стаж работы по специальности? | 7 |
| GRADE | Средний балл за время учебы? | 5 |
| POSITION | Предложенная должность | 2,6,8,9,10 |

Используя имена переменных из табл.1, вышеприведенное правило можно запишется в следующем виде:

**ЕСЛИ** QUALIFY = да **И** GRADE >=3,5 **И** PATENT = да,

**ТО** POSITION = научный сотрудник

В табл.2 приведены все правила для дерева решений, показанного на рис.1. Правила соответствуют всем шести путям, ведущим к шести возможным целям дерева решений. Правила желательно пронумеровать. Совокупность правил является формализованными знаниями (в рассматриваемом примере – знаниями руководителя) и представляет собой *базу знаний*.

*Таблица 2*

**База знаний**

|  |  |  |
| --- | --- | --- |
| № | Правило | Путь |
| 10 | **ЕСЛИ** DEGREE = *НЕТ*, **ТО** POSITION=*ОТКАЗАТЬ* | 2, 1 |
| 20 | **ЕСЛИ** DEGREE = *ДА*, **ТО** QUALIFY=*ДА* | 3, 1 |
| 30 | **ЕСЛИ** QUALIFY = *да* **И** GRADE >=3,5 **И** PATENT = *да*,  **ТО** POSITION = *научный сотрудник* | 6,4,5.3 |
| 40 | **ЕСЛИ** QUALIFY = *да* **И** GRADE >=3,5 **И** PATENT = *НЕТ*,  **ТО** POSITION = *ИНЖЕНЕР КОНСТРУКТОР* | 8,4,5,3 |
| 50 | **ЕСЛИ** QUALIFY = *ДА* **И** GRADE <= *3,5* **И** EXPERIENCE < *2*, **ТО** POSITION=*ОТКАЗАТЬ* | 9,7,5,3 |
| 60 | **ЕСЛИ** QUALIFY = *ДА* **И** GRADE <= *3,5* **И** EXPERIENCE > *2*, **ТО** POSITION = *ИНЖЕНЕРПО ЭКСПЛУАТАЦИИ* | 10,7,5, 3 |

Таким образом, дерево решений позволяет просто и наглядно представить ход рассуждений эксперта при решении задачи и формировать правила для базы знаний, а без базы знаний экспертную систему не построить. Аналогично можно построить базу знаний для своей проблемной области или решаемой задаче.

**2.4. Структуры данных экспертной системы.**

При создании экспертной системы для упрощения ответа на вопросы и решения поставленной задачи в систему включается ряд полезных таблиц или структур данных. Структуры данных нужны для работы с базой знаний. После определения метода решения выбранного круга задач можно приступить к разработке системы.

***Список логических выводов*** *- это структура данных, содержащая упорядоченный список возможных логических выводов.*

Список состоит из номера правила, логического вывода, связанного с этим правилом, и условий, которые формируют вывод. На каждое правило базы знаний в списке приходится одна запись. Создание записи списка поясним на примере правила 10. Часть ТО правила 10 содержит переменную POSITION, т.е. переменная POSITION связана с логическим выводом правила 10.

Список логических выводов используется исключительно для поиска вывода по номеру правила. Когда условия части ЕСЛИ истинны, вызывается часть ТО правила, ей присваивается значение. Например, если надо узнать, будет ли посетителю предложена работа, в списке ищется переменная POSITION. Она содержится в первой же записи, т.е. в правиле 10:

**ЕСЛИ** DEGREE=НЕТ,

**ТО** POSITION=НЕТ

Посетитель не будет принят на работу, если переменная DEGREE имеет значение НЕТ. Если же переменная DEGREE имеет значение ДА, тогда обращаться к части ТО правила нельзя, поскольку не выполняется условие части ЕСЛИ (DEGREE=ДА). Поэтому надо продолжить поиск правила, содержащего в части ТО переменную POSITION (в данном случае правило 30).

На рис.2 приведён полностью сформированный список логических выводов для всех правил базы знаний.

**10 POSITION  
20 QUALIFY  
30 POSITION  
40 POSITION  
50 POSITION  
60 POSITION**

Рис. 2. Список логических выводов.

Список считается сформированным, когда логический вывод каждого правила помещён в запись с номером, совпадающим с номером правила.

