



ПРОГРАММИРОВАНИЕ ДЛЯ ГУМАНИТАРИЕВ

D.Sc. Кузнецов Б.П., D.Sc. В.В. Сологубов

Существует мнение, что результативность подготовки специалистов гуманитарных профессий оказывается в современных условиях значительно меньшей по сравнению с подготовкой в ВУЗах инженеров различного профиля. Частично этот факт можно объяснить особым вниманием к выработке мышления в технических ВУЗах. Однако, по нашему мнению, не только гуманитарии, но и специалисты технического профиля, в том числе и программисты, не владеют таким необходимым навыком, как структурирование информации. Подобной дисциплины нет ни в одном учебном заведении, в результате чего большинство выпускников ВУЗов даже с солидным стажем работы по специальности, вплоть до ухода на пенсию, так и не могут эффективно работать с научно-технической литературой и максимально полно выразить свои мысли в виде текста, будь то технический отчет, описание работы изделия, методическое руководство и так далее, не говоря уже о написании статей, заявок на изобретения и тем более диссертаций.

Мысль о необходимости преподавания во всех без исключения ВУЗах дисциплины «Структурирование информации» возникла у авторов данной статьи в МУФО при их попытке найти общий язык с иностранцем арабского происхождения, медиком по образованию, и по счастливой случайности знакомого с основами программирования компьютеров. Сопоставим наши понятия на уровне элементарных формализмов программирования, далее мы объяснялись на некоем общем образно-формальном языке (фрагмент одной из бесед представлен в приложении), постепенно осваивали незнакомый язык и практически могли «разговаривать» на любые темы. Так возник замысел общедоступного образного языка программирования, который может быть освоен без особых склонностей к точным наукам и уже в раннем школьном возрасте. По нашему мнению, представляемый ниже язык является основных компонентной дисциплины «Структурирование информации».

Предлагаемый «Табличный язык программирования» (язык ТП) с одной стороны включает базовые конструктивны [1] практически всех языков процедурного программирования, не используя при этом ключевых текстовых слов и специальных буквенных синтаксических элементов, присущих всем известным текстовым языкам программирования. С другой стороны язык ТП оперирует в основном таблицами, которые более близки к образному представлению информации по сравнению с монотонным текстом и, следовательно, легче воспринимается большинством. Дополнительно снабдив язык ТП возможностями цифрового кодирования [2,3], мы получаем возможность компактного, наглядного и понятного представления весьма обширных процессов, не только вычислительного характера.

Рассмотрим неформально основные базовые конструктивны языка ТП.

Основной синтаксической конструкцией языка будем считать так называемый «атом», которым может быть:

- число,
- математический символ,
- текстовая фраза (слова, буква) на выбранном языке,
- произвольный графический образ.

Атомы могут быть связаны в «выражение» через знаки известных математических операций и другие специальные знаки (присущие той или иной научной дисциплине), набор которых в принципе не ограничен и здесь не обсуждается.

Действия над выражениями и атомами осуществляется в языке ТП посредством «блоков» и «таблиц».

Блоком будем называть прямоугольную область, ограниченную цветной рамкой, внутри которой могут быть заданы блоки меньшего размера, таблицы, выражения и атомы. (В последнем случае блок представляет собой задание структуры данных – программно обрабатываемой единицей информации).

Набор блоков, размещённых по горизонтали и вертикали, будем называть программой. Блоки, размещённые по одной горизонтали, считаются независимыми и выполняются параллельно во времени. Блоки, размещённые по вертикали, образуют последовательность и выполняются строго в порядке их следования сверху вниз. («Последовательность» является первым базовым конструктивным всех процедурных языков программирования.) При этом результаты выполнения некоторого блока используются как исходная информация в ниже размещённых блоках. Блок считается вырожденным, если содержит единственное выражение, и такой блок может не иметь рамки.

«Таблицей выбора» будем считать таблицу из трёх столбцов с одной или несколькими строками, снабжённую заголовком. В качестве заголовка, который размещается непосредственно над средним столбцом таблицы выбора, указывается атом или выражение, которые могут принимать более одного значения. В среднем столбце таблицы выбора перечисляются друг за другом (в каждой строке по одному) те значения, которые может принимать заголовок этой таблицы этой таблицы. В левом столбце в каждой строке указывается знак отношения заголовка и его значения, указанного в той же строке, то есть знаки: «равно», «больше», «меньше», «больше или равно», «меньше или равно» (или их текстовые или знаковые аналоги типа «выше», «ниже», «тяжелее» - «легче» и проч.). В каждой строке правого столбца задаётся блок, который следует реализовать при выполнении отношения «заголовок» - «знак отношения» - «значение заголовка». Как правило, блоки в правом столбце являются вырожденными.

