

**По следам книги
«Как построить свою экспертную систему»
(с) К. Нейлор**

**Краткое описание и исходники Delphi-компонета
TES_bayes (Байесовская ЭС)**

Оглавление

| | |
|---|----|
| Термины и сокращения | 3 |
| 1 Введение..... | 4 |
| 1.1 Цель этого документа | 4 |
| 1.2 Соглашения | 4 |
| 2 Delphi-компонент TES_bayes | 5 |
| 2.1 Основные свойства компонента..... | 5 |
| 2.1.1 fVersion/Version | 5 |
| 2.1.2 fYesInit/ComponentInitOK | 6 |
| 2.1.3 fTrainMode/TrainMode | 6 |
| 2.1.4 fInTraining/InTraining | 6 |
| 2.1.5 fTrainCycle/TrainCycle | 6 |
| 2.1.6 fTrainStep/TrainStep..... | 6 |
| 2.1.7 fTrainResult/TrainResult | 6 |
| 2.1.8 fObjTrainResult/ObjTrainResult | 6 |
| 2.1.9 fTrainProcessExit/TrainProcessExit | 6 |
| 2.1.10 fObjCount/ObjCount..... | 6 |
| 2.1.11 fPropCount/PropCount..... | 7 |
| 2.1.12 fObjTrain_J/ObjTrain_J..... | 7 |
| 2.1.13 fObjTrainCount/ ObjTrainCount | 7 |
| 2.1.14 fObjTrain_Rating/ObjTrain_Rating | 7 |
| 2.1.15 fObjBest_J/ObjBest_J..... | 7 |
| 2.1.16 fObjBest_Rating/ObjBest_Rating..... | 7 |
| 2.1.17 fTrainCycleMaxCount/ TrainCycleMaxCount..... | 7 |
| 2.1.18 fRatingRoundTo/RatingRoundTo..... | 7 |
| 2.2 Основные методы компонента..... | 8 |
| 2.2.1 Create | 9 |
| 2.2.2 Destroy | 9 |
| 2.2.3 Init | 10 |
| 2.2.4 Obj_Recognize..... | 11 |
| 2.2.5 ES_Train_Start..... | 14 |
| 2.2.6 ES_Train_Auto | 15 |
| 2.2.7 Obj_Train_Auto | 19 |
| 2.2.8 ES_Train_Manual_Next | 21 |
| 2.2.9 ES_Train_ExitForLimit | 22 |
| 2.2.10 ES_Train_Exit_UserAbort..... | 22 |
| 2.2.11 Obj_Add..... | 23 |
| 2.2.12 Prop_Add | 23 |

| | | |
|--------|---|----|
| 2.2.13 | ObjList_LoadFromTStrings..... | 23 |
| 2.2.14 | PropList_LoadFromTStrings | 24 |
| 2.2.15 | PropsValue_from_IntArray | 24 |
| 2.2.16 | PropsValue_from_TStrings | 25 |
| 2.2.17 | PropsValue_from_TCheckBox..... | 25 |
| 2.2.18 | PropsValue_Control..... | 25 |
| 2.2.19 | Obj_Rating_TStringGrid_Clear..... | 25 |
| 2.2.20 | Obj_Rating_to_TStringGrid | 26 |
| 2.2.21 | Obj_Rating_Percent_Recalc | 26 |
| 2.2.22 | Rules_TStringGrid_Clear | 26 |
| 2.2.23 | Rules_to_TStringGrid..... | 26 |
| 2.2.24 | Rules_From_TStringGrid | 27 |
| 2.3 | События компонента..... | 28 |
| 2.3.1 | fBeforeTrain/BeforeTrain | 28 |
| 2.3.2 | fAfterTrain/AfterTrain | 28 |
| 2.3.3 | fBeforeCycle/BeforeCycle..... | 28 |
| 2.3.4 | fAfterCycle/AfterCycle..... | 28 |
| 2.3.5 | fBeforeStep/BeforeStep | 28 |
| 2.3.6 | fAfterStep/AfterStep | 29 |
| 2.4 | Обучение ЭС | 30 |
| 2.4.1 | Обучение ЭС в автоматическом режиме | 30 |
| 2.4.2 | Обучение ЭС в ручном режиме..... | 31 |
| 3 | Использование компонента TES_bayes на примере простой программы..... | 33 |
| 3.1 | Основные операции..... | 33 |
| 3.1.1 | Формирование перечня задач | 34 |
| 3.1.2 | Обучение ЭС для выбранной задачи | 37 |
| 3.1.3 | Проверка качества обучения ЭС: распознавание объектов в рамках выбранной задачи | 44 |
| 4 | Используемые источники..... | 45 |

Термины и сокращения

В таблице ниже приведен перечень используемых в документе терминов и сокращений.

Таблица 1 – Перечень используемых в документе терминов и сокращений

| Сокращение | Обозначение |
|------------------|---|
| 1 | 2 |
| Delphi | Embarcadero Delphi 10.2 Tokyo |
| Delphi-компонент | Потомок класса TComponent (в Delphi) |
| ES_Bayes.exe | Простая программа, иллюстрирующая применение TES_bayes |
| TES_bayes | Delphi-компонент (Байесовская ЭС) |
| Документ | Статья «По следам книги «Как построить свою экспертную систему». (с) К. Нейлор» |
| Задача | Задача, в рамках которой определены конкретные объекты (предназначенные для распознавания) и их признаки (свойства) |
| Книга | Книга «Как построить свою экспертную систему». Автор К. Нейлор. Москва, ЭНЕРГОАТОМИЗДАТ, 1991 |
| Компонент | Delphi-компонент TES_bayes |
| ИИ | Искусственный интеллект |
| МНТ | Метод научного тыка |
| Пользователь | 1. Программист, использующий Delphi-компонент TES_bayes. 2. Пользователь программы ES_Bayes.exe |
| Признак | См. Свойство, пункт 2. Вид свойства: логическое |
| Свойство | 1. Свойство Компонента. 2. Свойство распознаваемого объекта |
| ЭС | Экспертная система |

1 Введение

К. Нейлор (в своей книге «Как построить свою экспертную систему») подробнейшим образом изложил как теоретические основы построения экспертных систем байесовского типа, так и их практическое применение для «домашних целей».

В книге приведены не только рассуждения и информационные послы, но и (что крайне важно) конкретные методы и алгоритмы (вплоть до исходных кодов) разработки экспертных систем.

Но... Исходные коды приведены в синтаксисе языка программирования Basic...
Что несколько затрудняет их использование в современных условиях.

Автор этого Документа озадачился вопросом создания собственной экспертной системы (байесовского типа) с целью разработки ряда простых игровых (с элементами обучения детей основам некоторых технологий ИИ) программ.

1.1 Цель этого документа

Цель этого документа: помочь (по возможности) тем, кого интересует соответствующая тематика (построение своих собственных ЭС и их сетей) для каких-либо своих нужд (например, обучающе-игровые программы для детей/внуков) на конкретном примере Delphi-компонента «Байесовская ЭС».

1.2 Соглашения

1. Автор не ставит перед собой цель – удивить этот мир «красотой» и оптимальностью исходного кода, а также применяемой методики.
2. Предполагается, что читатель знает Delphi и умеет программировать на достаточном, прикладном уровне.
3. Прилагаемые к этому документу исходники хотя и рабочие, но носят чисто иллюстративный характер.
4. При разработке компонента TES_bayes были использованы материалы из книги К. Нейлора «Как построить свою экспертную систему»

2 Delphi-компонент TES_bayes

2.1 Основные свойства компонента

В таблице 2 приведены основные свойства компонента.

Таблица 2 – Основные свойства компонента

| Свойство | Раздел | Назначение |
|--|-------------------|---|
| 1 | 2 | 3 |
| fVersion/Version | private/public | Версия компонента |
| fYesInit/ComponentInitOK | private/public | Флаг (признак): факт инициализации компонента |
| fTrainMode/TrainMode | private/public | Режим обучения ЭС (авто или ручной) |
| fInTraining/InTraining | private/public | Состояние ЭС. Если TRUE, то в режиме обучения |
| fTrainCycle/TrainCycle | private/public | Номер текущего цикла обучения |
| fTrainStep/TrainStep | private/public | Номер текущего шага в текущем цикле обучения |
| fTrainResult/TrainResult | private/public | Результат обучения ЭС |
| fObjTrainResult/ObjTrainResult | private/public | Результат распознавания объекта (на конкретном шаге) |
| fTrainProcessExit/TrainProcessExit | private/public | Причина окончания/прерывания процесса обучения ЭС |
| fObjCount/ObjCount | private/public | Кол-во распознаваемых объектов |
| fPropCount/PropCount | private/public | Кол-во признаков (свойств), используемых для распознавания объектов |
| fObjTrain_J/ObjTrain_J | private/public | Индекс объекта, используемого для обучения ЭС (на очередном шаге очередного цикла обучения) |
| fObjTrain_Rating/ObjTrain_Rating | private/public | Рейтинг объекта с индексом fObjTrain_J |
| fObjTrainCount/ObjTrainCount | private/public | Счетчик корректно распознанных объектов при обучении ЭС |
| fObjBest_J/ObjBest_J | private/public | Индекс (номер) объекта с наилучшим рейтингом |
| fObjBest_Rat/ObjBest_Rat | private/public | Значение рейтинга объекта с индексом fObjBest_J |
| fTrainCycleMaxCount/ TrainCycleMaxCount | private/published | Максимальное кол-во циклов обучения |
| fRatingRoundTo/RatingRoundTo | private/published | Кол-во цифр после запятой при округлении рейтингов объектов (%) |

2.1.1 fVersion/Version

Разделы: private/public.

Назначение: Версия компонента.

Тип: string.

2.1.2 fYesInit/ComponentInitOK

Разделы: private/public.

Назначение: Флаг (признак) – факт инициализации компонента.

Тип: boolean.

2.1.3 fTrainMode/TrainMode

Разделы: private/public.

Назначение: Режим обучения ЭС (авто или ручной).

Тип: TTrainMode.

2.1.4 fInTraining/InTraining

Разделы: private/public.

Назначение: Состояние ЭС. Если TRUE, то в режиме обучения.

Тип: boolean.

2.1.5 fTrainCycle/TrainCycle

Разделы: private/public.

Назначение: Номер текущего цикла обучения.

Тип: integer.

2.1.6 fTrainStep/TrainStep

Разделы: private/public.

Назначение: Номер текущего шага в текущем цикле обучения.

Тип: integer.

2.1.7 fTrainResult/TrainResult

Разделы: private/public.

Назначение: Результат обучения ЭС.

Тип: TTrainResult.

2.1.8 fObjTrainResult/ObjTrainResult

Разделы: private/public.

Назначение: Результат распознавания объекта (на конкретном шаге).

Тип: TObjTrainResult.

2.1.9 fTrainProcessExit/TrainProcessExit

Разделы: private/public.

Назначение: Причина окончания/прерывания процесса обучения ЭС.

Тип: TTrainProcessExit.

2.1.10 fObjCount/ObjCount

Разделы: private/public.

Назначение: Кол-во распознаваемых объектов.

Тип: integer.

2.1.11 fPropCount/PropCount

Разделы: private/public.

Назначение: Кол-во признаков (свойств), используемых для распознавания объектов.

Тип: integer.

2.1.12 fObjTrain_J/ObjTrain_J

Разделы: private/public.

Назначение: Индекс объекта, используемого для обучения ЭС (на очередном шаге очередного цикла обучения).

Тип: integer.

2.1.13 fObjTrainCount/ ObjTrainCount

Разделы: private/public.

Назначение: Счетчик корректно распознанных объектов при обучении ЭС.

Тип: integer.

2.1.14 fObjTrain_Rating/ObjTrain_Rating

Разделы: private/public.

Назначение: Рейтинг объекта с индексом fObjTrain_J.

Тип: integer.

2.1.15 fObjBest_J/ObjBest_J

Разделы: private/public.

Назначение: Индекс (номер) объекта с наилучшим рейтингом.

Тип: integer.

2.1.16 fObjBest_Rating/ObjBest_Rating

Разделы: private/public.

Назначение: Значение рейтинга объекта с индексом fObjBest_J.

Тип: integer.

2.1.17 fTrainCycleMaxCount/ TrainCycleMaxCount

Разделы: private/published.

Назначение: Максимальное кол-во циклов обучения.

Тип: integer.

2.1.18 fRatingRoundTo/RatingRoundTo

Разделы: private/published.

Назначение: Кол-во цифр после запятой при округлении рейтингов объектов (%).

Тип: integer.

2.2 Основные методы компонента

В таблице 3 приведены основные методы компонента.

Таблица 3 – Основные методы компонента

| Метод | Назначение |
|------------------------------|---|
| 1 | 2 |
| Create | Создать компонент (конструктор) |
| Destroy | Уничтожить компонент и освободить память, отведенную под него |
| Init | Инициализация компонента (обязательный шаг перед использованием компонента) |
| Obj_Recognize | Распознать объект (использование ЭС по назначению - для распознавания объектов) |
| ES_Train_Start | Старт процесса обучения ЭС |
| ES_Train_Auto | Обучение ЭС в автоматическом режиме |
| Obj_Train_Auto | Обучение ЭС для очередного объекта (очередной шаг обучения) в автоматическом или ручном режиме |
| ES_Train_Manual_Next | Обучение ЭС для очередного объекта (очередной шаг на очередном цикле) в ручном режиме |
| ES_Train_ExitForLimit | Досрочное завершение процесса обучения ЭС из-за превышения максимального кол-ва циклов обучения |
| ES_Train_Exit_UserAbort | Досрочное завершение процесса обучения ЭС по требованию Пользователя |
| Obj_Add | Добавить объект в перечень объектов |
| Prop_Add | Добавить признак (свойство) в перечень признаков (свойств) |
| ObjList_LoadFromTStrings | Загрузить список объектов из TStrings |
| PropList_LoadFromTStrings | Загрузить список признаков (свойств) из TStrings |
| PropsValue_from_IntArray | Импорт значений свойств из массива целых чисел (0–false, 1–true) |
| PropsValue_from_TStrings | Импорт значений свойств из TStrings ('0'–false, '1'–true) |
| PropsValue_from_TCheckBox | Импорт значений свойств из TCheckBox |
| PropsValue_Control | Контроль, что значение хотя бы один из признаков (свойств) = TRUE |
| Obj_Rating_TStringGrid_Clear | Создать пустую (из 2-х строк) таблицу Рейтингов объектов (TStringGrid) |
| Obj_Rating_to_TStringGrid | Выгрузить рейтинги объектов в таблицу Рейтингов (TStringGrid) |
| Obj_Rating_Percent_Recalc | Расчет нормированных значений рейтинга объектов и их же - в процентах |
| Rules_TStringGrid_Clear | Создать пустую (из 2-х строк) таблицу правил для распознавания объектов (TStringGrid) |
| Rules_to_TStringGrid | Экспорт таблицы правил из внутреннего представления в TStringGrid |
| Rules_From_TStringGrid | Импорт таблицы правил из TStringGrid |

2.2.1 Create

Назначение: Создать компонент (конструктор).

