

89-4

5400-1

ОБЛАСТНОЙ ИНСТИТУТ УСОВЕРШЕНСТВОВАНИЯ УЧИТЕЛЕЙ

УРОКИ ОСНОВ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ "РАПИРА"  
ДЛЯ ПЭВМ "АГАТ"

Методические рекомендации для учителей средних школ и СПТУ

Часть I

Куйбышев 1988

Курс "Основы программирования" предназначен для знакомства с основными концепциями программирования, алгоритмами и их записями на языке программирования. Курс рассчитан на учащихся, знакомых с понятием алгоритма и программы, а также с устройством ЭВМ и представлением информации в ней.

В пособии представлено 10 уроков, знакомящих с правилами программирования на Рапире и навыками работы на ПЭВМ "Агат" с системой "Школьница".

Рапира является диалоговым языком высокого уровня, предназначенным для обучения программированию и для решения на ЭВМ небольших прикладных задач. Язык ориентирован на школьников среднего и старшего возраста (II-I7 лет) и обладает следующими достоинствами по сравнению с другими алгоритмическими языками:

- современной методологией программного обеспечения персональных ЭВМ;
- русской лексикой;
- близостью к алгоритмическому языку, изучаемому в школьном курсе информатики;
- возможностью работы с цветной графикой и звуком.

Язык Рапира входит в систему "Школьница" и является ее основой.

Система "Школьница" представляет собой пакет прикладных программ для применения в учебном процессе на ПЭВМ "Агат". Кроме интерпретатора языка "Рапира" в систему входят:

- диалоговый экранный редактор программных текстов;
- дисковая операционная система;
- графическая система "Шпага";
- программа начальной разметки дисков пользователя;
- комплект тригонометрических и математических функций.

Система позволяет:

- ознакомить учащихся с возможностями ЭВМ, сформировать у них важнейшие навыки программирования и применения вычислительных машин;
- предложить для каждого учащегося индивидуальную возможность работы, позволяющую ему решать на ЭВМ задачи по различным дисциплинам самым удобным для него способом;
- повысить качество преподавания различных дисциплин, используя информационные, моделирующие, логические, демонстрацион-

Составитель - преподаватель информатики и вычислительной техники  
УИК Ленинского района О.В.Стопневич

Редактор - методист кабинета информатики и вычислительной техники  
облМУУ А.Г.Дипенкина

Рецензенты - старший преподаватель кафедры информатики и вычислительной техники Н.М.Марголис

- учитель основ информатики и вычислительной техники  
средней школы № 63 А.М.Иванов

ные возможности ПЭВМ;

- обеспечить возможность оперативного решения небольших задач на ЭВМ;
- организовать автоматизированный контроль знаний и умений учащихся.

Язык "Рапира" разработан в 1985 году в БЦ СО АН СССР и совершенствуется по настоящее время.

Настоящее пособие ориентировано на 2-ю версию языка.

## У Р О К I

ТЕМА: ПЭВМ "АГАТ". СИСТЕМА "ШКОЛЬНИЦА".

Цель:

- познакомить учащихся с ПЭВМ "Агат" (дать технические характеристики, показать основные блоки);
- дать понятие операционной системы;
- рассказать о назначении системы "Школьница";
- показать загрузку системы.

### Теоретические сведения

#### ПЭВМ "АГАТ"

Представляет собой персональную 8-разрядную ЭВМ с объемом оперативной памяти (ОЗУ) 64 Кбайт. Дополнена встроенным дисководом для гибких магнитных дисков диаметром 133 мм, цветным телевизионным монитором, стандартной клавиатурой. Для расширения возможностей системы могут быть дополнительно подключены:

- игровой блок (комплект потенциометров);
- печатающее устройство;
- второй дисковод.

#### Система "Школьница"

Для работы по ПЭВМ "Агат" сначала необходимо ввести в оперативную память ПЭВМ систему "Школьница", т.е. загрузить систему. Для этого нужно вставить диск с записанной на нем системой в дисковод и включить ПЭВМ. При этом на экране дисплея появится титульный лист системы и общее меню:

- Р - Рапира-интерпретатор
- О - автономный отладчик
- Ф - разметка рабочих дисков
- Что?

Для работы на Рапире достаточно ответить одной буквой - Р и система загрузит с диска интерпретатор языка.

В случае успешной загрузки в верхней строке появляется надпись

РАПИРА-АГАТ 1.2,

где 1.2 - номер версии, а в нижнем левом углу знак # и курсор (мигающий подчерк):



# - , что означает приглашение к диалогу.

При загрузке возможны следующие сообщения:

1) "Ошибка обмена с ДЗУ", т.е. информация не считывается с диска.

Причины:

- а) незакрытый дисковод;
- б) отсутствие диска в дисководе;
- в) порча информации или самого диска;
- г) попытка считывания с неразмеченного диска.

Результат: невозможность дальнейшей работы, необходим перезапуск системы.

2) "Программа испорчена. Продолжать л/н?".

Причина: испорчена информация.

Результат: в случае положительного ответа загрузка продолжается, но система не несет ответственности за правильность работы, в противном случае необходим перезапуск системы.

3) "Системный сбой".

Причина: внутренний сбой ЭВМ.

Результат: дальнейшая работа невозможна, необходим перезапуск системы.

После загрузки система находится в основном диалоговом режиме. Он характеризуется немедленным исполнением набранных команд (предписаний). В этом режиме удобно решать небольшие разовые задачи.

Для больших, многократно исполняемых задач существует режим запоминания, где задача оформляется в виде процедуры и может храниться на диске.

#### ПРАВИЛА ЗАГРУЗКИ

При загрузке необходимо соблюдать следующую последовательность действий:

- 1) включить монитор (телевизор);
- 2) вставить гибкий диск в дисковод;
- 3) закрыть крышку дисковода;
- 4) включить ПЭВМ.

Во избежание порчи диска не выключать ПЭВМ со вставленным в нее диском: сначала вынуть диск, затем выключить ПЭВМ.

#### Методические рекомендации

Знакомство с ПЭВМ "Агат" следует начать непосредственно в дисплейном классе. При этом предполагается, что учащиеся уже получили сведения об устройстве, назначении и представлении информации в ЭВМ.

Рассказав о техническом устройстве Агата и показав основные блоки, следует объяснить учащимся, что для работы на ЭВМ необходимо загрузить в ЭВМ операционную систему, рассказать о назначении операционной системы "Школьница" и ее составе. Показать загрузку. Привести примеры возможных сообщений при загрузке и реакции ЭВМ на них.

Рассказать о наличии двух режимов работы, объяснив их отличия.

Предложить учащимся самостоятельно загрузить ЭВМ, проследив за правильностью загрузки.

Рассказать о правилах работы на ПЭВМ, обратив внимание на технику безопасности.



У Р О К 2

ТЕМА ГРАФИКА ПЭВМ "АГАТ"

Цель:

- познакомить учащихся с графическими возможностями ЭВМ;
- дать возможность поработать в графическом режиме.

