

Графическое программирование только одной схемой (дугой) без операторов типа goto, if, for,... без меток и скобок типа begin-end, {-} или 5.6, 55.2, 15.4 преимуществ

Для всех существующих языков программирования предлагаются два способа записи любой программы в единственной для всех графической оболочке нового поколения [1–5], рис. 1, 2.

- TDelimitedReader // Полная реализация класса TDelimitedReader, который предназначен для чтения текстового файла в формате "delimited text" (файл в таком формате представляет собой последовательность строк; каждая строка состоит из значений, которые отделены друг от друга символом-разделителем) [6 стр.17-19].

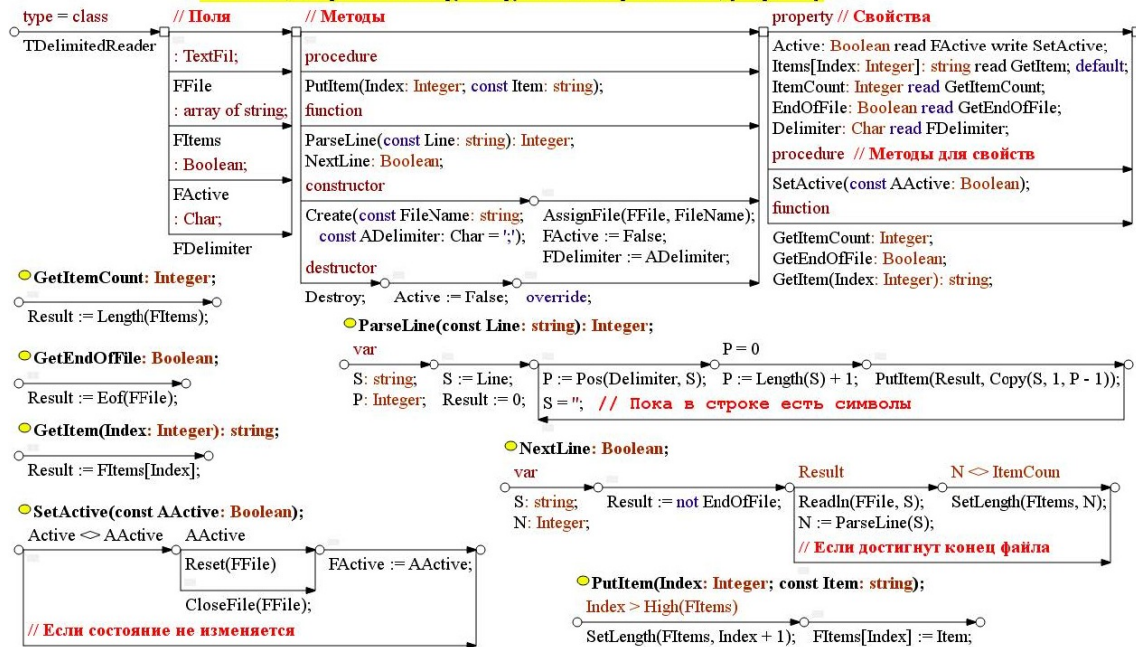


Рис. 1. Графическая программа *компактнее в 5.6 раз* и несравненно *нагляднее*, чем запись ее традиционным способом [6, стр.17-19]

В графическом программировании с помощью только одной схемы (дуги) впервые можно изобразить структуру (идею, архитектуру) программного проекта, без деталей реализации, как бы с «высоты птичьего полета» на весь проект, рис. 2. Это облегчает разработку, обсуждение, защиту, демонстрацию и обслуживание проекта. Переход от первой записи ко второй и обратно осуществляется одной функциональной клавишей.

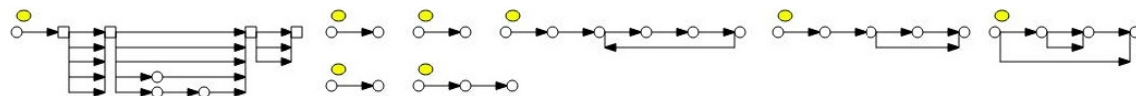


Рис. 2. Структура той же программы без деталей реализации *компактнее в 55.2 раза*

Такая запись программы, кроме перечисленных достоинств, впервые позволяет начать разработку «от идеи, архитектуры, эскиза» проекта, постепенно наращивая и развивая такую основу (скелет) деталями и возможностями реализации. Работа может проводиться в любой концепции (ООП, КОП и т.д.) и любом языке, из которого используются только простейшие, линейные установившиеся в классической математике объекты и понятия типа функций, процедур, выражений, формул, структур данных и т.д. Ввод такой программы в машину осуществляется *быстрее в 15.4 раза*. Для ввода программы рис. 2, например, требуется всего

37 (число горизонтальных дуг) нажатий кнопок мыши.