Наследуемый метод от TComponent.

```
constructor Create(AOwner : TComponent); override;
```

2.2.2 Destroy

Назначение: Уничтожить компонент и освободить память, отведенную под него.

Наследуемый метод от TComponent.

```
destructor Destroy; override;
```

2.2.3 Init

Назначение: Инициализация компонента (обязательный шаг перед использованием компонента).

```
function Init(ObjCount:integer; PropCount:integer):boolean;
```

Входные параметры:

ObjCount – кол-во распознаваемых объектов;

PropCount – кол-во признаков (свойств), используемых для распознавания объектов.

Возвращаемое значение: =TRUE, если успешно и =FALSE в противном случае.

На рисунке 1 приведена блок-схема, иллюстрирующая функционал метода Init.



Рисунок 1 – Метод Init

2.2.4 Obj_Recognize

Назначение: Распознать объект по заданным признакам (свойствам).

```
function Obj_Recognize(ListProp:TStrings;
ListFuzzi:TStrings=nil):integer; overload;
```

```
function Obj_Recognize(ListProp:TCheckListBox;
ListFuzzi:TStrings=nil):integer; overload;
```

Входные параметры:

ListProp – список значений признаков (свойств), на основании которого ЭС распознает объект;

ListFuzzi – список индексов распознанных объектов, имеющих одинаковый, максимальный рейтинг.

Возвращаемое значение: индекс объекта (начиная с нуля), который распознала ЭС на основании значений признаков (свойств). В случае ошибки: -1.

Формат ListProp типа TStrings:

1. Одна строка – одно значение.
2. Значения: 0 – FALSE, 1 – TRUE.

Пример:

```
0
0
1
0
1
```

На рисунке 2 приведена блок-схема, иллюстрирующая функционал метода **Obj_Recognize** с параметром ListProp:TStrings.

На рисунке 3 приведена блок-схема, иллюстрирующая функционал метода **Obj_Recognize** с параметром ListProp:TCheckListBox.

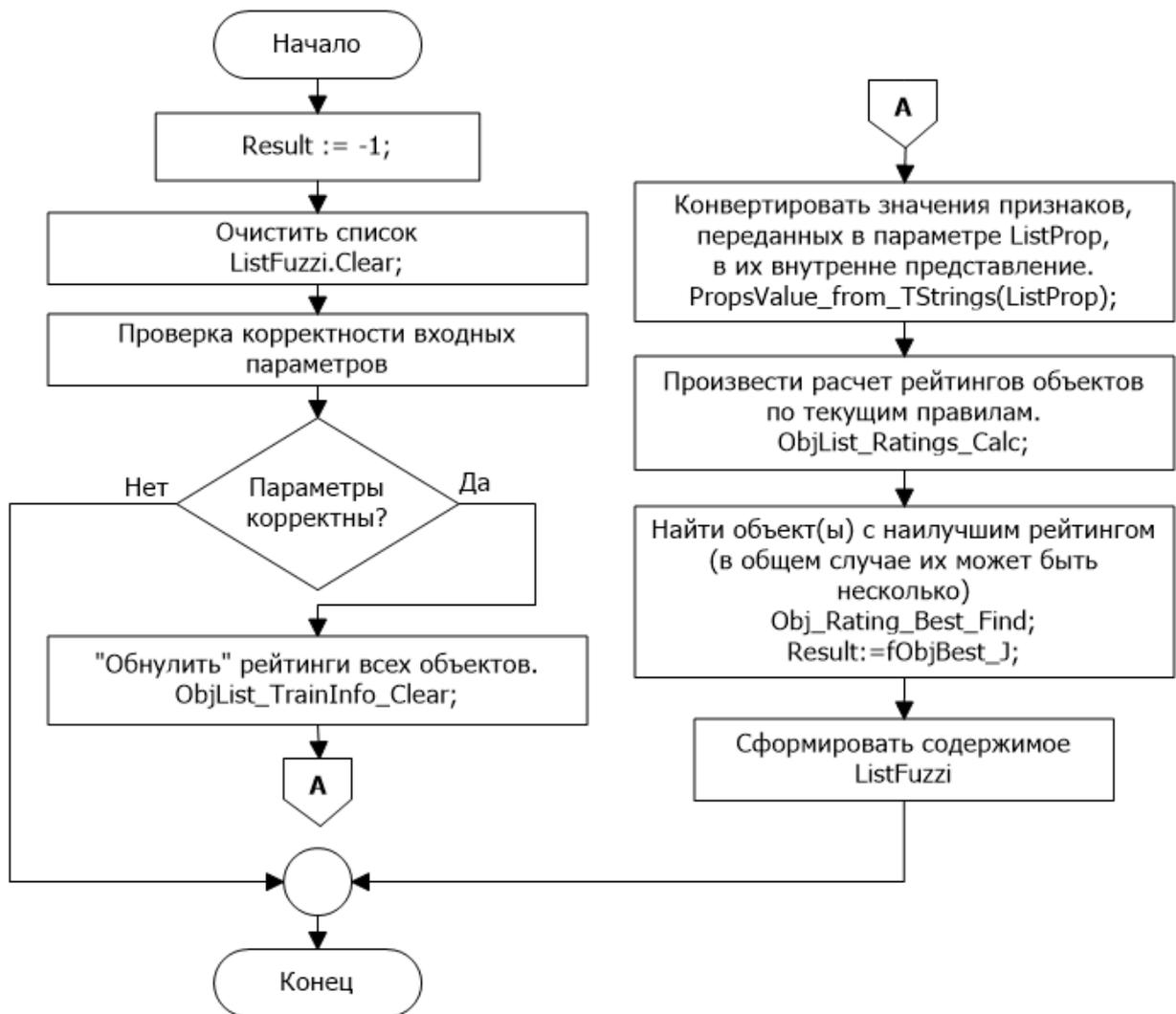


Рисунок 2 – Метод Obj_Recognize с параметром ListProp:TStrings

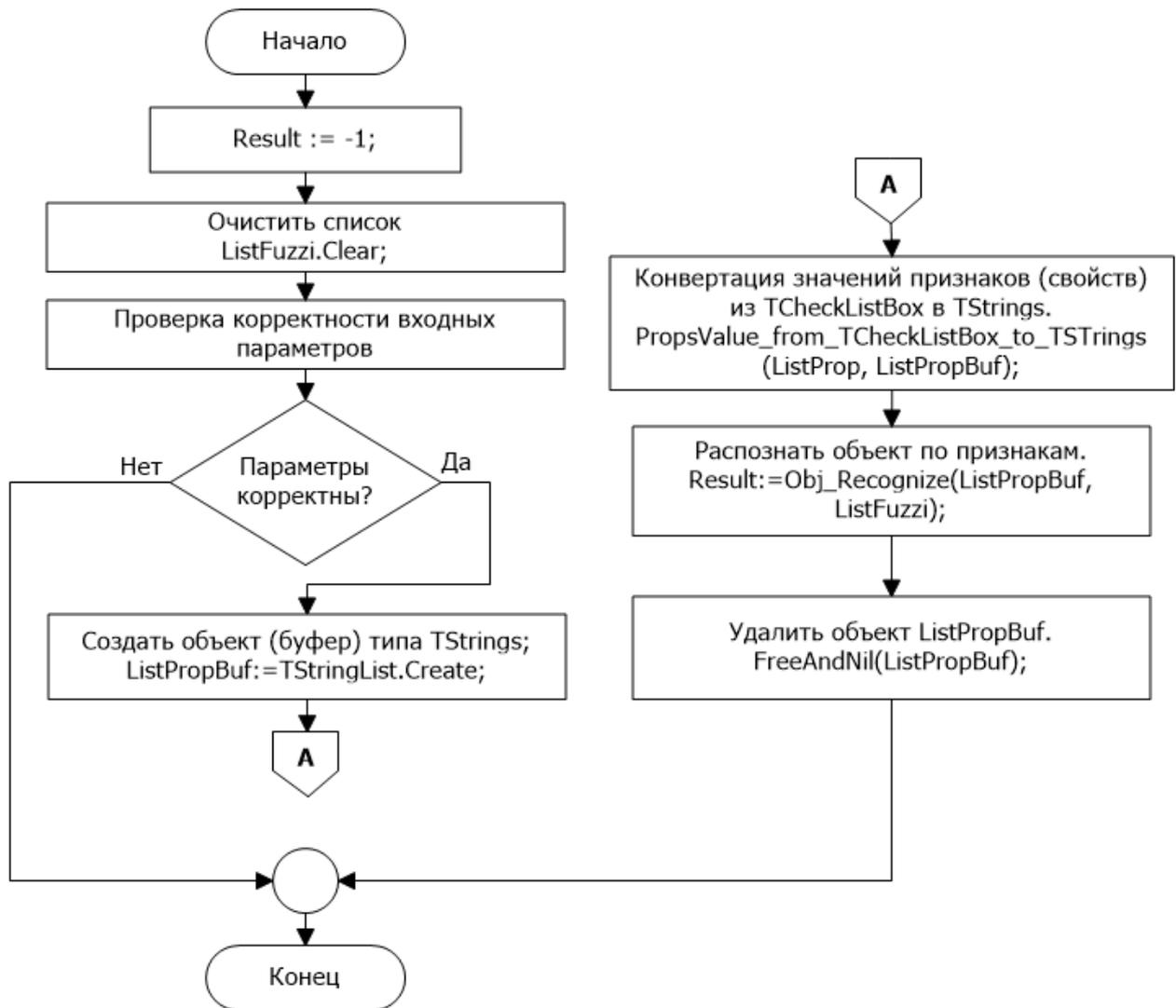


Рисунок 3 – Метод Obj_Recognize с параметром ListProp:TCheckBox

2.2.5 ES_Train_Start

Назначение: Старт процесса обучения ЭС (автоматический или ручной режим).

```
function ES_Train_Start(
    ES_TrainMode:TTrainMode;
    PropsListTrue_for_EachObj:TStrings
):boolean;
```

Входные параметры:

ES_TrainMode – режим обучения ЭС (ES_tmAuto или ES_tmManual);

PropsListTrue_for_EachObj – список корректных значений всех признаков, используемых для распознавания объектов (обучающая последовательность значений признаков).

Возвращаемое значение – в зависимости от значения ES_TrainMode.

Если автоматический режим обучения ЭС (ES_TrainMode=ES_tmAuto), то =TRUE, если ЭС обучена, или =FALSE в противном случае.

Если ручной режим обучения ЭС (ES_TrainMode=ES_tmManual), то =TRUE, если метод ES_Train_Manual выполнен успешно или =FALSE в противном случае.

На рисунке 5 приведена блок схема, иллюстрирующая функционал метода ES_Train_Start.

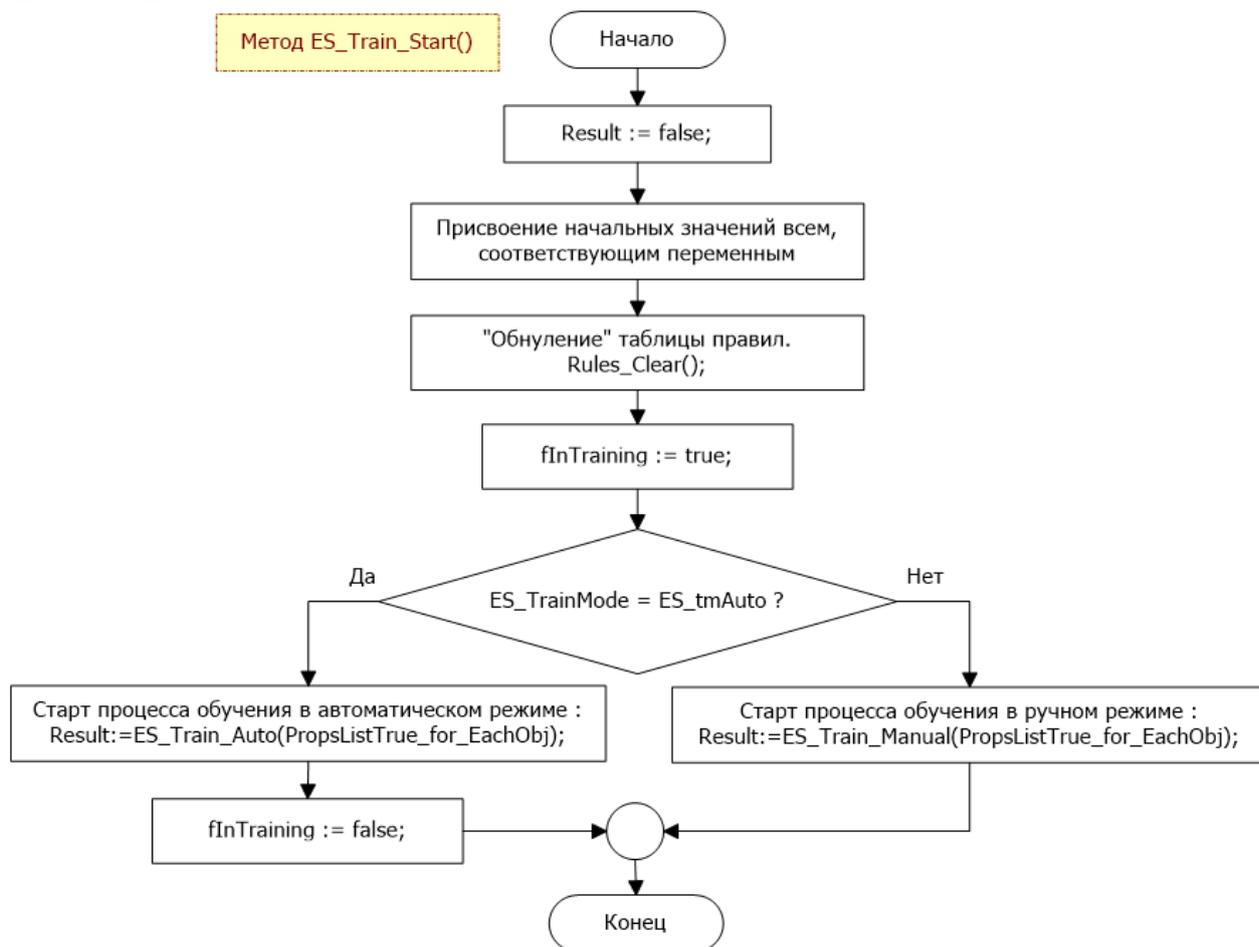


Рисунок 4 – Функционал метода ES_Train_Start

2.2.6 ES_Train_Auto

Назначение: Обучение ЭС в автоматическом режиме.

```
function ES_Train_Auto(
    PropsListTrue_for_EachObj:TStrings
):boolean;
```

Входные параметры:

PropsListTrue_for_EachObj – список корректных значений всех признаков, используемых для распознавания объектов (обучающая последовательность значений признаков).