Теоретические сведения

ПЭВМ "Агат" обладает большими графическими возможностями, которые значительно расширяют класс решаемых с ее помощью задач, позволяют повысить наглядность при программировании моделей реального мира.

Большой удачей является наличие графического комплекта "ШПАГА" (Школьный ПакеТ Адаптированных Графических Средств), ориентированного на обучение школьников работе с машинной графикой.

Наличие графических процедур позволяет чертить различные фигуры и рисунки, раскрашивать их в 8 различных цветов.

Экран в графическом режиме рассматривается как часть координатной плоскости с началом координат в левом нижнем углу экрана.

В зависимости от графического режима экран разбивается на клетки разного размера. Существует 3 режима. Из них 2 цветных: ГСР (графика среднего разрешения) - экран состоит из 128x128 клеток и ГНР (графика низкого разрешения) - 64x64 клетки.

Рисовать можно только закрашивая клетку целиком в тот или иной цвет, т.е. любое изображение можно представить на экране лишь приближенно.

Текущий цвет задается процедурой ЦВЕТ (N), где N - номер цвета.

Таблица цветов

0 - черный	4 - синий
1 - красный	5 - фиолетовый
2 - зеленый	6 - голубой
3 - желтый	7 - белый

Для управления графикой в Рапире предусмотрена специальная процедура РЕМ, имеющая собственный язык команд и графические процедуры, позволяющие рисовать геометрические примитивы: линию, прямоугольник и т.д.

Подробно о графическом режиме см. [3].

Для начального знакомства с графикой рекомендуем воспользоваться модулем "Графика", который сводит управление графическим режимом на уровень экранного графического Редактора. Для загрузки модуля с диска достаточно набрать

ВКЛ МОДУЛЬ ГРАФИКА

На экране появляется следующая таблица.

Система команд

Клавиши	!	Действия
Стрелочки		Передвижения курсора
К		Конец работы. Вывод на экран и в файл текста со - заданной программы
?		Выводится система команд
I-8		Установка номера текущего цвета
Ф		Закраска экрана текущим цветом
Н		Отметка начала при рисовании линии или прямоуголь - ников
Т		В позиции курсора ставится точка
О		Выполняется процедура ОБЛ (x,y), где x,y - коорди - наты курсора
Л		Рисуется линия от позиции и курсора до точки, от - меченной командой Н
Р		Смена режима, по этой команде следующая команда Л или П начинается в конце следующей линии или пря - моугольника. Для выхода нужно нажать Р во второй раз
С		После нажатия за курсором начинает оставаться след текущего цвета. Для выхода написать С во второй раз
У		Установка курсора в указанную позицию. Координаты указываются на запросы программы
П		Рисуется прямоугольник с диагональю от позиции кур - сора до отмеченной точки

Кроме указанного модуля могут использоваться и другие графические процедуры системы "ШПАГА".

Методические рекомендации

Машинной графике можно уделить несколько первых уроков. Указав на элементарные средства управления экранной графикой, необходимо дать возможность детям проявить свою фантазию в составе различных рисунков. Тем самым преодолевается психологический барьер, существующий при начальной работе на ЭВМ, у учащихся появляется интерес к ЭВМ, создается положительный эмоциональный настрой.

Далее можно предложить нарисовать конкретный рисунок (например, дом) или написать слово (например, АГАТ).

Графические возможности Агата - большое достоинство, которое должно использоваться в полной мере и зависит от желания и заинтересованности преподавателя.

УРОК 3

ТЕМА: ПРЕДСТАВЛЕНИЕ ЧИСЕЛ. ВЫЧИСЛЕНИЕ ЧИСЛОВЫХ ВЫРАЖЕНИЙ

Цель - научить правильно представлять числа и вычислять простые выражения в диалоговом режиме.

Теоретические сведения

Числовые выражения записываются по правилам, принятым в большинстве языков программирования.

Арифметический знак операции	!	Знак в Рапире
Сложение +		+
Вычитание -		-
Умножение x		*
Деление:		/
Возведение в степень		**

Кроме арифметических операций над целыми числами предусмотрена операция "деление нацело", обозначаемая //. Например, 86//3. Результат 28.

Приоритеты операции и правила расстановки скобок общеприняты. Допускаются только круглые скобки.

Для вычисления числового выражения необходимо:

- 1) набрать "?";
- 2) набрать нужное выражение;
- 3) набрать ";";
- 4) перевести строку /клавиша **F** /.

Пример 1. Чтобы вычислить выражение

$$5986 \times 1928 + 961^2 - 512:256,$$

необходимо набрать

$$?5986 * 1928 + 961 ** 2 - 512 / 256.$$

На экране дисплея появится результат: 12464527.

Для записи дробных чисел используются 2 вида записи:

- а) с десятичной точкой;
- б) с плавающей точкой /экспоненциальная форма/ - для очень больших или очень малых чисел.

Пример 2.

- а) 12,4I записывается 12.4I;
- б) 10,15x10<sup>24</sup> записывается 10.15E24;
- 5,6x10<sup>-18</sup> записывается 5.6E-18.

Диапазон целых чисел |M| < 2x10,16, дробных - 10x128 /x/ < 10x128.

Точность представления дробных чисел - 12 значащих цифр.

Для вычисления нескольких выражений их можно указать в одной строке, разделяя запятой.

Пример.3.

?5x8, 3+4, 80/5; √.

На экране дисплея появится результат: 40 7 16.

Выражение может занимать несколько экранных строчек. Пробелы между числами и знаками не влияют на запись. Внутри чисел пробелы не допускаются.

### Методические рекомендации

Вначале необходимо ознакомить с правилами и научить учащихся записывать числовые выражения на Рапире. Для этого можно предложить несколько упражнений.

Упражнение I. Записать следующие числовые выражения на Рапире:

- 1.  $\frac{18,6 \cdot 0,021 - 0,0018^2}{16}$  (18.6x0.21-0.0018x2)/16
- 2.  $\frac{3,8^3}{15,68} + \frac{18,3}{3,26}$  3.8x3/15.68+18.3/3.26
- 3.  $\frac{0,004 \cdot 0,271}{5,36} - 0,00023$  0.004x0.271/5.36-0.23E-3
- 4.  $\frac{137,85+81,97}{15,4^2-6,1^3}$  (137.85+81.97)/(15.4x2-6.1x3)

Упражнение 2. По записи на Рапире восстановить арифметическое выражение.

- 1. -1,1/(2,3x4)+8x2/(4,67x9x3)  $\frac{-1,1}{2,3 \cdot 4} + \frac{8^2}{4,67 \cdot 9^3}$
- 2. -6,5E-4x5,12/(0,1E-5+0,015E-4)  $\frac{-0,00065 \cdot 5,12}{0,00001+0,000015}$
- 3. 2,3E4+6,5E4/5,18E5  $\frac{23000+65000}{518000}$

Затем непосредственно в дисплейном классе дать возможность самостоятельно вычислить несколько числовых выражений.