Графическая оболочка нового поколения состоит из схем. Каждая схема имеет имя, которое записывается около ее желтого эллипса, рис. 1. Элементарная схема (называется Р-схемой), из которой строится любая схема (программа) на любом языке, состоит из одной горизонтальной дуги между двумя вершинами, рис. 3, 1–2. Дуга имеет любое направление (вправо, влево, без направления). На дуге сверху пишется условие прохождения по дуге, а снизу выполняемые при этом действия. Из одной вершины могут выходить любое число дуг влево и/или вправо, рис. 4, 1–2. Любая схема строится из элементарных Р-схем и используется на всем жизненном цикле программы и программиста.

Таким образом, главное зло в существующем программировании - это символьные операторы типа *goto*, *if*, *case*, *for*. Для сегодняшнего дня они слишком 1) ориентированы на компьютер (на задание команд ему) чем на человека (на обеспечение удобства и качества процесса его мышления при реализации своей конкретной задачи); 2) они примитивны по своим возможностям, 3) имеют сложный синтаксис, 4) используют массу лишних знаков-паразитов типа «;», метки, ключевые слова *then*, *else*, *do*, *continue*,... скобки типа *begin-end*, *{-}*, 5) не позволяют документировать мысли, мотивацию действий программиста при разработке своей программы для чего стало необходимо использовать при разработке второй, дополнительный язык диаграмм UML и наконец, 6) требуют переформулировки под себя условия задачи, которую надо запрограммировать. Поэтому они усложняют и запутывают в общем-то простой процесс программирования, создают ситуацию, когда «за деревьями леса не видно», не видна идея алгоритма и программы. Этого всего нет в графическом программировании, которое ближе к классической математике, нагляднее, проще, мощнее и компактнее в 5-50 и более раз по сравнению с традиционной записью программ. Новая концепция программирования позволяет доказывать правильность программ [5], объединять программистов и делать их язык программирования со свободным синтаксисом, строить самообучающиеся системы, которые накапливают опыт своих программистов, обучать программированию с 5-7 лет и многое еще, что начинается словами «впервые» и «нет» в традиционном программировании. *Сейчас это только вершина айсберга преимуществ графического программирования только одной Р-схемой (одной специальной дугой).*

На сайте Фонда Глушкова www.glushkov.org приведена некоммерческая реализация системы для C++. Программисты-разработчики: Ходаковский Антон и Губов Александр.

Литература. 1. В.М.Глушков, И.В.Вельбицкий, Технология программирования и проблемы ее автоматизации, УСИМ, Киев, №6, 1976, с.75-93. 2. И.В.Вельбицкий, Технология программирования, Техника, Киев, Украина, 1984, 279с. 3. INTERNATIONAL STANDART ISO/IEC 8631. Information technology-Program constructs and convention for their Representation – Second edition 1989.08.01 Geneve 20, Switzerland, p.7, 1989 4. McHenry William K. R-Technology: A Soviet Visual Programming, Journal of Visual Languages and Computing, vol.1, #2, p.199-212, 1990 5. I.V. Velbitskiy, Graphical Programming and Program Correctness Proof. (plenary report), 9th IEEE COMPUTER SCIENCE & INFORMATION TECHNOLOGIES CONFERENCE, Armenia, Yerevan, 23-27 Sept.2013, IEEE Xplore, CSIT-2013. 6. А.Н. Вальвачев и др. “Программирование на языке Delphi”, ИНТЕРНЕТ, Глава 3 ООП, 19с. 2005



Рис. 3. Элементарная схема (одна дуга) вправо, влево и без стрелки, для записи любой программы в любом языке программирования

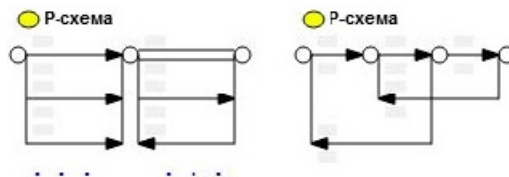


Рис. 4. Примеры записи условий без ограничений, циклов типа «ромашка» и «олимпийские кольца», которых нет в традиционных языках программирования

Сведения об авторах

Вельбицкий Игорь Вячеславович

доктор физ.-мат. наук, профессор, Лауреат Государственной премии Украины, Председатель Фонда Глушкова

Научные интересы: *Технология программирования.*

Веб-страницы автора:

- <http://www.glushkov.org>