Возвращаемое значение: если =TRUE, то ЭС обучена (FALSE в противном случае).

На рисунках 13 – 8 приведена блок схема, иллюстрирующая функционал метода ES_Train_Auto.

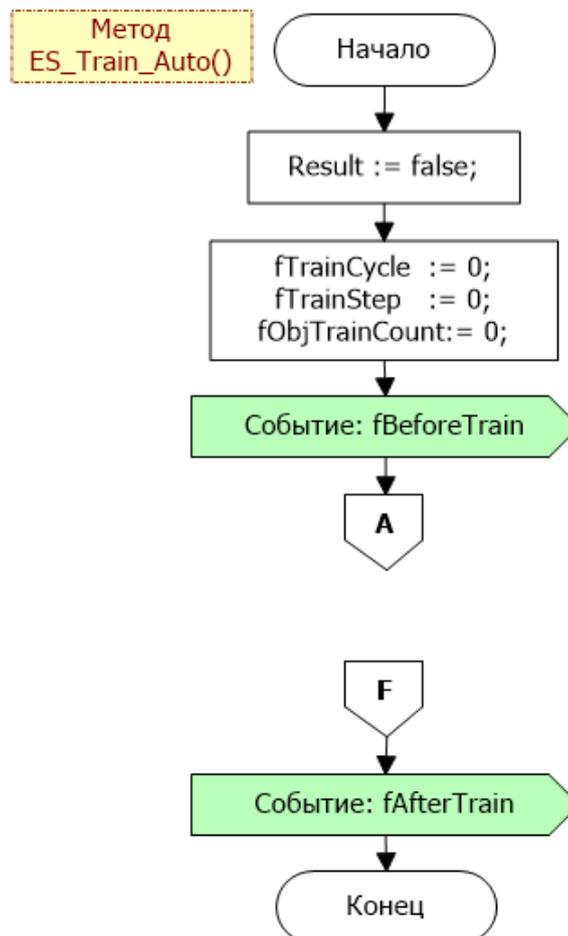


Рисунок 5 – Функционал метода ES_Train_Auto

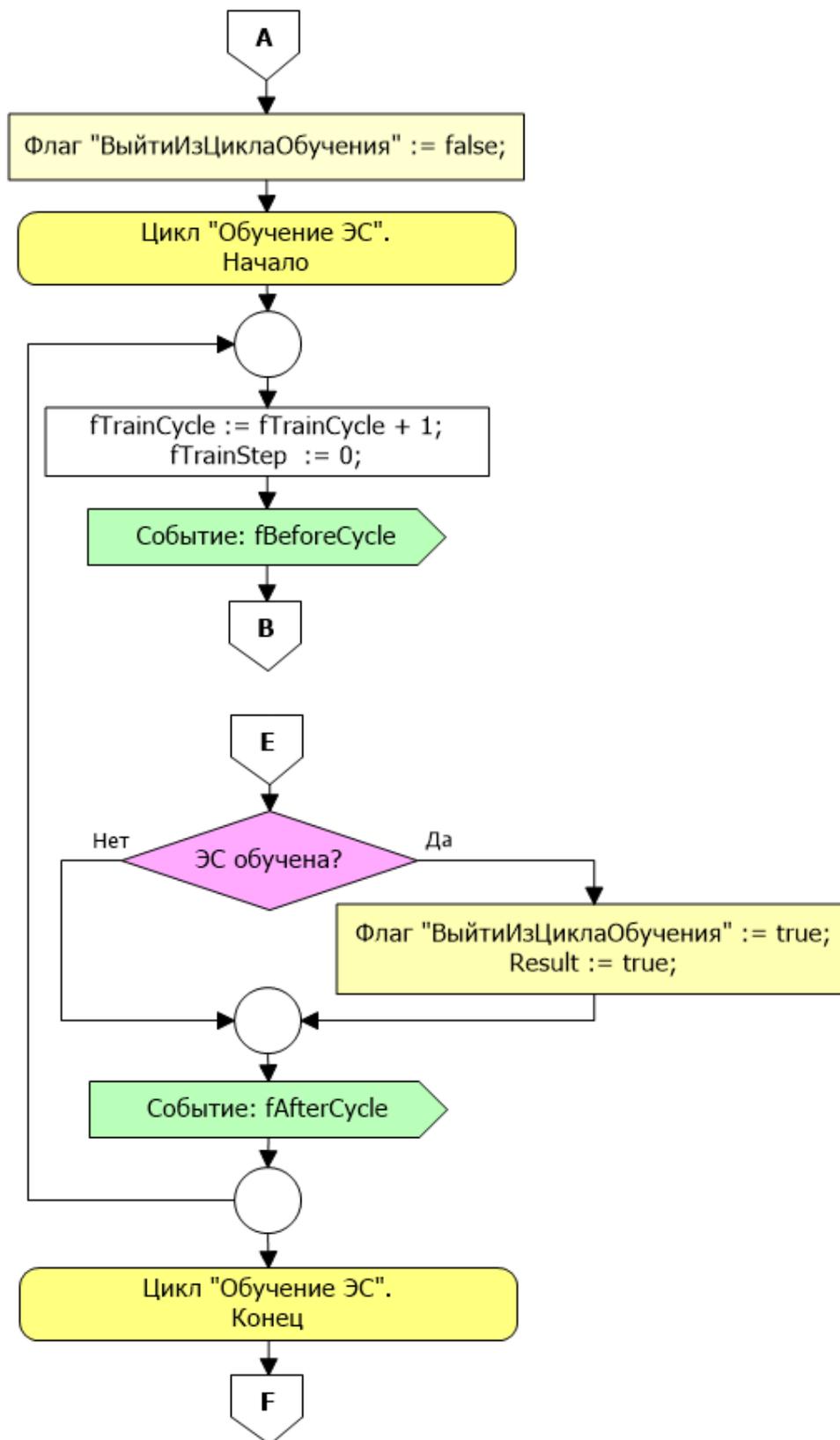


Рисунок 6 – Функционал метода ES_Train_Auto. Продолжение 1

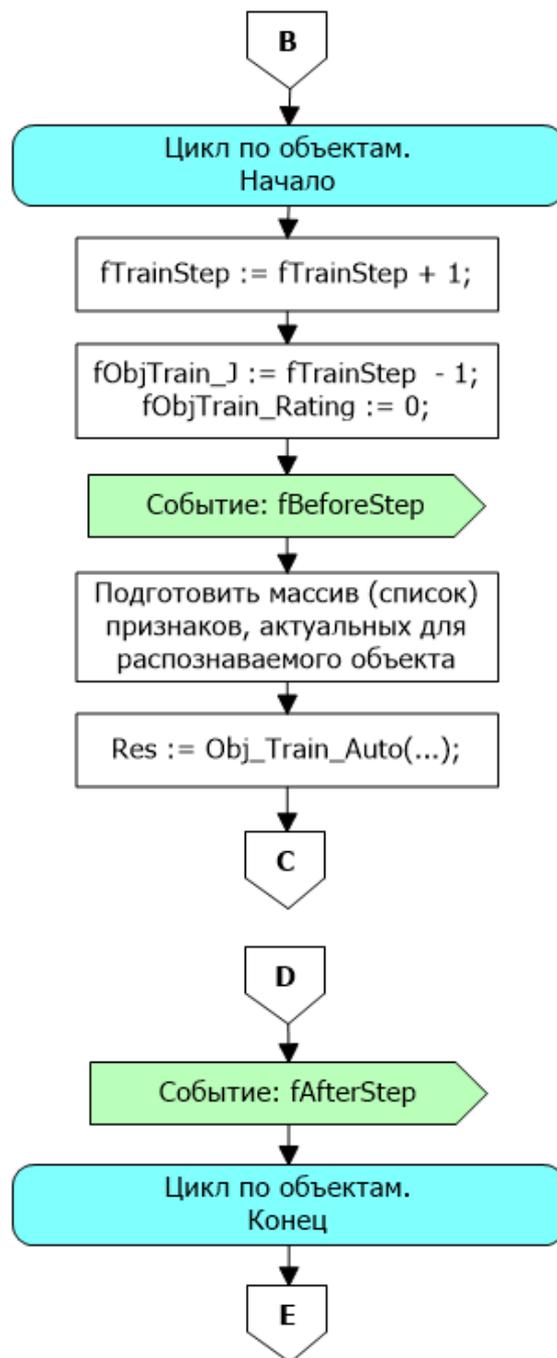


Рисунок 7 – Функционал метода ES_Train_Auto. Продолжение 2

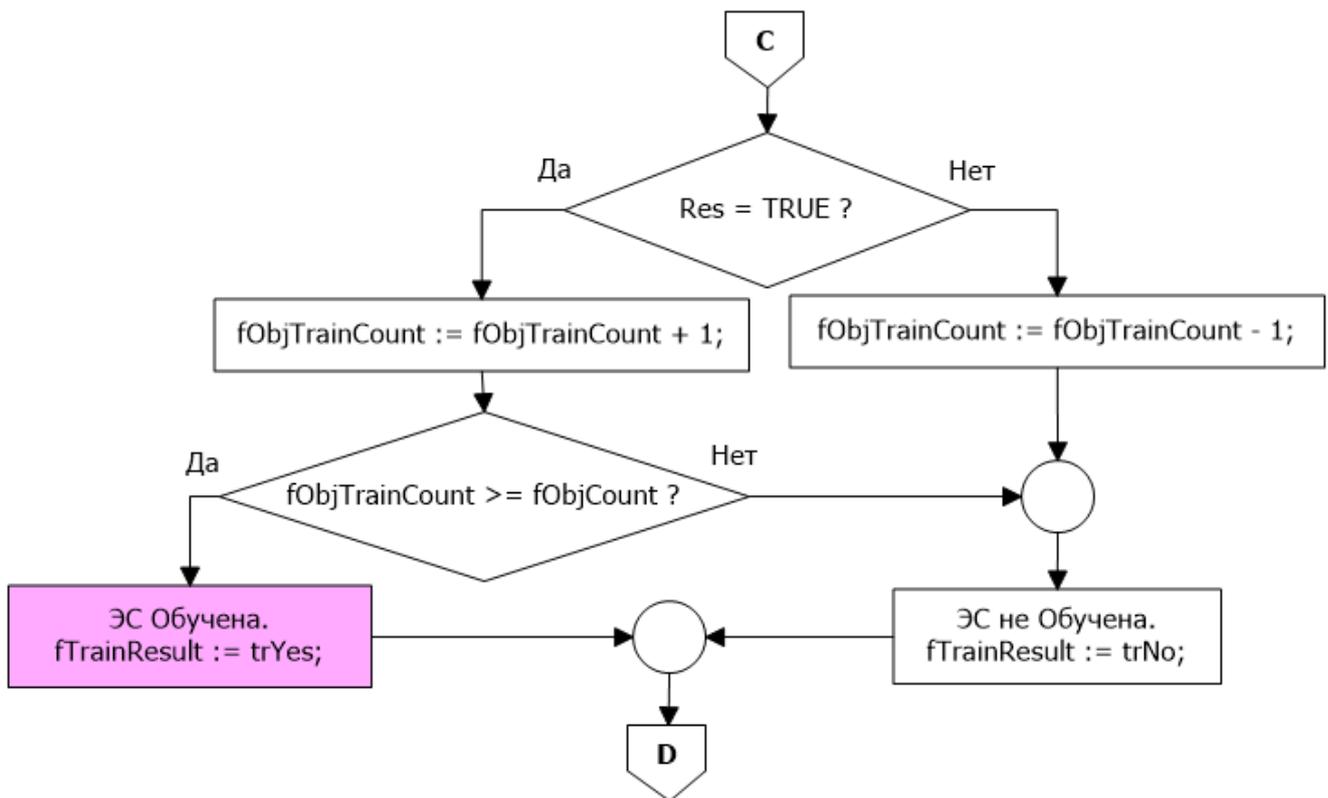


Рисунок 8 – Функционал метода ES_Train_Auto. Продолжение 3

2.2.7 Obj_Train_Auto

Назначение: Обучение ЭС для очередного объекта (очередной шаг обучения) в автоматическом или ручном режиме.

```
function Obj_Train_Auto(ListRat:TStrings):boolean;
```

Входные параметры:

ListRat – список рейтингов всех объектов при обучении ЭС для распознавания заданного объекта.

Возвращаемое значение: если =TRUE, то ЭС успешно распознала заданный объект (FALSE в противном случае).

На рисунках 9 – 10 приведена блок схема, иллюстрирующая функционал метода Obj_Train_Auto.