***Список переменных – это перечень*** *имен переменных для всех условных частей правил базы знаний и признак их инициализации.*

Признак инициализации показывает, присвоено ли переменной значение. Независимо от того, в скольких условиях встречается переменная, в список переменных она включается всего один раз. В этот список также нельзя включать переменные из списка логических выводов, поскольку их значения определяются с помощью правил. Например, правило 20 использует переменную QUALIFY. Список переменных приведён в таблице 3.

*Таблица 3*

**Список имен переменных**

|  |  |  |
| --- | --- | --- |
| **Имя** | **Признак** | **Значение** |
| DEGREE | I | НЕТ |
| PATENT | NI |  |
| EXPEREIENCE | NI |  |
| GRADE | NI |  |

Первоначально предполагается, что переменным значения еще не присвоены и признак инициализации для всех переменных равен NI. По мере того как полученная от посетителя информация передается системе и переменным присваиваются значения, признак инициализации меняется на I. С этого момента, в каком бы правиле в условной части не встретилась переменная, она будет считаться проинициализированной, имеющей какое либо значение и ее можно использовать для работы с любыми правилами.

Таким образом, до того, как правило включается в работу, все переменные, входящие в его условную часть, должны быть проинициализированы.

***Список переменных условия – это перечень всех переменных для всех условных частей всех правил базы знаний.***

Условная часть правила (ЕСЛИ) может содержать несколько переменных. Под каждое правило выделяется одинаковое число позиций в списке переменных условия. Минимальное число позиций равно числу переменных условия самого «длинного» правила. Можно добавить еще одну-две позиции «про запас» на случай доработки базы правил.

На рис.3 показан список переменных условия для шести правил рассматриваемой базы знаний. Для простоты программирования предполагается, что каждое правило не может содержать больше четырех переменных условия (т.к. самое длинное правило, например №40, содержит три переменных условия). Четвертая позиция добавлена «про запас».

|  |
| --- |
| **1 DEGREE 2 3 4 5 DEGREE 6 7 8 9 QUALIFY 10 GRADE 11 PATENT 12 13 QUALIFY 14 GRADE 15 PATENT 16 17 QUALIFY 18 GRADE 19 EXPERIENCE 20  21 QUALIFY 22 GRADE**  **23 EXPERIENCE**  **24** |

Рис.3. Список переменных условия.

Слева от имен переменных даны числа (1-24), указывающие индекс элемента массива (по четыре на правило), в который помещается имя соответствующей переменной. Незанятые элементы массива, отведенные правилу, остаются пустыми. В принципе можно запрограммировать любое число переменных для каждого правила. Однако при отведении места под переменные условия лучше для каждого правила резервировать одинаковое число элементов массива. Это упростит вычисление индекса первого элемента, отведенного правилу в списке. Его можно вычислить с помощью простой формулы:

№ = 4\* (номер правила/10 — 1)+1

Например, переменные правила 50 будут размещаться, начиная с 17-го элемента массива: 4\* (50/10—1)+1=17. №=17.

Теперь посмотрим, каким образом три описанные структуры данных соотносятся с мыслительной деятельностью человека в процессе обратной цепочки рассуждений. Прежде всего человек просматривает все возможные пути, способные привести к решению задачи (список логических выводов). Затем он выделяет условия, составляющие эти пути (список переменных и список переменных условия). Такие структуры данных позволяют быстро обрабатывать информацию, не повторяя одни и те же шаги по нескольку раз, потому что значения переменных можно использовать в определенной ситуации для различных логических выводов. Если же при разговоре с человеком, устраивающимся на работу, у директора нет не только компьютера, но даже карандаша и бумаги, ему придется много раз переспрашивать, ведь сразу просто невозможно запомнить. Конечно, в конце концов он примет решение, но затратит много сил и времени.