Например, 2<sup>1016</sup> ?2x1016.

Результат: "Слишком большое число".

Это занятие дает возможность убедиться в одной из главных способностей ЭВМ как быстрого вычислителя.



УРОК 4

ТЕМА: ЯЗЫК ПРОГРАММИРОВАНИЯ "РАПИРА". ОСНОВНЫЕ ПРАВИЛА.  
ПРЕДПИСАНИЕ ПРИСВАИВАНИЯ.

Цель - дать представление о языках программирования; охарактеризовать язык "Рапира", познакомить с его правилами, объяснить смысл предписания присваивания

Теоретические сведения

Основные правила языка Рапира

1. Любая программа состоит из предписаний. Каждое из них заканчивается ";".
2. Предписания состоят из отдельных слов.
3. Под словами подразумеваются: числа, тексты, имена, специальные знаки (скобки, знаки арифметических операций и т.д.) и ключевые слова (слова, входящие в состав языка "Рапира").

Объекты языка

Объекты - это те данные, которыми оперирует программа. В Рапире различаются следующие виды объектов:

- 1) целые числа;
- 2) дробные (вещественные) числа;
- 3) тексты;
- 4) множества;
- 5) кортежи;
- 6) записи;
- 7) процедуры;
- 8) функции;
- 9) файлы.

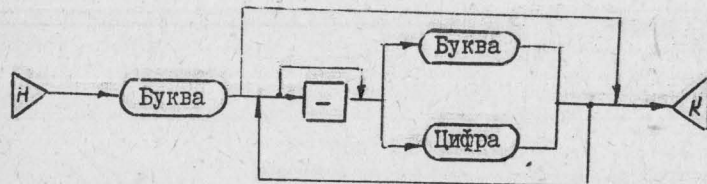
Объекты, указанные непосредственно в тексте, называются константами.

Объекты, получающиеся в результате некоторых действий над другими объектами, называются промежуточными (временными) значениями. После выполнения любой операции над таким значением оно уничтожается.

Чтобы сохранить в памяти некоторое промежуточное значение, его необходимо присвоить некоторому имени.

Имена

Состоят из букв и цифр, причем первый символ - буква. Синтаксическая диаграмма:



Например, X, A5, S I, ИМЯ УЧЕНИКА.

Правила

1. Число символов в имени может быть любым, но не более 255.
2. Совпадающие по написанию русские и латинские буквы не разделяются.
3. Используются только заглавные буквы.
4. Имя, состоящее из нескольких слов, разделяется отдельными подчеркиками.
5. Количество разных по написанию имен, используемых в программе, не должно превышать 256.

Предписание присваивания

Предписание - это указание ЭВМ совершить какое-либо действие над объектами.

Синтаксическая диаграмма:



Выражение - это последовательность соединенных знаками констант и имен;

ИМЯ - имя блока памяти, где будет храниться вычисленное выражение;

-> - знак присваивания.

Смысл предписания: вычислить выражение, записанное слева знака присваивания, и результат поместить в блок (ячейку) с данным именем.

Имена, которым не были присвоены значения в программе, имеют значение "пусто".

Предусмотрено стандартное имя ПИ, имеющее значение

3.14 I59265359.

Для просмотра текущих значений имен удобно использовать простейшую форму вывода "?".

Пример.

5 → X;  
3+X\*28417 → СКОРОСТЬ;  
X+8 → X;  
? X, СКОРОСТЬ;  
Результат: I3 I42088.

Методические рекомендации

Эта тема является введением в программирование. Здесь даются важные понятия: язык программирования, объекты языка, имена. Происходит знакомство с первым предписанием - предписанием присваивания.

Начать занятие рекомендуется с обзора языков программирования, истории создания, назначения. Дать характеристику языка "Рапира". Затем следует дать основные понятия языка и записать основные положения.

При описании синтаксиса языка рекомендуется пользоваться синтаксическими диаграммами, которые полно и наглядно представляют все языковые конструкции. Подробно о синтаксических диаграммах см. I.

При объяснении работы предписания присваивания обратить внимание на его смысл и не путать со знаком равенства в математике.

После теоретической части можно рассмотреть следующие упражнения.

Упражнение I. Вычислить объем тела снеговика, состоящего из 3 снежных шаров радиусами R1=0,75 м; R2=0,55 м; R3=0,35 м.

Составим программу, учитывая, что Vшар = 4πR³/3.

0,75 → P1; 0,55 → P2; 0,35 → P3;  
4\*PI\*P1\*\*3/3 ОБ1;  
4\*PI\*P2\*\*3/3 ОБ2;  
4\*PI\*P3\*\*3/3 ОБ3;

ОБ1+ОБ2+ОБ3 → ОБЦ;

? ОБЦ;

На дом следует задать написание программы вычисления по одной из формул геометрии (площади и объема фигур), физики, химии и т.д.

Варианты заданий

Упражнение 2. Вычислить площадь и объем усеченного конуса, если известно, то площадь S = π(R+r)l + πR² + πr²

объем V = 1/3 π(R²+r²+R\*r)\*h

Дано: R = 15,87 м; h = 8,21 м; r = 5,32 м;  
l = 10,03 м.

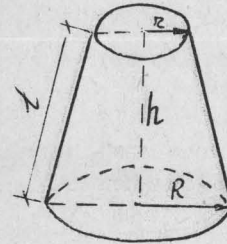
Составим программу:

15,87 → R I; 5,32 → R 2; 8,21 → H;  
10,03 → L ;

PI\*(R1+R2)\*L + PI\*(R1\*\*2 + R2\*\*2) → S ;

PI\*(R1\*\*2 + R2\*\*2 + R1+R2) → V ;

? S , V ;



Упражнение 3. Вычислить путь S при равнопеременном движении, если известно, что S = v₀t + at²/2

Дано: v₀ = 18,3 м/сек; t = 60 сек; a = -0,3 м/сек².

Составим программу:

18,3 → V ; 60 → T; -0,3 → A;

V\*T + A\*T\*\*2/2 → S ;

? S ;

Примечание. При задании имен, учитывая русскую лексику языка, предпочтительнее пользоваться русским алфавитом.



ТЕМА: ИСПОЛНЕНИЕ ПРОСТЕЙШИХ ЛИНЕЙНЫХ ПРОГРАММ НА ЭВМ. ДИАГНОСТИКА ОШИБОК И ИХ ИСПРАВЛЕНИЕ В ДИАЛОГОВОМ РЕЖИМЕ

Цель - научить решать простейшие задачи с помощью ЭВМ, вводить составленную программу в ЭВМ и исправлять ошибки в диалоговом режиме

Теоретические сведения

Диагностика ошибок

Сообщения об ошибках, выдаваемые ЭВМ, называются диагностической ошибкой.

А. При вводе.

Если во вводимой строке содержится несколько предписаний, анализ и выполнение каждого из них начинается только после выполнения предыдущего предписания. Обнаружив ошибку в записи очередного предписания, система подает звуковой сигнал, дублирует на экране строку, в которой допущена ошибка, выделяя ошибочное место красным цветом, и выдает сообщение об ошибке. Вся оставшаяся часть введенной строки (в том числе и все последующие предписания игнорируются).