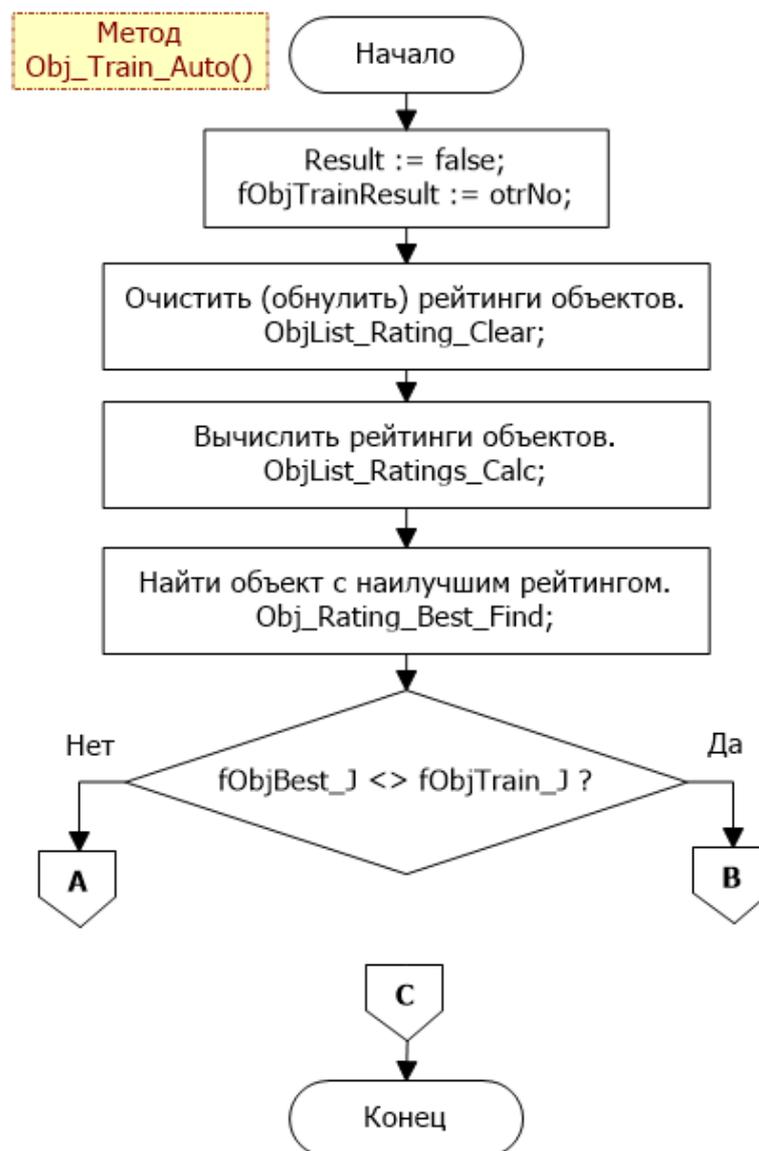


Рисунок 9 – Функционал метода Obj_Train_Auto

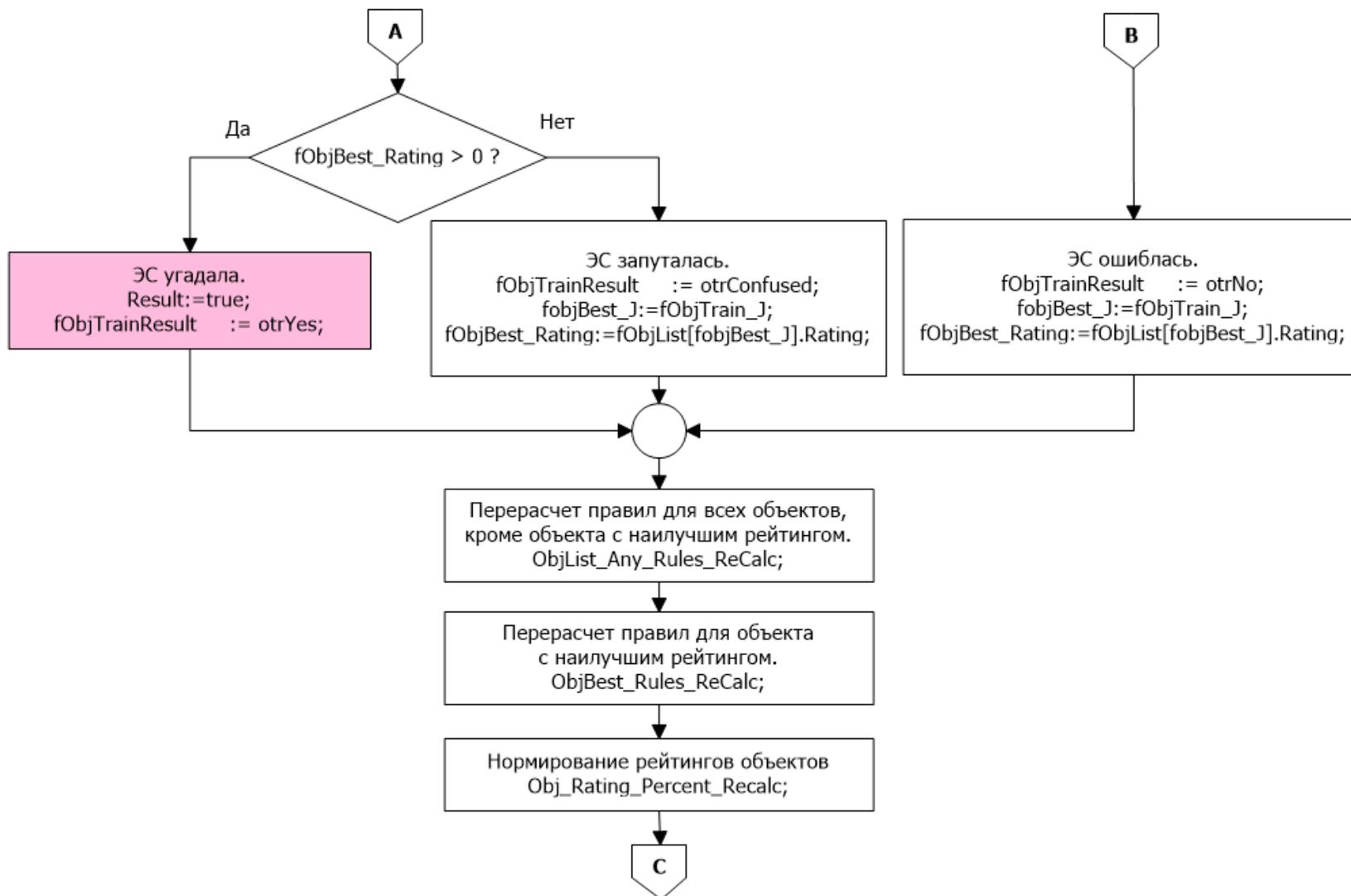


Рисунок 10 – Функционал метода Obj_Train_Auto. Продолжение

2.2.8 ES_Train_Manual_Next

Назначение: Обучение ЭС для очередного объекта (очередной шаг на очередном цикле) в ручном режиме.

```
function ES_Train_Manual_Next(  
    PropsListTrue_for_EachObj:TStrings  
):boolean;
```

Входные параметры:

PropsListTrue_for_EachObj – список корректных значений всех признаков, используемых для распознавания объектов (обучающая последовательность значений признаков).

Возвращаемое значение: если =TRUE, то входной параметр корректен (FALSE в противном случае).

Примечание – возвращаемое значение вряд ли является чем-то существенно важным.

На рисунках 11 – 12 приведена блок схема, иллюстрирующая функционал метода ES_Train_Manual_Next.

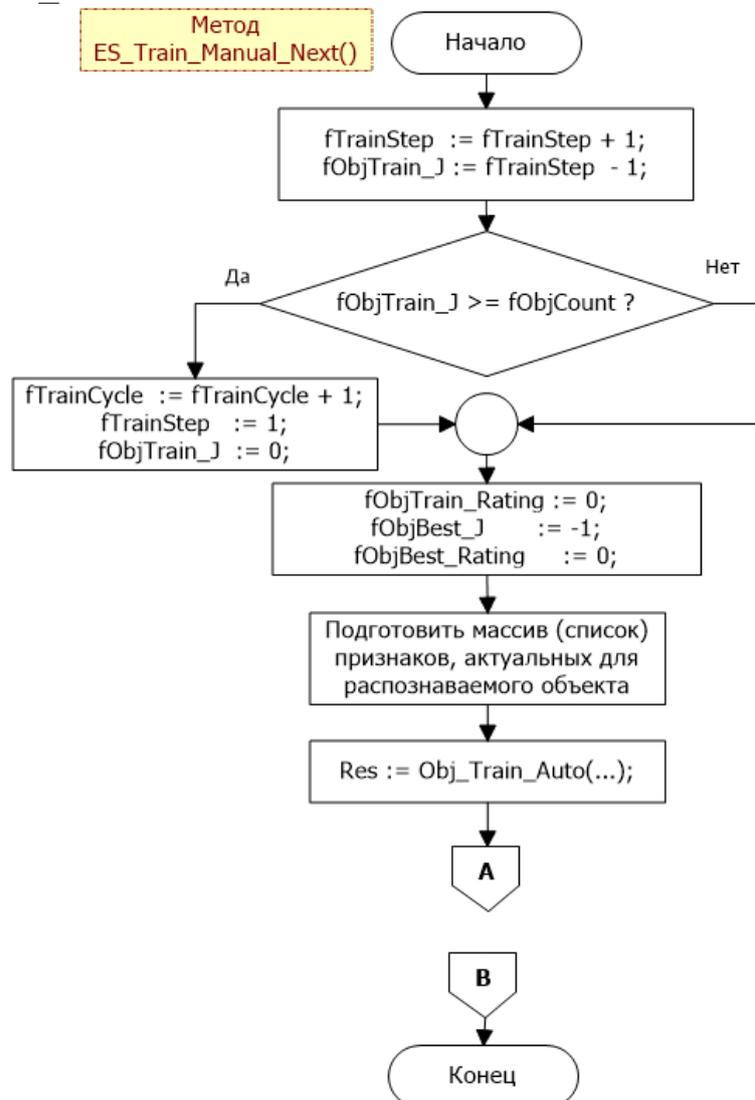


Рисунок 11 – Функционал метода ES_Train_Manual_Next

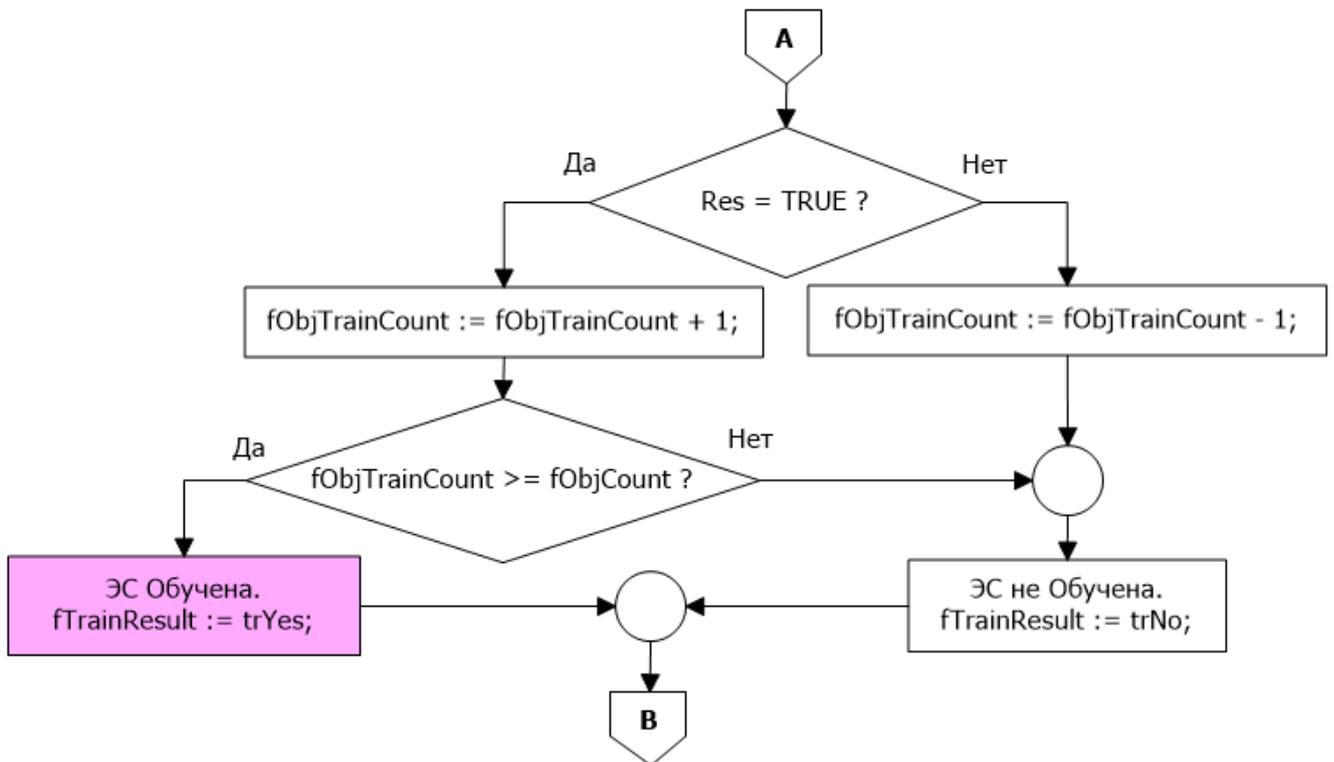


Рисунок 12 – Функционал метода ES_Train_Manual_Next. Продолжение

См., также, раздел «2.2.7 Obj_Train_Auto».

2.2.9 ES_Train_ExitForLimit

Назначение: Досрочное завершение процесса обучения ЭС в автоматическом режиме из-за превышения максимального кол-ва циклов обучения.

```
procedure ES_Train_ExitForLimit;
```

2.2.10 ES_Train_Exit_UserAbort

Назначение: Досрочное завершение процесса обучения ЭС в автоматическом режиме по требованию Пользователя.

```
procedure ES_Train_Exit_UserAbort;
```

2.2.11 Obj_Add

Назначение: Добавить объект в перечень объектов.

```
function Obj_Add(Id:integer):integer;
```

Входные параметры:

Id – идентификатор объекта.

Возвращаемое значение: индекс объекта (начиная с нуля) или -1 в случае ошибки .

Важно!

1. Значение Id должно быть больше 0.
2. Id может быть использовано, при необходимости, для связи с Базой Данных.

2.2.12 Prop_Add

Назначение: Добавить признак (свойство) в перечень признаков (свойств).

```
function Prop_Add(Id:integer; Yes:boolean=false):integer;
```

Входные параметры:

Id – идентификатор свойства.