**3. Рекомендации по выполнению и защита курсового проекта.**

Рекомендации по выполнению второй части проекта:

* Количество правил в базе знаний: на оценку «отлично» - не менее 30, на «хорошо» - не менее 25, на «удовлетворительно» - не менее 20.
* Дерево решений составлять исходя из требуемого функционала, например, цель приобретения компьютера (телефона и т.д.), а не сумма денежных средств для этого.
* Вопросы в дереве решений могут повторяться не более двух раз.
* Рекомендуемая длина ветви дерева решений 3-6 вершин.
* Промежуточные вершины вывода делать не рекомендуется.
* Программная реализация экспертной системы возможна на любом языке программировании. Предпочтительно на объектно-ориентированных языках (Delphi, C#) или Пролог.
* Программная реализация экспертной системы должна иметь дружественный интерфейс, содержать картинки и иллюстрации, эстетично оформлена.
* Желательно отражать в окне программы вопросы из дерева решений и ответы на них, т.е. отобразить «цепочку логических выводов».

Постановка задачи, дерево решений и его описание, таблицы, структуры данных распечатываются на листах формата А4. Если дерево решений не вмещается, то можно склеить листы.

Программная реализация экспертной системы записыватся на CD-диск .

**ВНИМАНИЕ!** Программа должна работать в автономном режиме (exe – файл), имя файла должно начинаться с фамилии и содержать название дисциплины, например: **Васильев-ИИС.exe .**

Защита курсового проекта**.** На кафедре определяются сроки защиты проектов студентами. Защита проводится в виде доклада по теме реферата и разработанной экспертной системе. Также необходимо продемонстрировать работу экспертной системы на компьютере.

Для заочников. Курсовой проект должен быть сдан для регистрации в деканат факультета заочного обучения. **Перед началом сессии ОБЯЗАТЕЛЬНО узнать допущен или нет курсовой проект к защите** (позвонить по телефону на кафедру)**.**

**Заключение**

Разработанная экспертная система относится к классу поверхностных демонстрационных (учебных) систем. Поверхностные ЭС представляют знания в виде правил (условие – действие). Условие правила определяет образец некоторой ситуации, при соблюдении которой правило может быть выполнено. Поиск решения состоит в выполнении тех правил, образцы которых сопоставляются с текущими данными.

Создание экспертной системы в рамках данного курсового проекта позволяет изучить и реализовать все этапы разработки ЭС:

1. идентификация,
2. концептуализация,
3. формализация,
4. выполнение,
5. тестирование,
6. опытная эксплуатация.

Особенностью является то, что студент выполняет функционал всех членов коллектива разработчиков ЭС – эксперта, инженера по знаниям, программиста и пользователя.

Выполнение курсового проекта позволяет закрепить и расширить знания по теме экспертных систем в рамках дисциплины «Интеллектуальные информационные системы».

**Библиографический список**

1. Романов, В.П. Интеллектуальные информационные системы в экономике: Учебное пособие.-.: «Экзамен», 2003. – 496 с.
2. Гаврилова, Т.А., Хорошевский, С.В. Базы знаний интеллектуальных систем: учебное пособие. – СПб.: Питер,2006. -382 с.
3. Гаскаров, Д.В. Интеллектуальные информационные системы:Учебник для вузов. -М.: ВШ, 2005. – 432
4. Паклин, Н.Б., Орешков, В.И. Бизнес аналитика: от данных к знаниям: Учеб. пособие .2-е изд. – СПб.: Питер, 2010. – 704 с.: ил.
5. Программирование на языке Пролог для искусственного интеллекта: /Пер. с англ. И.Братко.-М.: Мир, 1990. - 560 с., ил.
6. Х. Уэно, М. Исидзука. Представление и использование знаний. М: Мир, 1989. -220 с.
7. 5. **Роберт Левин, Диана Дранг, Барри Эделсон. Практическое введение в технологию искусственного интеллекта и экспертных систем с иллюстрациями на бейсике.** Перевод с англ. М.Л.Сальникова, Ю.В.Сальниковой. М.: 2005 – 380с. (http://virtuald.narod.ru/ExpDob/Zastavka.htm)

*Приложение 1*

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

Пермская государственная сельскохозяйственная академия имени академика Д.Н.Прянишникова

Кафедра Информационных систем

###### Курсовой проект

по дисциплине:

*«***Интеллектуальные информационные системы***»*

на тему:

*«****Разработка экспертной системы «Выбор должности****»****.***

Название в соответствии со второй частью проекта

Выполнила:

студентка 4 курса

специальности: 080801 «Прикладная информатика в экономике»

шифр Пи-03-404

Суханова Любовь Сергеевна

Проверил:

направления

профессор

Козлов Алексей Николаевич

Пермь-20...