Б. При исполнении.

При обнаружении ошибки во время исполнения предписания выполняются те же действия, но строка с ошибкой не выдается.

Исправление ошибок

А. До перевода строки.

Если ошибка замечена в процессе набора, надо вернуться к этому месту с помощью клавиши ← ("стрелка назад"), исправить ошибку, а затем набрать оставшуюся часть заново или вернуться к концу строки клавишей → ("стрелка вперед"). После этого продолжить набор предписания.

Б. После перевода строки.

Если ошибка обнаружена после получения сообщения о ней, необходимо ввести предписание заново, исправив ошибку.

Если предыдущий текст еще виден на экране, достаточно подвести курсор к его началу и ввести этот текст в ЭВМ с помощью клавиши → и "ПВТ", попутно исправив ошибку.

Методические рекомендации

Это занятие проводится на ЭВМ. Учащимся предлагается ввести программы (упр. 1,2,3). При возникновении ошибок объясняются правила их исправления.

Для демонстрации работы предписания присваивания предложить учащимся сразу после присваивания конкретных значений блокам делать проверку содержимого блоков с помощью "?".

Например (упр.1):

0,75 → P1; 0,55 → P2; 0,35 → P3;

?P1, P2, P3;

Результат:

0,75 0,55 0,35.

Преподаватель объясняет назначение функциональных клавиш, помогает учащимся правильно ввести и исполнить программу. При успешном исполнении программ можно дать дополнительное задание. Необходимо добиваться от учащихся получения правильного результата, указав на трудоемкость и длительность процесса отладки.

Можно предложить учащимся упражнения по освоению клавиатуры ЭВМ.



У Р О К 6

ТЕМА: ТЕКСТЫ И ОПЕРАЦИИ НАД НИМИ

Цель - познакомить с другим объектом языка - текстами, показать способы обработки текстов.

Теоретические сведения

Текст - это произвольная последовательность символов. При записи в программе тексты заключаются в кавычки.

Пример 1.

"ТЕКСТ", "РАПИРА", "ЭВМ "АГАТ"", "123+45".

Все символы текста, включая пробелы, занумерованы слева направо, начиная с единицы. Например: "НУМЕРАЦИЯ"  
123456789.

Тексты могут входить в состав выражений и над ними могут выполняться следующие операции:

1. Определение мощности ~~//~~.

Мощностью текста называется количество символов в нем.

Пример 2.

"КРОКОДИЛ" → Т;

? ~~//~~ Т;

Результат: 8.

2. Конкатенация (слияние) + соединяет два текста в один.

Пример 3.

"ВЕЛО"+"СИПЕД" → Т;

?Т;

Результат: ВЕЛОСИПЕД.

3. Выборка - извлечение одного символа по его номеру (индексу) в тексте. В качестве индекса используется любое выражение с целым положительным значением. Индекс заключается в квадратные скобки.

Пример 3:

а) 9"ДИНОЗАВР" [3];

Результат: Н;

б) 5 → Х; "КРОКОДИЛ" → Т; ? (Т+Т) [3\*Х];

Результат: И.

4. Вырезка - извлечение участка текста, определяемого индексами начального и конечного символов. Обозначается также, как и выборка, но в скобках два индексных выражения, разделенные двоеточием. Обязательное условие: второй индекс не меньше первого.

Пример 4:

а) А [3:4], "БЕТЕМОТ" [N-2: N+1];

б) ?"ВЕЛОСИПЕД"[5:7].

Результат: СИП.

Методические рекомендации

Это занятие расширяет представление учащихся о возможностях ЭВМ. Познакомив с операциями над текстами, можно предложить следующие упражнения.

Упражнение 1. Получить результат выполнения следующей программы:

"ПРИМУС" → Х;

Х [4:4] + Х [3] + Х [6] + Х [~~//~~X] + [~~//~~X-1] + Х [2:3] → РЕКА;  
?РЕКА;

Результат: МИССУРИ.

Упражнение 2. Получить результат выполнения следующей программы:

"КОМПЬЮТЕР" → А;

"ОРГАН" → Б;

"ДУШ" → С;

Б [~~//~~5] + Б [~~//~~Б-1] + С [~~//~~С] → Н;

С [1] + Б [2] + С [2] + Б [3] → Д;

?А, "-", Н, " ", Д, "!";

Результат: КОМПЬЮТЕР-НАШ ДРУГ!

В качестве самостоятельной или домашней работы можно предложить следующее упражнение.

Упражнение 3. Составить новые слова из слова "КОМПЬЮТЕР" путем различных операций.

В дисплейном классе предложить ввести и выполнить вышеуказанные упражнения. Предложить самостоятельно придумать задачи и выполнить.

## У Р О К 7

### ТЕМА: РЕЖИМ ЗАПОМИНАНИЯ. ОФОРМЛЕНИЕ, ВЫЗОВ И РЕДАКТИРОВАНИЕ ПРОЦЕДУРЫ

#### Цель:

- показать наличие режима редактирования и его отличие от диалогового режима;
- дать понятие процедуры, показать ее оформление и вызов;
- научить исправлению ошибок в процедуре.

#### Теоретические сведения

В ЭВМ "Агат", кроме основного диалогового режима, который характеризуется немедленным исполнением предписания, существует режим запоминания, при котором предписания не исполняются, а запоминаются в процессе набора.

Программа, записанная в режиме запоминания на Рапире, называется процедурой. Процедуру можно изменять (т.е. редактировать), записывать на диск, вызывать с диска. Режим запоминания называется еще режимом редактирования. Компонент системы, организующий работу в режиме редактирования носит название Редактор.

Для того чтобы перейти в режим запоминания, достаточно оформить любую программу как процедуру. Для этого перед первым предписанием необходимо набрать директиву описания:

ПРОЦ имя программы;

где имя программы - название программы.

При этом включается дисковод, а на экране зелеными символами выдается:

1 ПРОЦ имя программы;

2

Если работа ведется без записи на диск, то внизу выдается ОШИБКА ОБМЕНА С ДЗУ.

Если предполагается запись на диск и рабочий диск находится в дисковом, то выдается сообщение

ФАЙЛ НЕ НАЙДЕН (т.е. имя процедуры уникально).

Для продолжения работы достаточно нажать красную клавишу, и курсор перемещается на вторую строчку. Это означает, что работающий находится в режиме редактирования и система готова к за-

поминанию процедуры. Набираемый текст не анализируется, а только запоминается в ОЗУ.

Отличие этого режима от основного диалогового в следующем.

1. Экран рассматривается как окно, через которое мы видим некоторую часть текста.
2. Текст выдается на экран зеленым цветом и разбит на строки, имеющие номер.
3. Всякое изменение текста на экране влечет изменение текста в памяти.
4. Движение по тексту осуществляется с помощью 4 клавиш со стрелками.
5. Для образования новой строки текста после текущей служит клавиша перевода строки.
6. Для замены текущего символа надо просто набрать другой.