Yes – значение свойства.

Возвращаемое значение: индекс свойства (начиная с нуля) или -1 в случае ошибки .

Важно!

1. Значение Id должно быть больше 0.
2. Id может быть использовано, при необходимости, для связи с Базой Данных.

2.2.13 ObjList_LoadFromTStrings

Назначение: Загрузить список распознаваемых объектов из TStrings.

```
function ObjList_LoadFromTStrings(List:TStrings):boolean;
```

Входные параметры:

List – список распознаваемых объектов в формате TStrings.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

Формат List типа TStrings:

1. Одна строка – одно свойство (признак).
2. Формат строки: **Id=Значение** (1).
3. Значение Id должно быть больше 0.

Важно! Id может быть использовано, при необходимости, для связи с Базой Данных.

Пример:

```
1=1
2=1
3=1
```

2.2.14 PropList_LoadFromTStrings

Назначение: Загрузить список признаков (свойств) из TStrings.

```
function PropList_LoadFromTStrings(List:TStrings):boolean;
```

Входные параметры:

List – список свойств и их значений в формате TStrings.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

Формат List типа TStrings:

4. Одна строка – одно свойство (признак).
5. Формат строки: **Id=Значение** (0 или 1).
6. Значение Id должно быть больше 0.

Важно! Id может быть использовано, при необходимости, для связи с Базой Данных.

Пример:

```
1=0
2=0
3=1
4=1
5=0
```

2.2.15 PropsValue_from_IntArray

Назначение: Импорт значений свойств из массива целых чисел (0–false, 1–true).

```
function PropsValue_from_IntArray(
    ArrayProp:array of integer;
    ArraySize:integer
):boolean;
```

Входные параметры:

ArrayProp – целочисленный массив (список) значений свойств;
ArraySize – размер массива (кол-во свойство).

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.16 PropsValue_from_TStrings

Назначение: Импорт значений свойств из TStrings ('0'—false, '1'—true).

```
function PropsValue_from_TStrings(
    ListProp:TStrings
):boolean;
```

Входные параметры:

ListProp – список значений свойств в формате TStrings.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.17 PropsValue_from_TCheckListBox

Назначение: Импорт значений свойств из TCheckListBox.

```
function PropsValue_from_TCheckListBox(
    ListProp:TCheckListBox
):boolean;
```

Входные параметры:

ListProp – список значений свойств в формате TCheckListBox.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.18 PropsValue_Control

Назначение: Контроль, что значение хотя бы одного из признаков (свойств) = TRUE.

```
function PropsValue_Control:boolean;
```

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.19 Obj_Rating_TStringGrid_Clear

Назначение: Создать пустую (из 2-х строк) таблицу рейтингов объектов (TStringGrid).

```
function Obj_Rating_TStringGrid_Clear(SG:TStringGrid):boolean;
```

Входные параметры:

SG – таблица рейтингов в формате TStringGrid.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.20 Obj_Rating_to_TStringGrid

Назначение: Выгрузить рейтинги объектов в таблицу рейтингов (TStringGrid).

```
function Obj_Rating_to_TStringGrid(SG:TStringGrid):boolean;
```

Входные параметры:

SG – таблица рейтингов в формате TStringGrid.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.21 Obj_Rating_Percent_Recalc

Назначение: Расчет нормированных значений рейтинга объектов (в том числе и в процентах).

```
function Obj_Rating_Percent_Recalc:boolean;
```

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.22 Rules_TStringGrid_Clear

Назначение: Создать пустую (из двух строк) таблицу правил для распознавания объектов (TStringGrid).

```
function Rules_TStringGrid_Clear(SG:TStringGrid):boolean;
```

Входные параметры:

SG – таблица правил в формате TStringGrid.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.23 Rules_to_TStringGrid

Назначение: Экспорт таблицы правил из внутреннего представления в TStringGrid.

```
function Rules_to_TStringGrid(SG:TStringGrid):boolean;
```

Входные параметры:

SG – таблица правил в формате TStringGrid.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.2.24 Rules_From_TStringGrid

Назначение: Импорт таблицы правил из TStringGrid во внутренний формат.

```
function Rules_From_TStringGrid(SG:TStringGrid):boolean;
```

Входные параметры:

SG – таблица правил в формате TStringGrid.

Возвращаемое значение: =TRUE, если успешно или =FALSE в противном случае .

2.3 События компонента

В таблице 4 приведены события компонента.

Таблица 4 – События компонента

| Событие | Раздел | Назначение |
|--------------------------|-------------------|--|
| 1 | 2 | 3 |
| fBeforeTrain/BeforeTrain | private/published | Перед стартом процесса АвтоОбучения ЭС |
| fAfterTrain/AfterTrain | private/published | После останова/прерывания процесса АвтоОбучения ЭС |
| fBeforeCycle/BeforeCycle | private/published | В начале очередного цикла (режим АвтоОбучения) |
| fAfterCycle/AfterCycle | private/published | В конце очередного цикла (режим АвтоОбучения) |
| fBeforeStep/BeforeStep | private/published | В начале каждого шага (обучение ЭС для распознавания очередного объекта) |
| fAfterStep/AfterStep | private/published | В конце каждого шага (обучение ЭС для распознавания очередного объекта) |

2.3.1 fBeforeTrain/BeforeTrain

private/published Перед стартом процесса АвтоОбучения ЭС

Разделы: private/published.

Назначение: Перед стартом процесса АвтоОбучения ЭС.

Тип: TES_Common_Event.

2.3.2 fAfterTrain/AfterTrain

Разделы: private/published.

Назначение: После останова/прерывания процесса АвтоОбучения ЭС.

Тип: TES_Common_Event.

2.3.3 fBeforeCycle/BeforeCycle

Разделы: private/published.

Назначение: В начале очередного цикла (режим АвтоОбучения).

Тип: TES_Auto_Cycle_Before_Event.

2.3.4 fAfterCycle/AfterCycle

Разделы: private/published.

Назначение: В конце очередного цикла (режим АвтоОбучения).

Тип: TES_Auto_Cycle_After_Event.

2.3.5 fBeforeStep/BeforeStep

Разделы: private/published.

Назначение: В начале каждого шага (обучение ЭС для распознавания очередного объекта).

Тип: TES_Auto_Cycle_Step_Before_Event.

2.3.6 fAfterStep/AfterStep

Разделы: private/published.

Назначение: В конце каждого шага (обучение ЭС для распознавания очередного объекта).

Тип: TES_Auto_Cycle_Step_After_Event.

2.4 Обучение ЭС

2.4.1 Обучение ЭС в автоматическом режиме

На рисунке 13 приведена обобщенная схема, иллюстрирующая действия Пользователя в процессе обучения ЭС в автоматическом режиме.

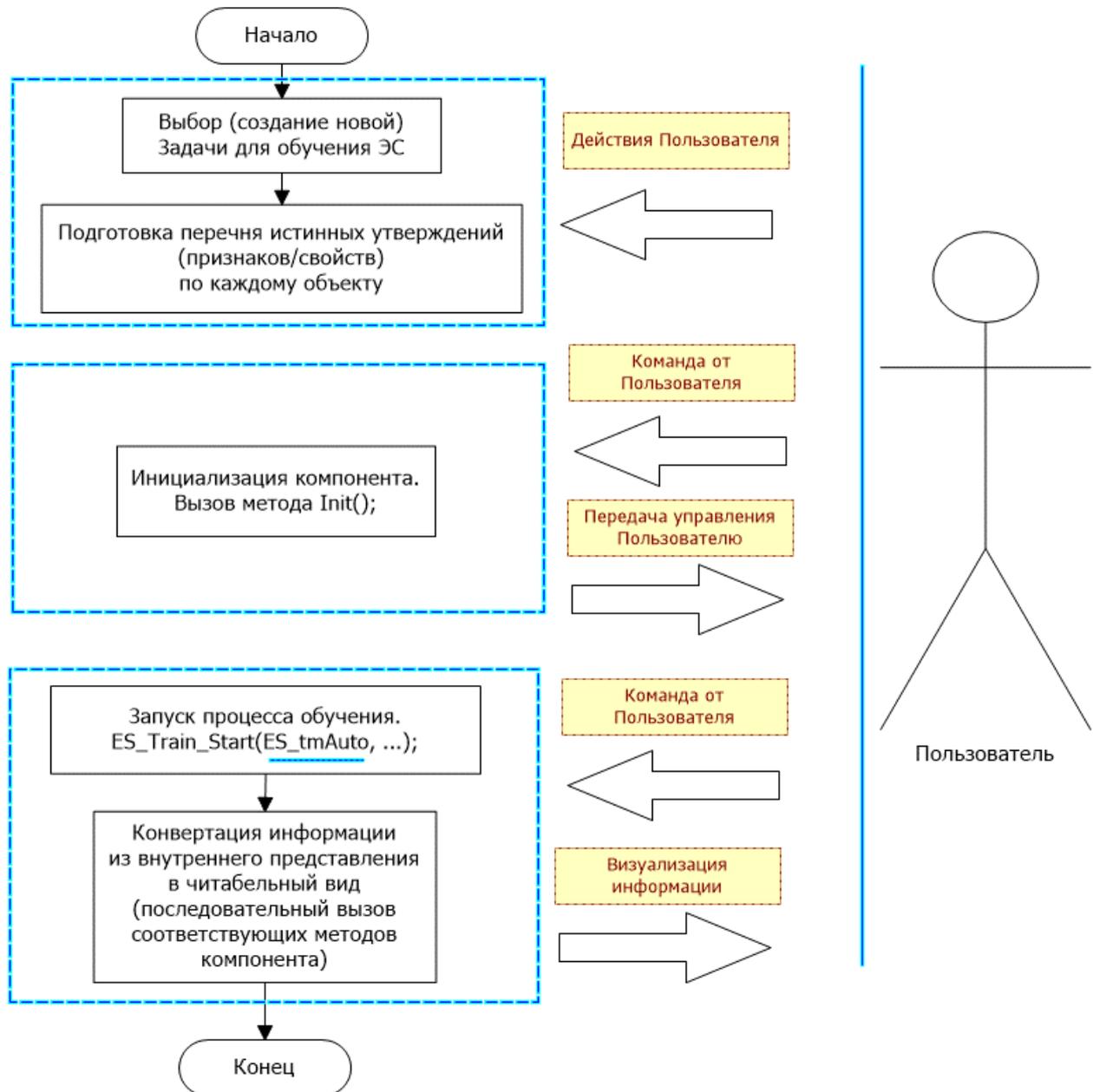


Рисунок 13 – Обобщенная схема, иллюстрирующая действия Пользователя в процессе обучения ЭС в автоматическом режиме

2.4.2 Обучение ЭС в ручном режиме

На рисунках 14 и 15 приведена обобщенная схема, иллюстрирующая действия Пользователя в процессе обучения ЭС в ручном режиме.

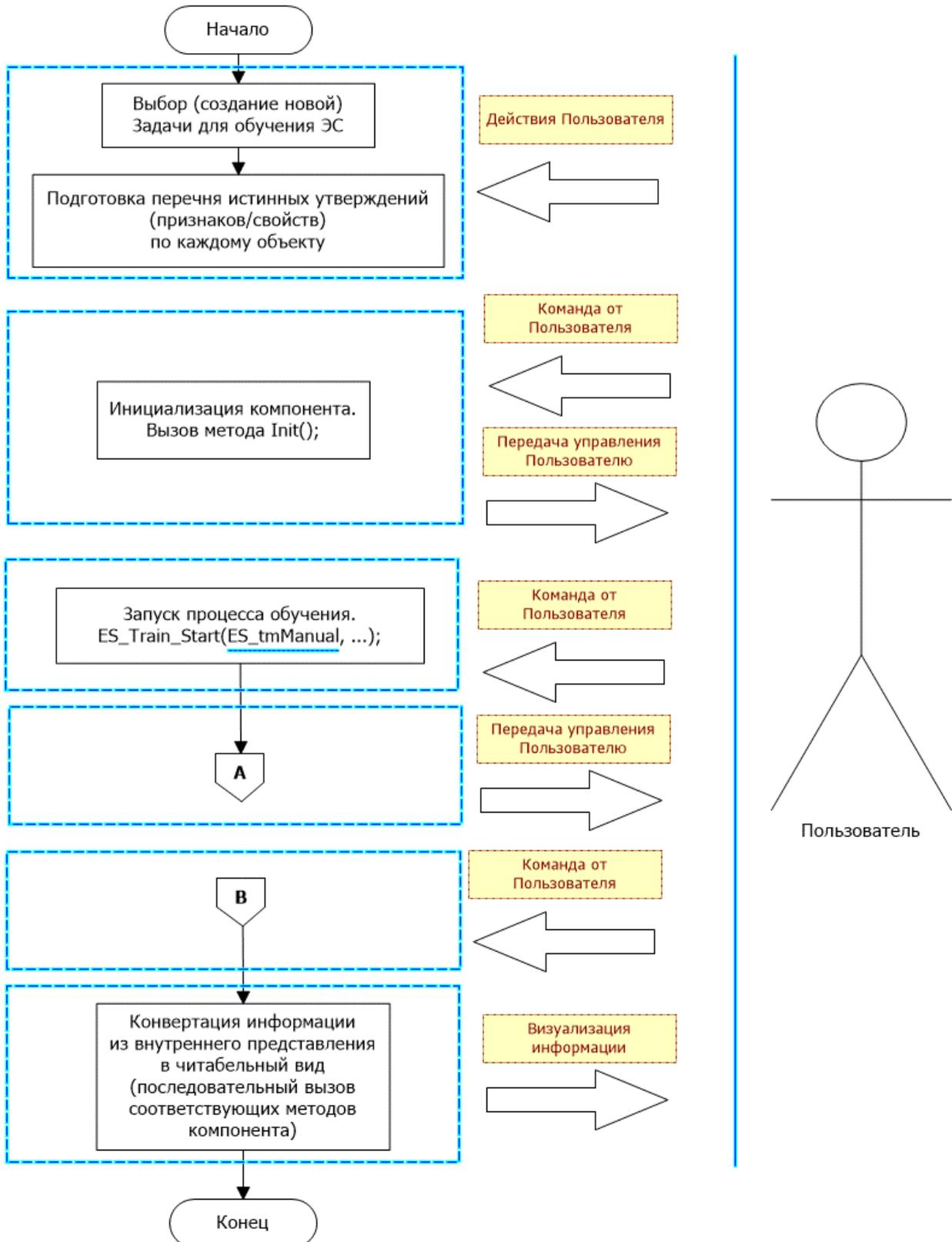


Рисунок 14 – Обобщенная схема, иллюстрирующая действия Пользователя в процессе обучения ЭС в ручном режиме