Таблица выбора в языке ТП заменят так называемые условные операторы: как операторы двоичного выбора (ЕСЛИ-ТО, ЕСЛИ-ТО-ИНАЧЕ), так и операторы множественного выбора (например, оператор SWITCH-CASE языка С и др.), а также расширяют возможности выбора за счёт сравнения результатов вычислений выражений, ранее недоступные в известных языках программирования. При этом исчезает необходимость в так называемых ключевых словах и операторных скобках типа begin-end.

Третьим ключевым конструктивным любого процедурного языка программирования является «Цикл». В языке ТП ему соответствует «Блок повторений». Он отличается от обычного блока тем, что последовательность действий, перечисленных в блоке повторяется многократно: либо заданное число раз, либо до тех пор, пока выполняется заданная совокупность некоторых условий (одно условие). Блок повторений также задаётся прямоугольной рамкой, как и простой блок, со следующими отличиями. На левой стороне рамки указывается стрелка вверх, в зависимости от вида цикла на верхней или нижней стороне рамки блока указывается «Заголовок цикла». Для простоты будем различать три вида циклов: счётный, с предусловием и с постусловием повтора. Счётный цикл – это блок, который повторяется N раз, где N – число повторений, указываемое в заголовке цикла (подробности опускаем) на верхней стороне рамки блока повторений. В цикле с предусловием заголовок на верхней стороне рамки блока повторений. В цикле с предусловием заголовок на верхней рамке блока повторений содержит таблицу выбора с единственной строкой с отсутствующим правым столбцом. Блок повторений выполняется первый раз и ещё несколько раз только при условии

выполнения отношения, указанного в заголовке. В цикле с постусловием аналогичный заголовок размещён в нижней рамке блока повторений и показывает, что блок, выполненный один раз повторяется при выполнении условия (отношения), заданного в его заголовке.

Простой блок, блок повторений и любая клетка правого столбца таблицы выбора допускают произвольную вложенность других блоков и таблиц выбора.

Ещё одной разновидностью блока служит подпрограмма (процедура, функция), отличающаяся заголовком, в котором указываются входные, выходные и модифицируемые параметры. Однако в данной статье ограничимся рассмотренным.

Полное описание языка ТП требует отдельной публикации, как в виде спецификации, так и в виде руководства пользователю с примерами.

Язык ТП может быть использован в двух ипостасях: как новый компьютерный формальный язык программирования, требующий создания соответствующего программного продукта – компилятора со средой программирования, и как формализм, служащий платформой для описания процессов, процедур и различной деятельности.

По нашему мнению, язык необходимо опубликовать в полном объёме для использования и в преподавании и в любой деятельности специалистами разнообразного профиля, включая и гуманитарный.

ЛИТЕРАТУРА

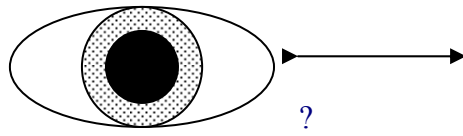
1. Вольфенгаген В.Э. Конструкции языков программирования. М., 2001.
2. Агеев В.Н. Семиотика. М., 2002.
3. Кузнецов Б.П. Цветное программирование // Вестник МАИСУ, октябрь. 2005

ПРИЛОЖЕНИЕ. Фрагмент беседы на языке, послужившем основной для разработки языка табличного программирования.

MEDICIN

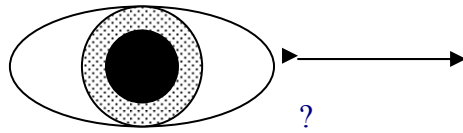
If($t^0 \wedge$)

then begin



end

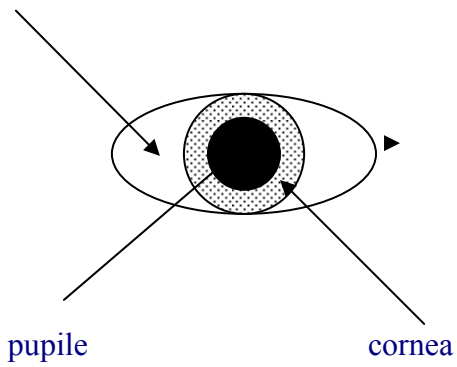
else begin



end

...

sclera



if ($t^0 > 38^0$)

then begin

sclera.color = Red

end

ПРИЛОЖЕНИЕ 2. Пример программы «Разбор предложения» (условно-упрощённо)

Разбор предложения