#### Оформление и вызов процедуры

Для оформления процедуры и последующего, возможно неоднократного, исполнения необходимо:

1. Набрать директиву описания

ПРОЦ имя;

2. После входа в Редактор набрать текст программы.

3. В конце программы набрать КНЦ.

4. Нажать клавишу РЕД, при этом появляется меню Редактора и курсор (желтая полоска) должен находиться на строке КОНЕЦ ОПИСАНИЯ.

5. Нажать клавишу  $\sqrt{\quad}$ . При этом происходит проверка программы на синтаксическую правильность.

6. Если синтаксических ошибок нет, появляется запрос:

имя:ЗАПОМИНАТЬ (Д/Н)?,

что означает: синтаксических ошибок в программе не обнаружено и программу можно записывать на диск. Для этого нужно ответить Д(да).

Если ответить Н(нет), то программа остается в ОЗУ.

7. Для исполнения программы необходимо сделать вызов процедуры. Если она находится в ОЗУ, достаточно набрать: имя ( ).

Если она вызывается с диска, то надо набрать следующую директиву: ВВОД ИЗ ДЗУ: имя; имя ( ).

8. Если при исполнении программы выдано сообщение об ошибке или оказалось, что она неверна, можно вернуться в режим редактирования и исправить программу. Для этого необходимо опять набрать ПРОЦ имя.



### Исправление ошибок в режиме редактирования

По директиве КСНЕЦ ОПИСАНИЯ в Редакторе производится синтаксический анализ текста. Если обнаруживаются ошибки, то происходит возврат в режим редактирования и выдается сообщение об ошибке. Ошибочное место выделяется курсором (желтым прямоугольником).

В режиме редактирования предусмотрены все средства для эффективного редактирования: уничтожение символов, вставка символов и строк, склеивание и т.д. Подробно см. [3].

После исправления ошибки нужно опять нажать клавишу РЕД и выйти из Редактора по "концу описания" (нажать клавишу перевод строки). Если текст синтаксически правилен, происходит трансляция всех предписаний, т.е. создание объекта языка вида Процедура. После завершения трансляции выдается запрос:

Имя: ЗАПОМИНАТЬ (Д/Н)?

Следует ответить, нужно ли запоминать на диск отредактированный текст. Если текст не редактировался или был запомнен еще в Редакторе, данный запрос не выдается.

### Методические рекомендации

Данное занятие является практическим и проводится в дисплейном классе. Здесь необходимо показать наличие у ПЭВМ "АГАТ" второго режима запоминания и рассказать об отличии его от основного диалогового режима. Затем следует показать оформление процедуры. Для этого можно взять любую ранее разобранный программу (упражнения 1,2,3 урока 4), умышленно допустив в ней ошибки. Затем отредактировать ее и вызвать.

Можно с помощью редактирования изменить в программе исходные данные и снова вызвать процедуру.

Для усвоения работы функциональных клавиш в режиме редактирования можно предложить учащимся написание любого текста с последующим его редактированием: уничтожением и вставкой символов, склеиванием символов и строк и т.д.

На этом занятии происходит лишь знакомство с режимом запоминания, в дальнейшем предполагается работа в этом режиме и все последующие программы оформляются как процедуры.

### У Р О К 8

### ТЕМА: ПРЕДПИСАНИЯ ВВОДА И ВЫВОДА. ДИАЛОГОВЫЕ ПРОЦЕДУРЫ

Цель:

- дать понятие операций ввода-вывода;
- познакомить с предписаниями ввода и вывода и их формами;
- научить составлять простейшие диалоговые программы.

### Теоретические сведения

Предписания ввода-вывода являются неотъемлемой частью любой программы, они помогают организовать диалог человека с ЭВМ, делают программу понятной и универсальной.

### Предписание вывода

Простейшая форма предписания вывода на экран дисплея ? список элементов вывода

нами уже использовалась.

Аналогично:

ВЫВОД: список элементов вывода.

Предписание предназначено для посимвольного вывода на экран телемонитора объектов, указанных в списке вывода в естественном формате. Элементами списка вывода могут быть тексты, имена переменных, выражения.

### Пример 1.

ВЫВОД: "PI=", PI, "A=", I8.6ж6.35;

Результат: PI=3.14159265359 A=118.11.

Элементами списка вывода являются (по порядку): текст, имя переменной, текст, выражение.

Рассмотрим формы вывода различных объектов.

1. Тексты выводятся в виде цепочки символов без кавычек. Число занимаемых позиций равно мощности текста.

### Пример 2.

ВЫВОД: "ЭВМ "АГАТ"";

Результат: ЭВМ "АГАТ".

2. Целые числа выводятся как последовательность цифр с одним пробелом впереди. Перед отрицательным числом выводится знак "-".

### Пример 3.

ВЫВОД: 2жж10, 7, -456789;

Результат: 1024 7 -456789.



3. Дробные числа из диапазона 0.1; 1.0E13 (по модулю) выдаются в форме с фиксированной точкой.

Дробные числа, порядок которых отрицателен или больше 12, выдаются в нормализованной экспоненциальной форме, т.е. в виде мантиссы, лежащей в диапазоне 0.1; 1.0 и десятичного порядка.

Пример 4. Формы представления чисел:

а) с фиксированной точкой: 75.8736, -12712.0, 0.0;

б) в экспоненциальной форме: 0.2453634E-01, -0.9227E 126.

Информацию можно выводить не только на экран, но и на бумагу, диск. Для этого в предписании вывода надо указать устройство вывода.

Пример 5.

ВЫВОД НА БУМАГУ: "2x2=", 2x2;

Результат печатается на бумаге: 2x2=4.

Для вывода на диск используется следующее предписание:

ВЫВОД В ФАЙЛ имя файла;

Кроме естественного формата существуют другие формы вывода: позиционный и фиксирующие форматы. Подробно см. [3].

### Предписание вывода

Рассмотрим ввод информации с клавиатуры. Синтаксическая диаграмма



Имя - имена блоков памяти. Следует отличать данные от текстов. Текст - это любая последовательность символов, вводимая без кавычек. Вводимый текст не должен быть длиннее 255 символов. Данные - это числа, написанные по правилам Рапиры.

Выполнение этого предписания начинается с появления на экране "?" и курсора. Система переходит в режим ввода. С этого момента любая последовательность символов, набранная с клавиатуры до нажатия клавиши "перевод строки", присваивается очередному имени из списка ввода. Данные и тексты во вводимой строке разделяются пробелами, переводами строк и запятыми. Число вводимой информации должно соответствовать числу имен в списке. Если оно меньше, знак

3  
5  
5  
1

"?" будет повторяться до тех пор, пока не будет набрана недостающая информация. Если оно больше, то лишняя информация игнорируется.

Пример 6.