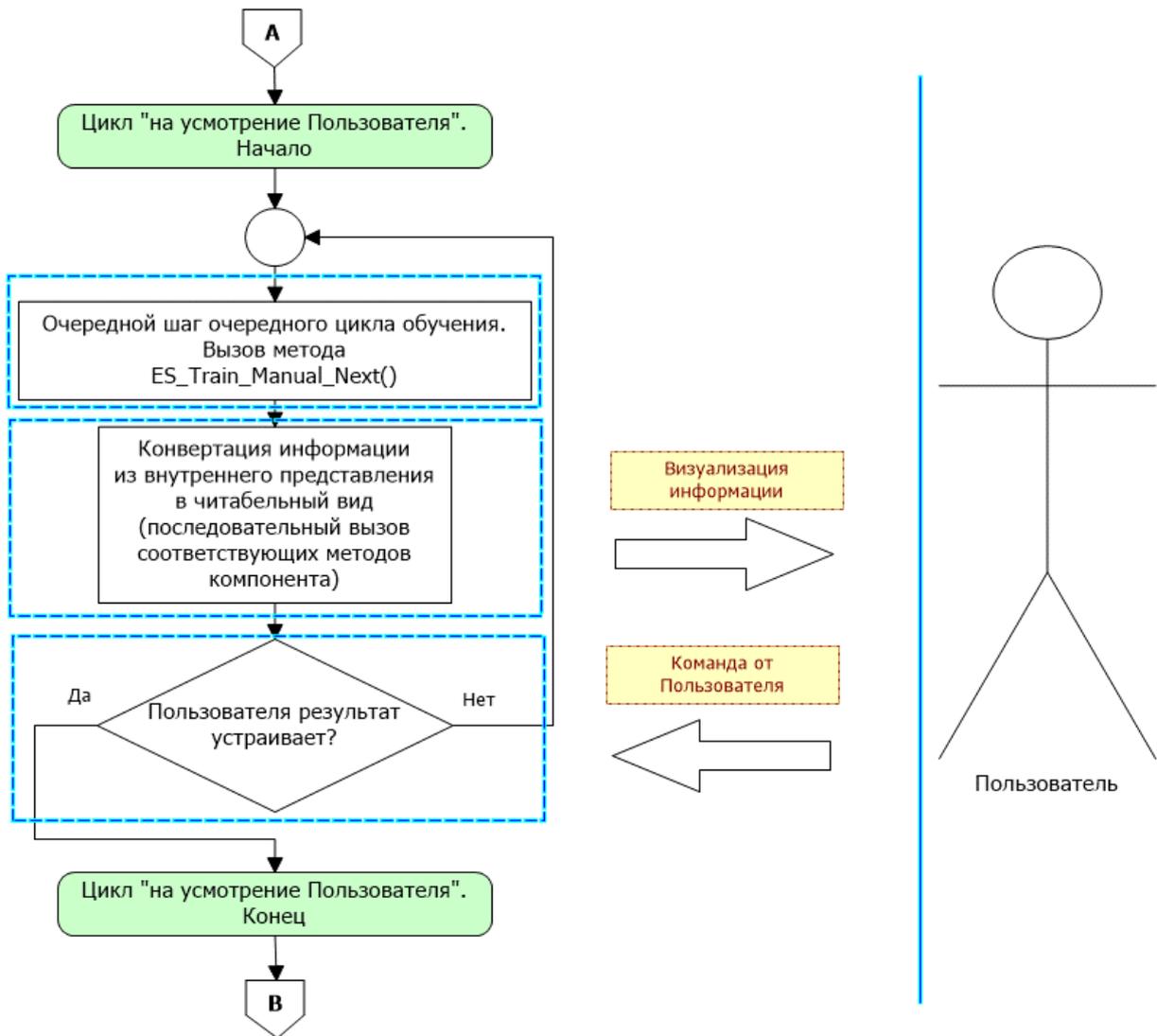


Рисунок 15 – Обобщенная схема, иллюстрирующая действия Пользователя в процессе обучения ЭС в ручном режиме. Продолжение

3 Использование компонента TES_bayes на примере простой программы

В этом разделе приведено краткое описание простой программы, иллюстрирующей применение компонента TES_bayes.

Исполняемый модуль (файл) программы: ES_Bayes.exe.

3.1 Основные операции

Все операции выполняются в рамках главной формы программы (см. рисунок 16).

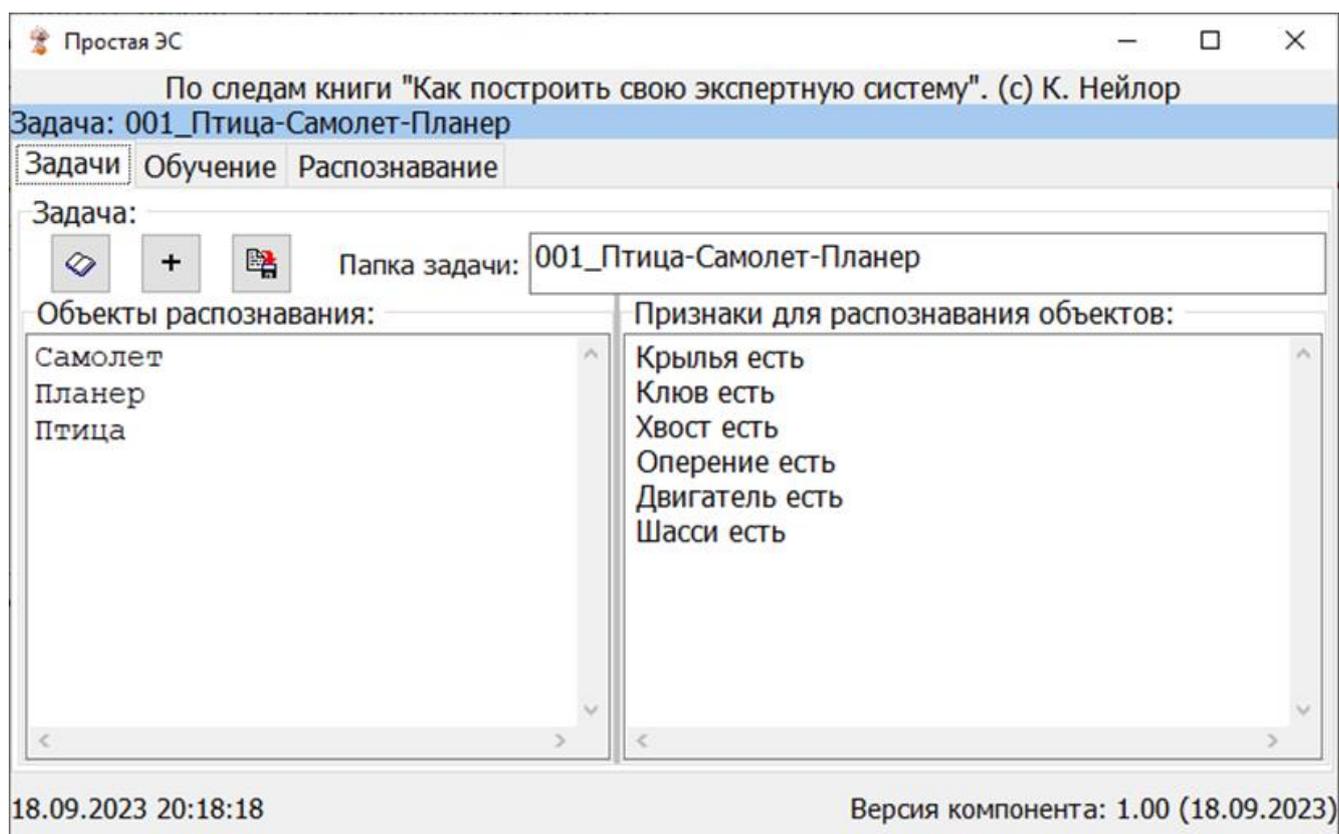


Рисунок 16 – Главная форма программы. Вкладка «Задачи»

Описание всех операций приведено в разделах ниже.

В качестве иллюстрирующего примера рассматривается задача распознавания (по внешним признакам) грибов 2-х видов: Белый гриб и Подберезовик.

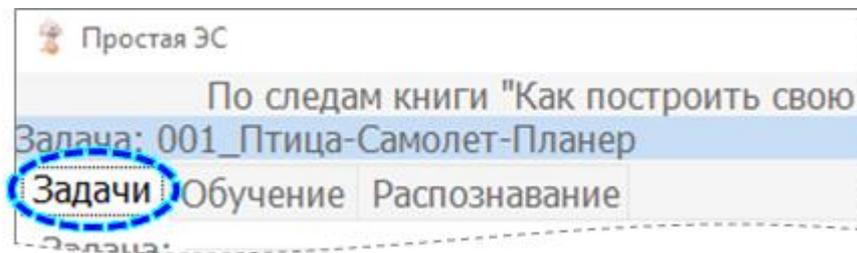
Признаки, которые используются для распознавания грибов:

- ножка покрыта мелкими серыми или черными чешуйками;
- ножка (на срезе) быстро темнеет;
- внутренняя сторона шляпки желтая;
- внутренняя сторона шляпки белая;
- сильный запах.

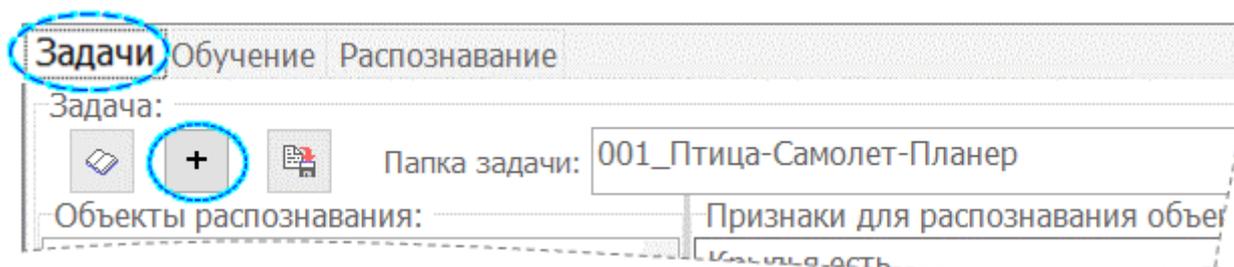
3.1.1 Формирование перечня задач

3.1.1.1 Создать новую задачу

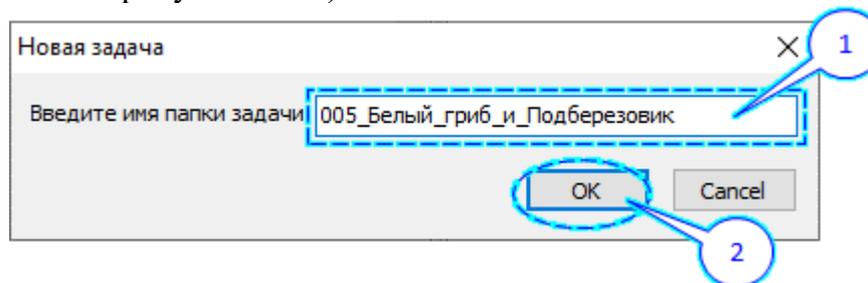
1. Открыть вкладку «Задачи» (см. рисунок ниже).



2. Нажать на кнопку  (см. рисунок ниже).



3. Ввести наименование задачи (выноска-1 на рисунке ниже) и нажать на кнопку ОК (выноска-2 на рисунке ниже).



ВАЖНО! Наименование задачи – это имя новой папки (каталога) на диске (со всеми вытекающими). Т.е., крайне не рекомендуется использовать какие-то спец. символы.

На рисунке 17 приведен результат (см. выноска-1).

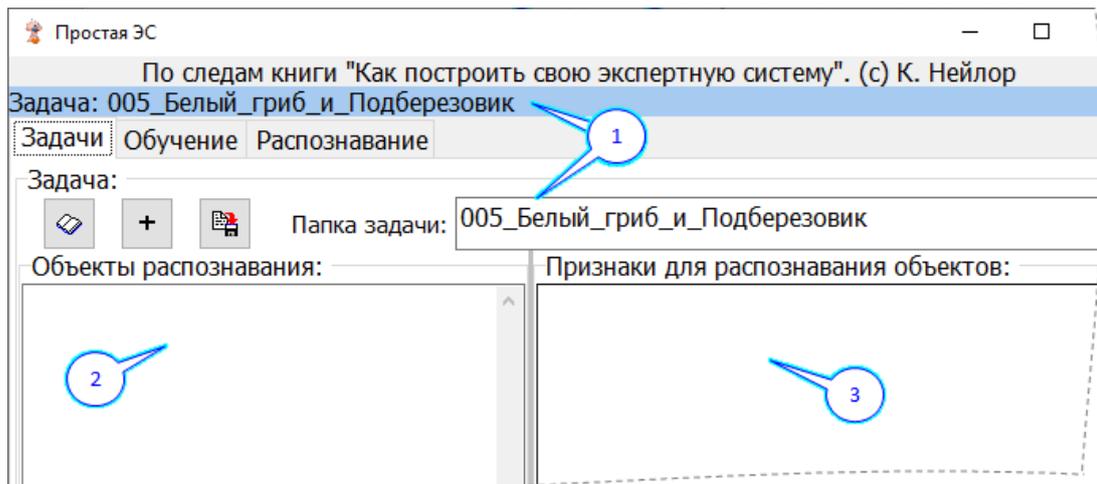
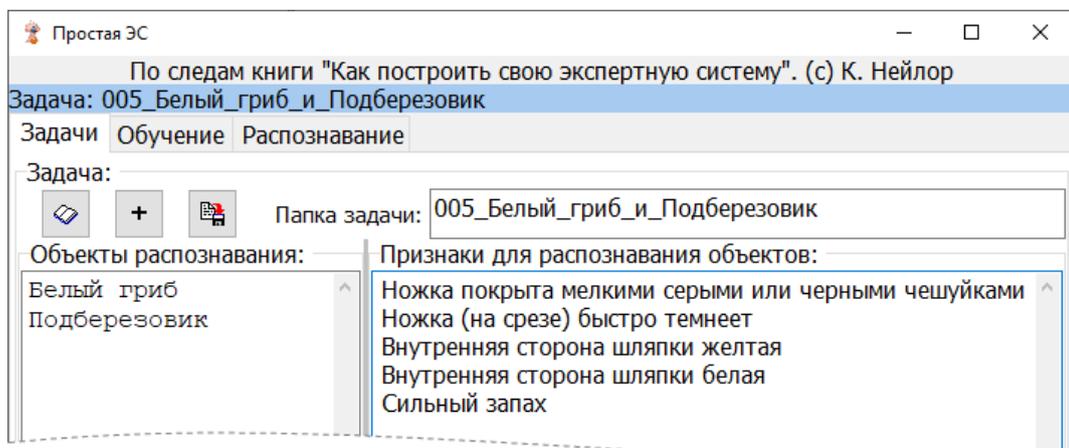


