## Исходные данные

Необходимо реализовать программу на языке Delphi, выполняющую преобразование блок – схемы алгоритма, выполненной в программе Microsoft Visio, на язык Геракл. Пример алгоритма приведен на рисунке 3.

|  |
| --- |
|  |
| Рис.3. Алгоритм программы, выполненный в виде блок – схемы |

## Добавление компонента в проект

Проект был создан в Delphi 2011. Для работы с Microsoft Visio из Delphi была использована объектная модель Visio, путем создания COM – объекта класса Visio.

Для доступа к объектной модели Visio необходимо добавить в проект дополнительные компоненты. Это можно сделать, перейдя в меню в пункт Component – Import component. В появившемся меню следует выбрать пункт “Import a type library” (рис. 5).

|  |
| --- |
|  |
| Рис. 5. Меню добавления компонента. |

В появившемся меню следует выбрать “Microsoft Visio type library” и нажать Next.

|  |
| --- |
|  |
| Рис. 6. Выбор компонента из списка |

В следующем меню можно выбрать папку, в которой будет выполняться поиск подключаемых модулей. Укажем здесь папку проекта.

|  |
| --- |
|  |
| Рис. 7. Выбор директории |

В последнем меню укажем добавить новый модуль в проект.

|  |
| --- |
|  |
| Рис. 8. Добавление модуля в проект |

В папке с проектом будет сгенерирован новый файл Visio\_TLB.pas, который будет содержать необходимые классы для работы с Visio.

## Подключение компонентов

Компонент был добавлен, но еще не используется программой. Перейдем к директиве uses файла проекта, и добавим в него Visio\_TLB – обьектная модель Visio, COMOBJ – модуль для работы с COM обьектами, и Generics.Collections – стандартные коллекции, которые будут использованы при написании программы. Конечный список импортируемых модулей должен выглядеть примерно так (рис. 9):

|  |
| --- |
|  |
| Рис. 9. Список подключаемых модулей |
|  |

## Реализация программы

При написании программы были реализованы функции, выполняющие считывание информации из файла Visio, и выполняющие их преобразование на язык Геракл. В основном они работают с объектами класса Shape из объектной модели Visio. Класс Shape представляет собой одну из фигур документа Visio – процесс, решение, подпроцесс, или соединительная стрелка между фигурами. За тип фигуры в объектной модели Visio отвечает свойство NameU которое может иметь значения 'Terminator', 'Process', 'Decision', 'Dynamic connector', 'Predefined process', 'Annotation'.

Также важными свойствами класса Shape являются:

1. Index – порядковый уникальный номер фигуры на странице Visio;
2. Connects – массив, содержащий фигуры, которые соединены стрелкой. Он доступен только для фигуры – стрелки, 'Dynamic connector';
3. Text – комментарий, записанный для фигуры;
4. CellsU – массив, из которого можно получить доступ к внутренним параметрам фигуры, недоступным из объектной модели напрямую. Был использован для определения координаты фигуры CellsU['PinX'].ResultIU.

## Пример работы программы

В качестве исходных данных выступает документ visio.vsd расположенный в корне диска D. Он содержит блок – схему, которую необходимо преобразовать на язык Геракл.

|  |
| --- |
| E:\Акунович\kourse\1.PNG |
| Рис. 12. Исходные данные |
|  |
|  |

По нажатию кнопки в программе выполняется запуск данного документа в Microsoft Visio, из него извлекаются нужные данные, выполняется преобразование, после чего документ Visio закрывается, а программа заполняется текстом на языке Геракл, полученным из блок – схемы.

|  |
| --- |
| E:\Акунович\kourse\4.PNG |
| Рис. 13. Результат выполнения программы |
|  |
|  |

|  |
| --- |
|  |

# 