а) ВВОД ТЕКСТОВ: А,Б,В;

? АГАТ, -ЭТО, ЭВМ (ввод текстов с клавиатуры)

ВЫВОД: В.Б,А;

Результат: ЭВМ-ЭТО АГАТ.

Пример 7.

ВВОД ДАННЫХ: А,Б,В;

? 4,5,2 (ввод данных с клавиатуры)

ВЫВОД: "4x5/2=", АxБ/В;

Результат: 4x5/2=10.

### Методические указания

Познакомив учащихся с записью предписаний ввода и вывода, способами ввода-вывода информации, необходимо научить их грамотно и корректно составлять диалоговые программы, оформив их как процедуры.

Указать, что существует два способа занесения информации в блок: 1) путем присваивания конкретного значения данному имени с помощью предписания присваивания; 2) путем введения конкретного значения с клавиатуры с помощью предписания ввода. Второй способ делает программу универсальной и предполагает ввод различных исходных данных для одной и той же программы. Для наглядности и понятности диалога его необходимо снабдить информацией: какие данные вводить и в каком порядке. Для этого используется предписание вывода. Диалоговые программы оформляются как процедуры. Примером может служить диалоговая часть программы упражнения 2 урока 4.

Упражнение 1.

ПРОЦ КОНУС;

ВЫВОД: "ВВЕДИТЕ ЗНАЧЕНИЯ НИЖНЕГО И ВЕРХНЕГО РАДИУСОВ ,  
РАЗДЕЛЯЯ ИХ ПРОБЕЛОМ";

ВВОД ДАННЫХ:

ВЫВОД: "ВВЕДИТЕ ЗНАЧЕНИЕ ВЫСОТЫ Н И ОБРАЗУЮЩЕЙ L ,РАЗДЕЛЯЯ  
ИХ ПРОБЕЛОМ";

ВВОД ДАННЫХ: Н, L ;

и т.д.

Рекомендуется таким образом оформить несколько процедур, предоставив учащимся возможность самим составить диалоги, а затем ввести, отредактировать и выполнить процедуру с различными данными.

Упражнение 2.

1). Написать программу вычисления площади треугольника по формуле Герона:

где A, B, C - стороны треугольника;  
P - полупериметр  $P=(A+B+C)/2$ .

2) Исполнить программу для следующих значений сторон треугольника

Сторона!	Значения		
A	3.15	6.21	17.83
B	4.75	3.43	26.14
C	6.01	7.21	20.01

Формула Герона:

$$S = \sqrt{P(P-A) \cdot (P-B) \cdot (P-C)}$$

ПРОЦ ГЕРОН;

ВЫВОД: "ВВЕДИТЕ ЗНАЧЕНИЯ СТОРОН ТРЕУГОЛЬНИКА, УЧИТЫВАЯ, ЧТО

$A+B > C, A+C > B, C+B > A$ ";

ВВОД ДАННЫХ: A,B,C;

$(A+B+C)/2 \rightarrow P$ ;

$SQRT(P \cdot (P-A) \cdot (P-B) \cdot (P-C)) \rightarrow S$  ;

ВЫВОД: "ПЛОЩАДЬ ТРЕУГОЛЬНИКА =", S ;

КНЦ;

*SQRT*- стандартная функция "корень квадратный" ( $\sqrt{\quad}$ ).

У Р О К 9

ТЕМА: ВЕТВЛЕНИЕ. УСЛОВНОЕ ПРЕДПИСАНИЕ. ПРЕДПИСАНИЕ ВЫБОРА

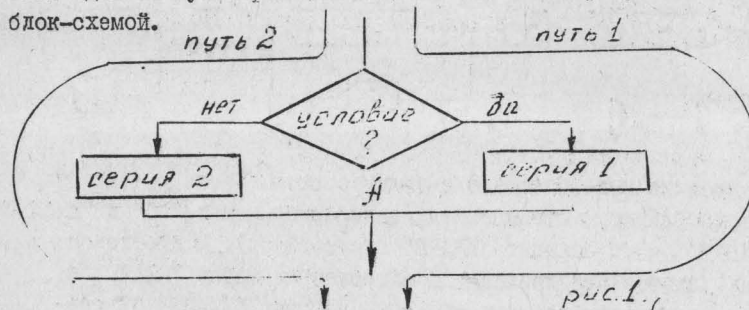
Цель:

- представить алгоритмы с ветвлением;
- познакомить с условным предписанием и предписанием выбора;
- научить составлять программы с ветвлением.

Теоретические сведения

Ветвление

В программировании часто используются алгоритмы с ветвлением, т.е. такие алгоритмы, где при проверке какого-либо условия выбирается один из путей решения задачи. Это иллюстрируется следующей блок-схемой.



При истинности условия выбирается 1-й путь следования, при ложности - 2-й. Серия 1 и серия 2 - это серии предписаний, которые следует исполнить при выбранном пути.

Условие

Условием называется некоторое утверждение об объектах, которое может быть истинным или ложным.

Пример 1.

$12 > 3$  - условие истинно;

$7 < 4$  - условие ложно.

Для составления простых условий числовых значений используются знаки:

- $>$  - больше,
- $<$  - меньше,
- $\leq$  - меньше или равно,
- $\geq$  - больше или равно,
- $=$  - равно,
- $\neq$  - неравно.







3:ВЫВОД: "УДОВЛЕТВОРИТЕЛЬНО"!

1,2:ВЫВОД: "ПЛОХО"

ИНАЧЕ ВЫВОД: "НЕПОНЯТНО"

ВСЁ;

Примечание. Здесь параметром выбора является имя блока ОЦЕНКА, его значение поочередно сравниваются с числами 5,4,3,1,2.

Методические рекомендации

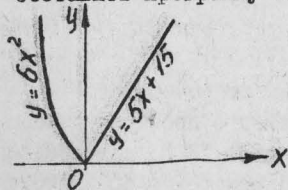
Одна из основных целей курса - формирование у учащихся логического мышления. Этому способствует составление разветвленных алгоритмов. Поэтому на это необходимо уделить значительную часть времени. Сначала необходимо научить составлять разветвленные алгоритмы, используя блок-схемы, затем познакомить с условным предписанием и показать, как записывать программы с ветвлением. Для лучшего понимания сначала лучше использовать простые условия, а затем можно перейти к составным.

Имеет смысл разобрать алгоритмы с вложенным ветвлением, показав запись таких программ сначала с условным предписанием, а затем с предписанием выбора.

После усвоения учащимися понятия ветвления и понимания работы условного предписания можно предложить им несколько упражнений на составление программ с ветвлением.

Упражнение 1.

Составить программу вычисления сложной функции:



$$y = \begin{cases} 6x^2 & \text{при } x < 0 \\ 5x + 15 & \text{при } x \geq 0 \end{cases}$$

ПРОЦ ФУНКЦИИ;

ВЫВОД: "ВВЕДИТЕ ЗНАЧЕНИЕ X";

ВВОД ДАННЫХ: X;

ЕСЛИ  $x < 0$  ТО  $6 \cdot x \cdot x \rightarrow y$

ИНАЧЕ  $5 \cdot x + 15 \rightarrow y$

ВСЁ;

ВЫВОД: "y=", y;

КНЦ;

3  
5  
5  
1

Упражнение 2.