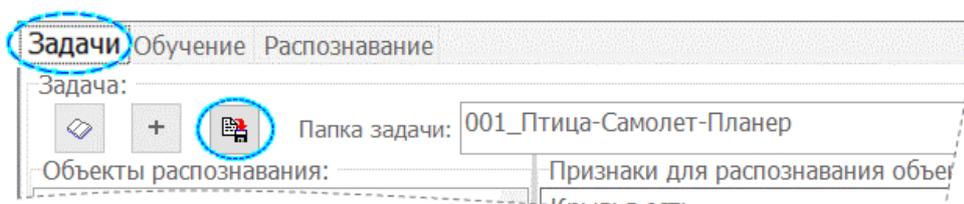
Рисунок 17 – Создание новой задачи

4. В области формы, отмеченной выноской-2, необходимо ввести список объектов, предназначенных для распознавания.
Ввод данных производится прямо с клавиатуры (поскольку это обычный ТМемо).
5. В области формы, отмеченной выноской-3, необходимо ввести полный список признаков для всех объектов.
Ввод данных производится прямо с клавиатуры (поскольку это обычный ТМемо).

Результат приведен на рисунке ниже.

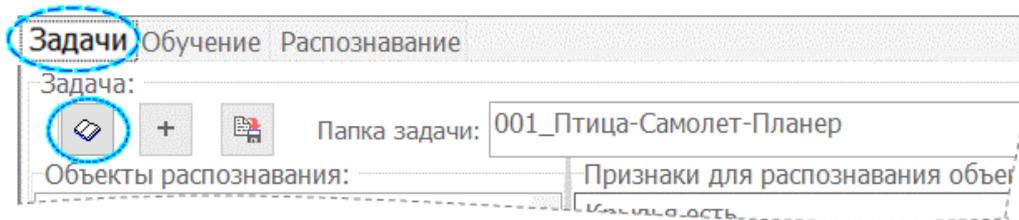


6. Далее, следует нажать на кнопку  (см. рисунок ниже).

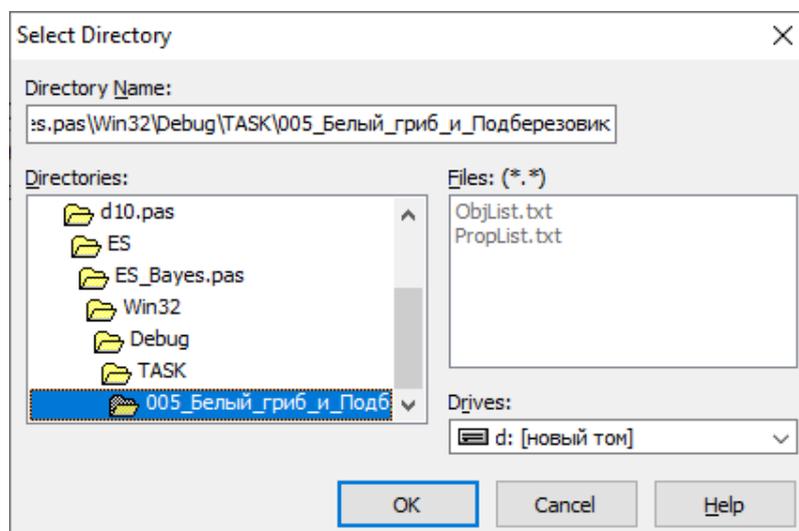


3.1.1.2 Выбрать и открыть существующую задачу

1. Нажать на кнопку  (см. рисунок ниже).



2. В открывшемся диалоговом окне выбрать папку соответствующей задачи и нажать на кнопку ОК (см. рисунок ниже).



3.1.1.3 Редактировать задачу

Редактирование задачи производится простым исправление (добавлением) текста прямо в соответствующем поле ТМето (см. рисунок 17).

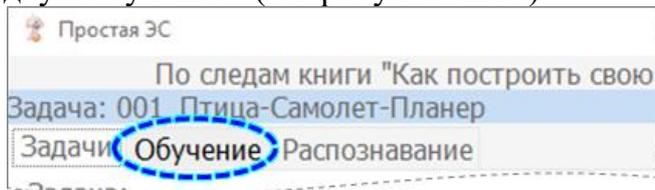
3.1.1.4 Сохранить изменения на диск

Для сохранения изменений на диск, следует нажать на кнопку .

3.1.2 Обучение ЭС для выбранной задачи

3.1.2.1 Общие операции

1. Открыть вкладку «Обучение» (см. рисунок ниже).



2. В том случае, если данная операция ранее не выполнялась (или данные не были сохранены на диск) – для КАЖДОГО объекта (см. рисунок 18, выноска-1) отметить «галочкой» те признаки (см. выноска-2), которые являются для него истинными и сохранить их, нажав на кнопку  (см. выноску-3 на рисунке 18).

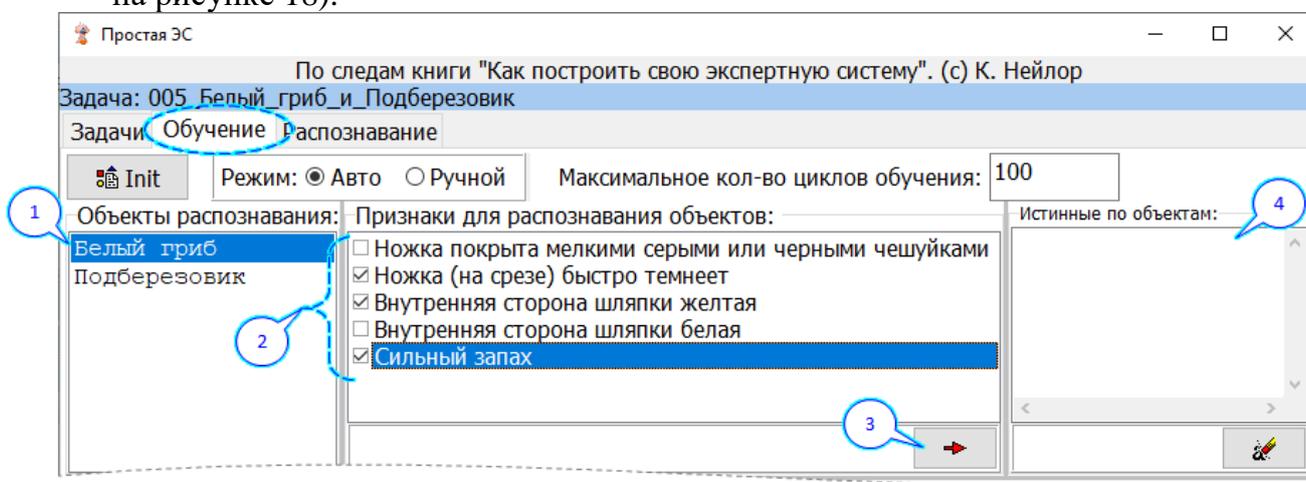
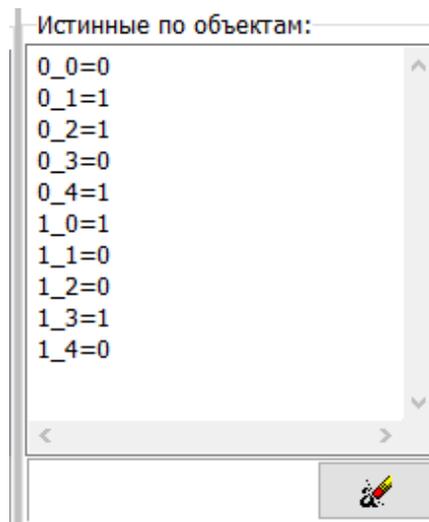


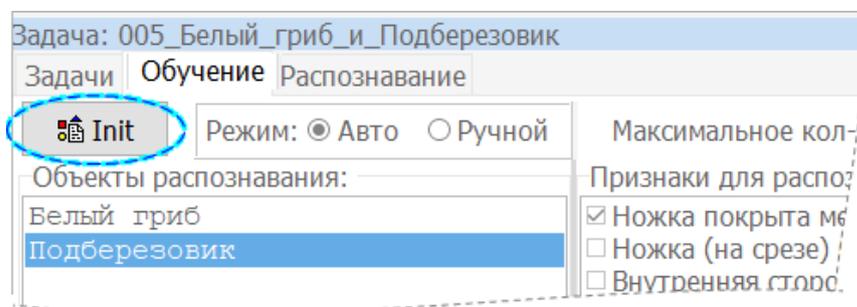
Рисунок 18 – Сформировать актуальные значения признаков для **КАЖДОГО** объекта

Для этого примера список сохраненных значений будет таким (см. рисунок 18, выноска-4 и рисунок ниже).



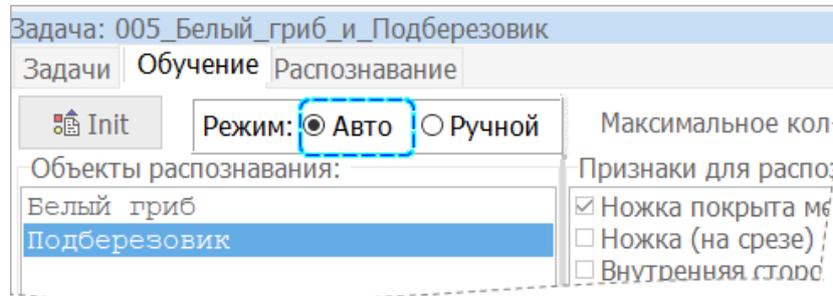
Важно! Термин «Сохранить» в данном случае – не точен. Значения признаков сохраняются только в список (см. рисунок выше), но не на диск. На диск этот список сохранится только, когда будет нажата кнопка «Старт» (см. разделы ниже).

3. Нажать на кнопку «**Init**» (см. рисунок ниже).

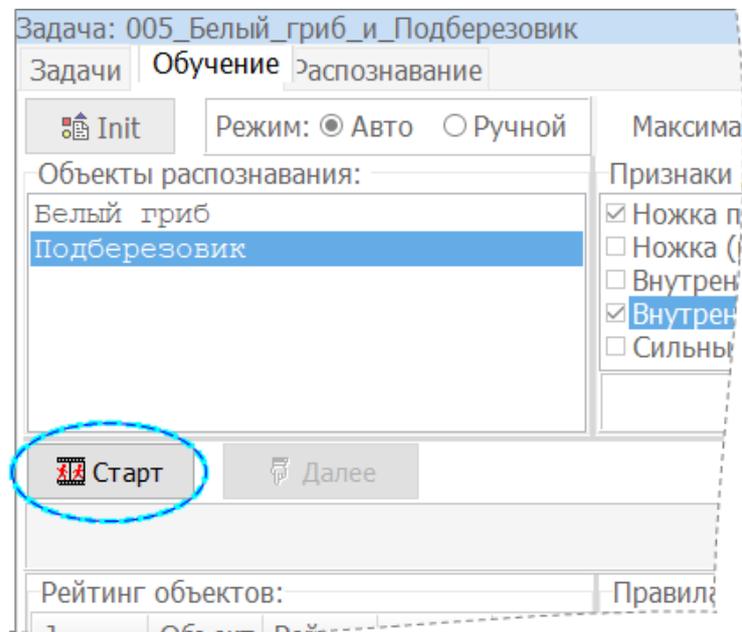


3.1.2.2 Автоматический режим

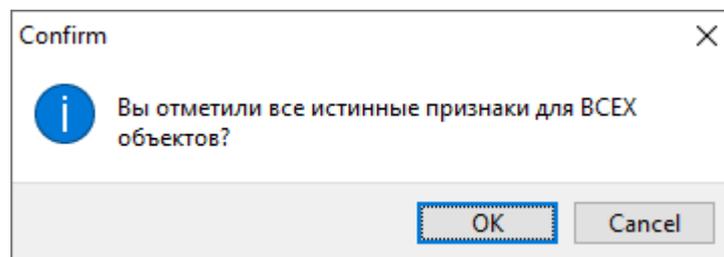
1. Выполнить операции, указанные в разделе «3.1.2.1 Общие операции».
2. Выбрать режим «Авто» (см. рисунок ниже).



3. Нажать на кнопку «Старт» (см. рисунок ниже).



4. В ответ на запрос (см. рисунок ниже) нажать на кнопку ОК.



5. Дождаться окончания процесса обучения ЭС (см. рисунок ниже).
Процесс обучения для простых задач занимает мало времени (секунды).