Определить, принадлежит ли точка M с координатами X, Y кругу радиусом R (центр круга совпадает с центром координат). Составить программу.

Решение. Если расстояние OM меньше или равно R, то точка принадлежит кругу



ПРОЦ КРУГ;

ВЫВОД: "ВВЕДИТЕ КООРДИНАТЫ ТОЧКИ X, Y";

ВВОД ДАННЫХ: X, Y;

ЕСЛИ  $\sqrt{X \cdot X + Y \cdot Y} \leq R$

ТО ВЫВОД: "ТОЧКА ПРИНАДЛЕЖИТ КРУГУ"

ИНАЧЕ ВЫВОД: "ТОЧКА НЕ ПРИНАДЛЕЖИТ КРУГУ"

ВСЁ;

КНЦ;

Упражнение 3.

Составить программу вычисления функции  $y = \frac{1}{x} + \frac{1}{x-1}$ .

При  $x=0$  или при  $x=1$  выражение не имеет смысла.

ПРОЦ ДИЗ;

ВЫВОД: "ВВЕДИТЕ X";

ВВОД ДАННЫХ: X;

ЕСЛИ  $x=0$  ИЛИ  $x=1$

ТО ВЫВОД: "ЗНАЧЕНИЕ Y НЕОПРЕДЕЛЕНО"

ИНАЧЕ  $1/x + 1/(x-1) \rightarrow y$ ;

ВЫВОД: "y=", y

ВСЁ;

КНЦ;

Примечание. Имеет место логическое сложение (дизъюнкция) "ИЛИ".

Упражнение 4. Определить, можно ли из трех заданных отрезков A, B, C построить треугольник. Составить программу.

Условием существования треугольника является одновременное выполнение трех неравенств:  $A+B > C$ ,  $A+C > B$ ,  $B+C > A$ .

ПРОЦ ТРЕУГОЛЬНИК;  
 ВЫВОД: "ВВЕДИТЕ ЗНАЧЕНИЯ ОТРЕЗКОВ А, В, С";  
 ВВОД ДАННЫХ: А, В, С;  
 ЕСЛИ  $A+B > C$  И  $B+C > A$  И  $A+C > B$   
 ТО ВЫВОД: "ТРЕУГОЛЬНИК СУЩЕСТВУЕТ"  
 ИНАЧЕ ВЫВОД: "ТРЕУГОЛЬНИК НЕ СУЩЕСТВУЕТ"

ВСЕ;  
 КНЦ;

Примечание. Имеет место двойное логическое умножение "И" (конъюнкция).

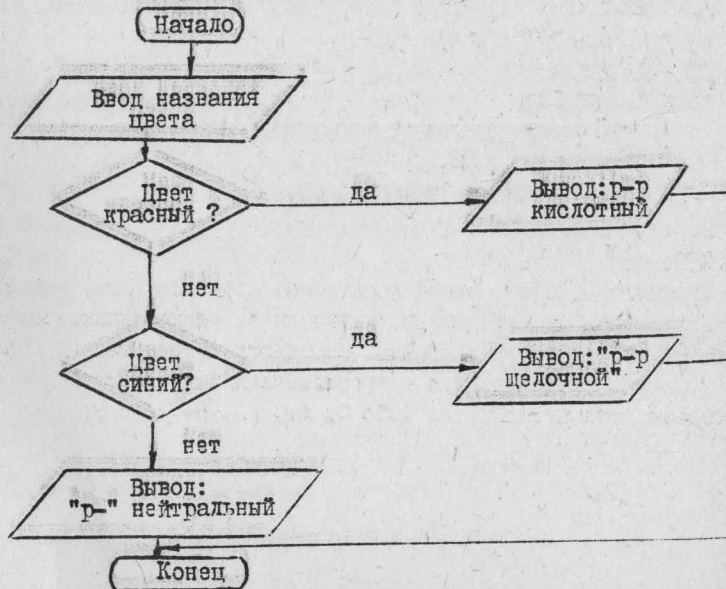
**Упражнение 5.**

Составить программу определения кислотности раствора по цвету лакмусовой бумажки.

Кислотность определяется по таблице

Цвет лакмусовой бумажки	Раствор
Красный	Кислотный
Синий	Щелочной
Белый	Нейтральный

Составим программу с помощью условного предписания. Для этого составим блок-схему.



ПРОЦ РАСТВОР I;  
 ВЫВОД: "ВВЕДИТЕ ЦВЕТ ЛАКМУСОВОЙ БУМАЖКИ (КРАСНЫЙ, СИНИЙ, БЕЛЫЙ)";  
 ВВОД ТЕКСТОВ: ЦВ;  
 ЕСЛИ ЦВ="КРАСНЫЙ" ТО ВЫВОД: "р-р КИСЛОТНЫЙ"  
 ИНАЧЕ ЕСЛИ ЦВ="СИНИЙ" ТО ВЫВОД: "р-р ЩЕЛОЧНОЙ"  
 ИНАЧЕ ВЫВОД: "р-р НЕЙТРАЛЬНЫЙ"

ВСЕ;

КНЦ;

Эта же программа с помощью предписания выбора:

ПРОЦ РАСТВОР 2;  
 ВЫВОД: "ВВЕДИТЕ ЦВЕТ ЛАКМУСОВОЙ БУМАЖКИ(КРАСНЫЙ, СИНИЙ, БЕЛЫЙ)";  
 ВВОД ТЕКСТОВ: ЦВ;  
 ВЫБОР ЦВ ИЗ  
 "КРАСНЫЙ": ВЫВОД: "р-р КИСЛОТНЫЙ";  
 "СИНИЙ": ВЫВОД: "р-р ЩЕЛОЧНОЙ";  
 "БЕЛЫЙ": ВЫВОД: "р-р НЕЙТРАЛЬНЫЙ";  
 ИНАЧЕ ВЫВОД: "НЕВЕРНО ЗАДАН ЦВЕТ"

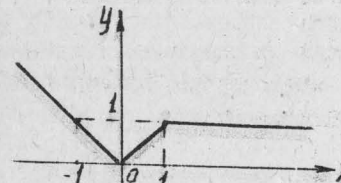
ВСЕ;

Очевидно преимущество второй программы. Во-первых, более понятная и компактная запись; во-вторых, существует гарантия от неверно заданного цвета. В первой программе при любом цвете, отличном от красного и синего, результатом будет вывод: "РАСТВОР НЕЙТРАЛЬНЫЙ".

**Упражнение 6.**

Составить программу вычисления функции, заданной графиком:

при  $X < 0$   $Y = -X$ ;  
 при  $0 \leq X < 1$   $Y = X$ ;  
 при  $X \geq 1$   $Y = 1$ ;



ПРОЦ ГРАФИК;  
 ВЫВОД: "ВВЕДИТЕ ЗНАЧЕНИЕ X";  
 ВВОД ДАННЫХ: X;  
 ВЫБОР ИЗ  
 X 0: -X --> Y!