Обучение ЭС. Стоп. Результат: ЭС обучена

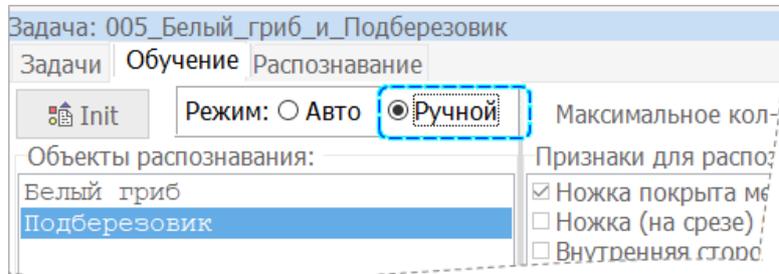
Цикл-3. Объект-1 (Подбере:

| Рейтинг объектов: | | | | | | Правила распознавания: | | | Результат: |
|-------------------|--------------|---------|--------------|-----------|----|------------------------|----|----|--|
| J | Объект | Рейтинг | Рейтинг(норм | Рейтинг,% | id | i \ j | 1 | 2 | |
| 1 | Белый гриб | -2 | 0 | 0 | 1 | 1 | -1 | 3 | Режим обучения: Автоматический Причина окончания обучения: Норм Результат обучения: ЭС обучена |
| 2 | Подберезовик | 4 | 6 | 100 | 2 | 2 | 3 | -1 | |
| | | | | | | 3 | 3 | -1 | |
| | | | | | | 4 | -1 | 3 | |
| | | | | | | 5 | 3 | -1 | |

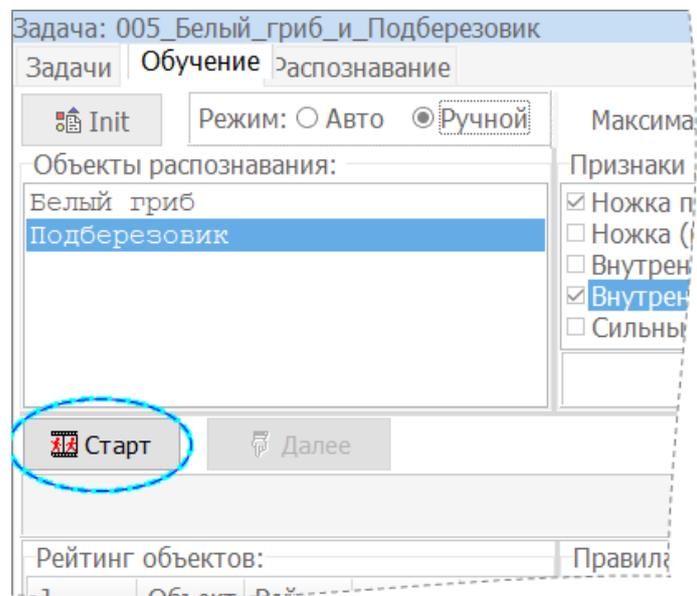
Правила распознавания объектов для этой задачи

3.1.2.3 Ручной режим

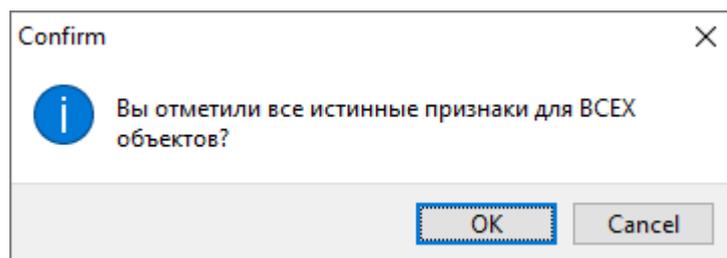
1. Выполнить операции, указанные в разделе «3.1.2.1 Общие операции».
2. Выбрать режим «**Ручной**» (см. рисунок ниже).



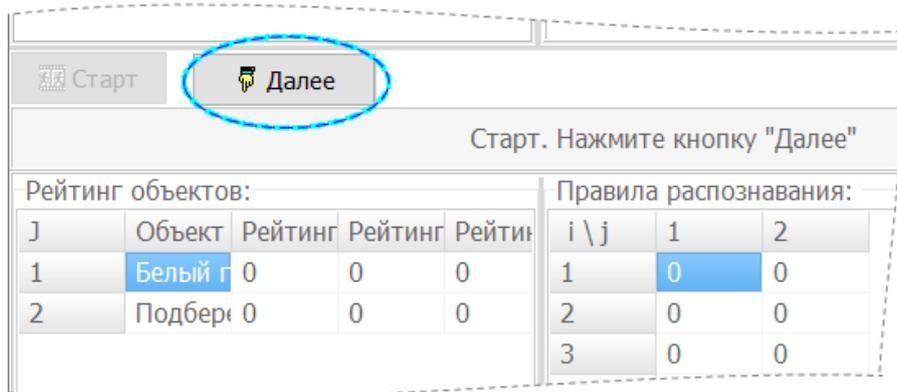
3. Нажать на кнопку «Старт» (см. рисунок ниже).



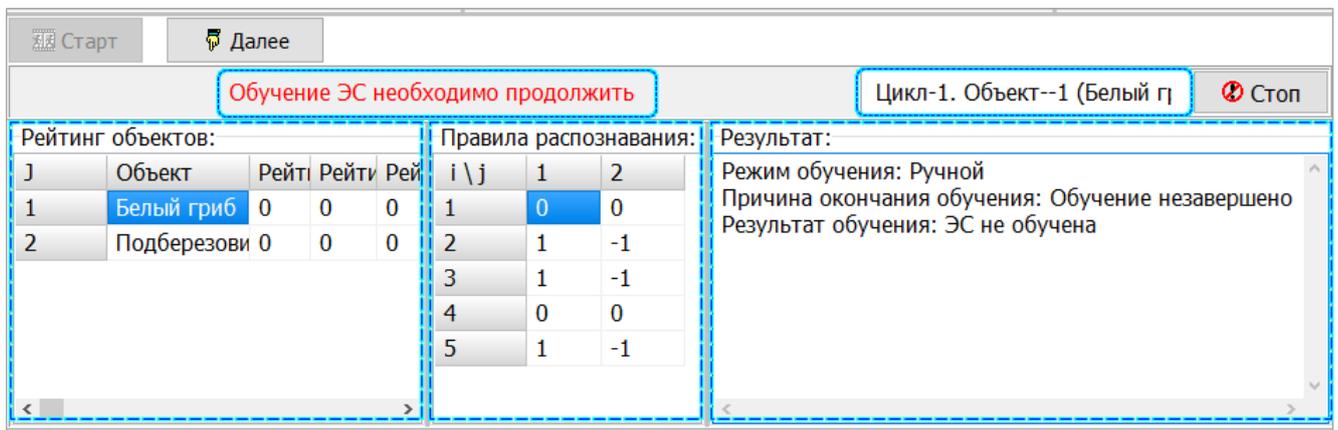
4. В ответ на запрос (см. рисунок ниже) нажать на кнопку ОК.



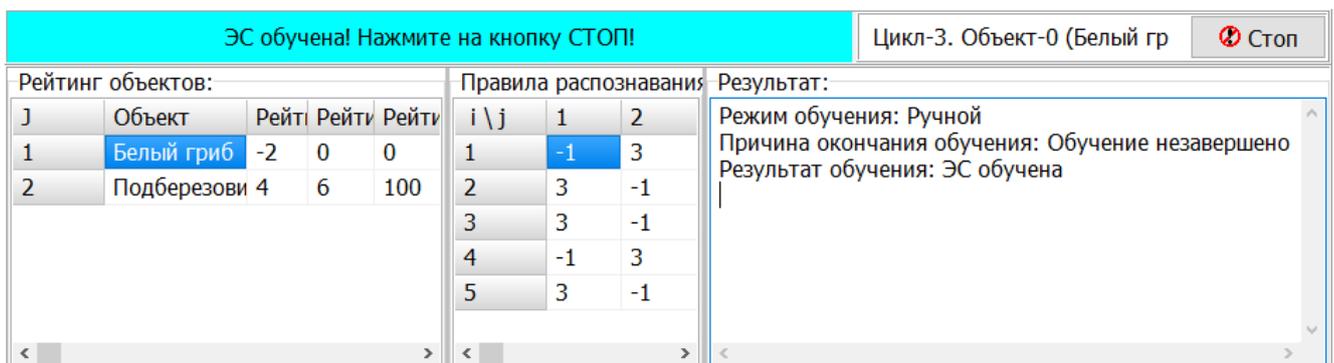
5. Нажать на кнопку «Далее» (см. рисунок ниже).



6. Произвести визуальный контроль информации (см. рисунок ниже).



7. Выполнять поочередно операции (действия), приведенные в пунктах 5 и 6 (см. выше) до тех пор, пока результат обучения ЭС не будет (на Ваш взгляд) удовлетворительным (см., например, рисунок ниже).



8. Нажать на кнопку «**Стоп**» (см. рисунок ниже).

The screenshot shows a software interface with a cyan header bar. The header contains the text "ЭС обучена! Не на кнопку СТОП!" on the left and "Цикл-3. Объект-0 (Белый гр" on the right. A red circle highlights a "Стоп" button with a red stop sign icon in the top right corner.

Below the header, there are three main sections:

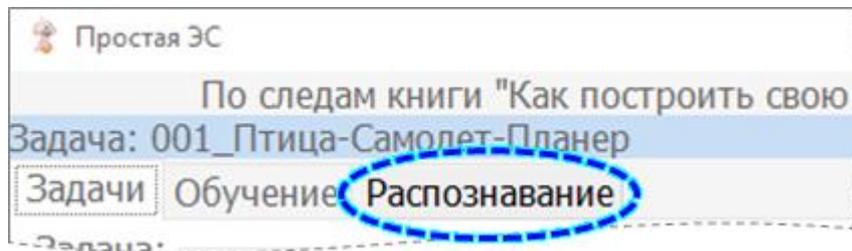
- Рейтинг объектов:** A table with columns "J", "Объект", "Рейт", and "Рейт".

| J | Объект | Рейт | Рейт |
|---|-------------|------|------|
| 1 | Белый гриб | -2 | 0 |
| 2 | Подберезови | 4 | 6 |
- Правила распознавания:** A table with columns "i \ j", "1", and "2".

| i \ j | 1 | 2 |
|-------|----|---|
| 1 | -1 | |
| 2 | 3 | |
| 3 | 3 | |
- Результат:** Text area containing:
 - Режим обучения: Ручной
 - Причина окончания обучения: Обучение незавершено
 - Результат обучения: ЭС обучена

3.1.3 Проверка качества обучения ЭС: распознавание объектов в рамках выбранной задачи

1. Открыть вкладку «Распознавание» (см. рисунок ниже).



2. Отметить «галочкой» соответствующие признаки (см. выноска-1 на-рисунке 19)

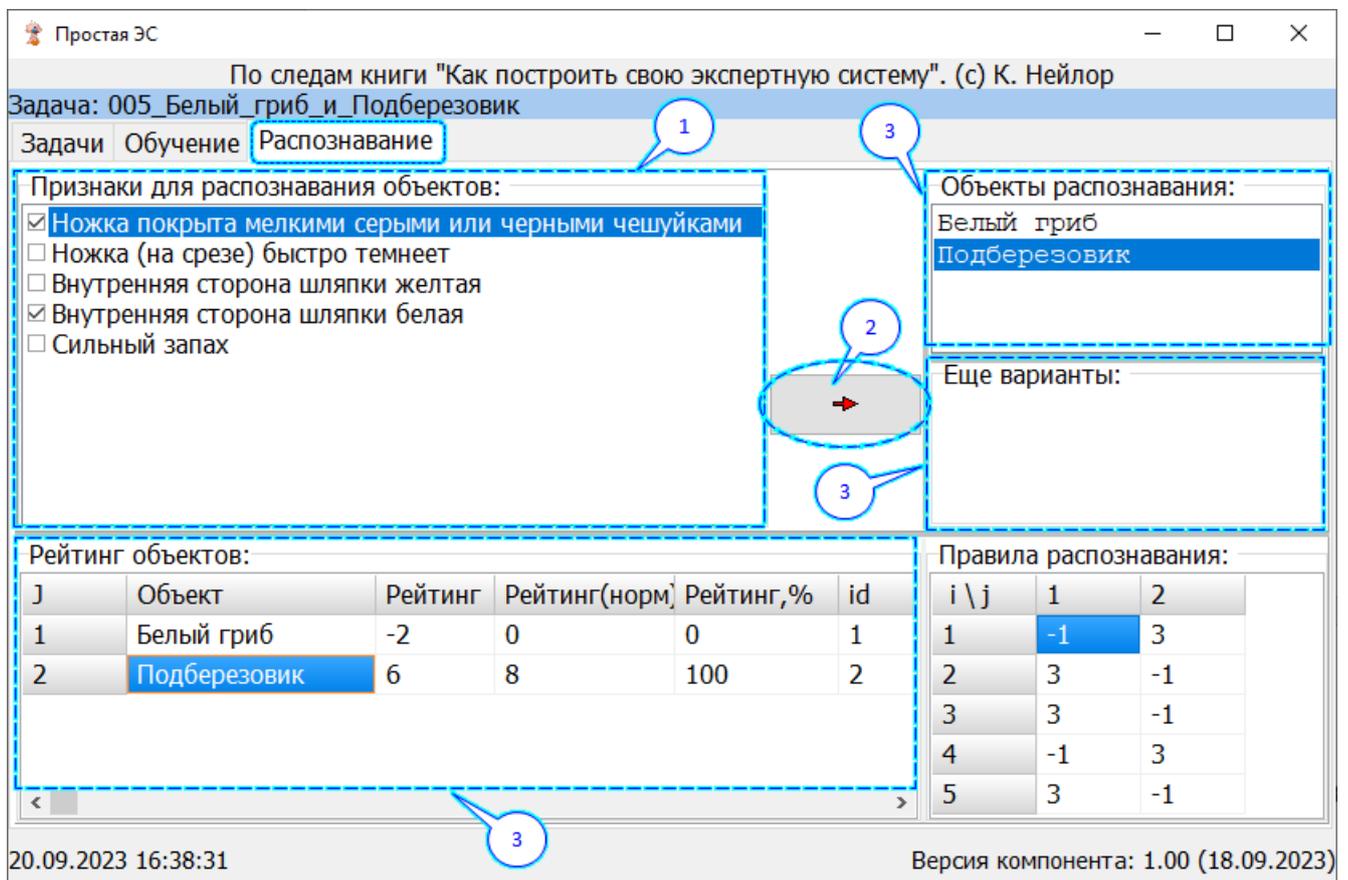


Рисунок 19 – Распознавание объектов

3. Нажать на кнопку  (см. выноска-2 на-рисунке 19).
4. Смотреть результат (см. выноска-3 на рисунке 19) Делать соответствующие выводы.

4 Используемые источники

В статье использовалась информация из книги «Как построить свою экспертную систему». Автор К. Нейлор. Москва, ЭНЕРГОАТОМИЗДАТ, 1991.