X I: X → Y  
 ИНАЧЕ I → Y  
 ВСЕ;  
 ВЫВОД: "Y=", Y;  
 КНЦ;

Примечание. Возможен и другой вариант без "иначе":

ВЫБОР ИЗ  
 X < 0: -X → Y!  
 X < I: X → Y!  
 X ≥ I: I → Y  
 ВСЕ;

Упражнение 7.

Составить программу решения уравнения вида AX+B=0, в зависимости от A и B.

При A=0 и B=0 корней бесконечно много,  
 при A=0 и B≠0 корней нет,  
 при A≠0 корень один; X=-B/A.

ПРОЦ КОРНИ;

ВЫВОД: ВВЕДИТЕ ЗНАЧЕНИЯ КОЭФФИЦИЕНТОВ A И B;

ВВОД ДАННЫХ: A, B;

ВЫБОР ИЗ

A=0 и B=0: ВЫВОД: "КОРНЕЙ БЕСКОНЕЧНО"!

A=0 и B≠0: ВЫВОД: "КОРНЕЙ НЕТ"!

A≠0: ВЫВОД: "X=", -B/A

ВСЕ;

КНЦ;

После разбора подобных упражнений следует дать учащимся самостоятельно написать несколько программ на ветвление и выполнить их на ЭВМ. При составлении процедур рекомендуется придерживаться записи лесенкой для наглядности и удобства чтения.

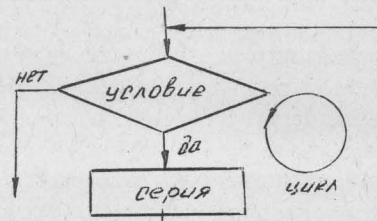
ТЕМА: ЦИКЛЫ. ЦИКЛИЧЕСКИЕ ПРЕДПИСАНИЯ

Цель:

- познакомить с понятием цикла;
- привести примеры циклических алгоритмов;
- объяснить работу циклических предписаний различных видов;
- научить составлять программы с циклами.

Теоретические сведения

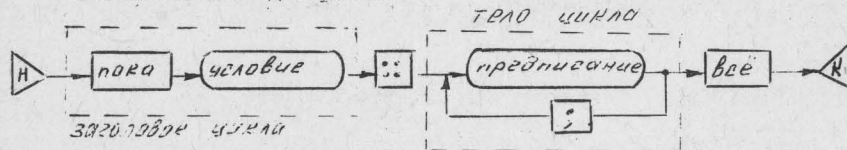
Циклом называется многократно повторяющаяся серия предписаний при выполнении какого-либо заданного условия. Блок-схема цикла:



В Рапире существует 3 вида циклических предписаний.

Цикл "ПОКА"

Синтаксическая диаграмма:



Вначале проверяется условие, указанное в заголовке цикла. Если оно истинно, выполняются все предписания от знака: ";;" (читается "повторять") до слова "ВСЕ". Затем снова проверяется условие и т.д. Если при очередной проверке условие оказалось ложным, выполнение цикла прекращается. Если оно было ложным при первой же проверке, тело цикла не выполняется ни разу.

Пример I.

I → X;

ПОКА X < =30::ВЫВОД:X;

X+I → X

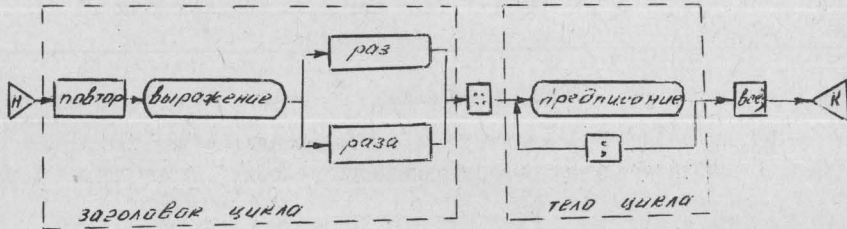
ВСЕ;



Примечание. Результатом выполнения этого предписания будет печать в столбец значений X от I до 30.

Цикл "ПОВТОР"

Синтаксическая диаграмма:



Позволяет выполнить указанную серию предписаний заданное число раз. Вначале вычисляется значение выражения в заголовке цикла. Оно должно быть целым неотрицательным числом. Затем тело цикла выполняется указанное число раз. В случае нулевого или отрицательного числа проходов тело цикла не выполняется ни разу. Варианты "раз" и "раза" эквивалентны, служат для удобства чтения и могут быть опущены.

Пример 2.

```

I → X;
ПОВТОР 30 РАЗ::ВЫВОД:X;
      X+I → X

```

ВСЕ;

Примечание. Результат тот же, что и в примере 1.

Пример 3.

```

Ю → К;
I → X;
ПОВТОР К-5 РАЗ::ВЫВОД:"ХОД I", X;
ВСЕ;

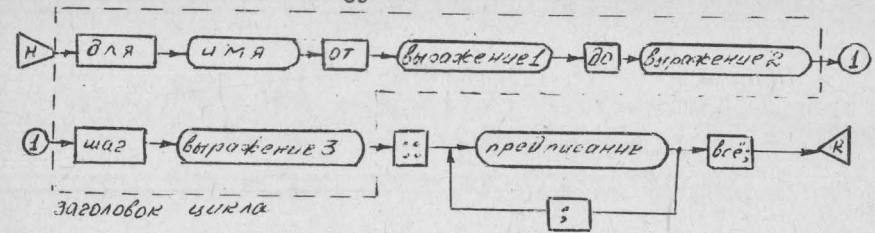
```

Примечание. Результатом будет печать 5 строчек "ХОД I", "ХОД 2", "ХОД 3", "ХОД 4", "ХОД 5", т.е. указание номеров прохода цикла.

Цикл "ДЛЯ"

Цикл "ДЛЯ" или цикл перебора существует в двух вариантах. I вариант. Цикл "ДЛЯ-ОТ". Синтаксическая диаграмма:

3  
5  
5  
1



Имя, указанное после слова "ДЛЯ", называется переменной цикла. Выражение 1 определяет начальное значение переменной цикла, выражение 2 – конечное. После слова "ШАГ" указывается приращение. Если шаг опускается, то он принимается равным единице. Значение всех трех выражений должны быть целыми или дробными числами.

Цикл выполняется в следующем порядке:

1. Вычисляются все три указанные в заголовке выражения.
2. Переменной цикла присваивается начальное значение.
3. Проверяется условие окончания цикла: текущее значение переменной цикла должно быть больше конечного значения при положительном шаге или меньше его при отрицательном.
4. Если оно ложно, выполняется тело цикла, затем к переменной цикла прибавляется значение шага, вновь проверяется условие и т.д.
5. Если при очередной проверке условие оказывается истинным, цикл завершается. При этом переменная цикла получает пустое значение.

Пример 4.

```

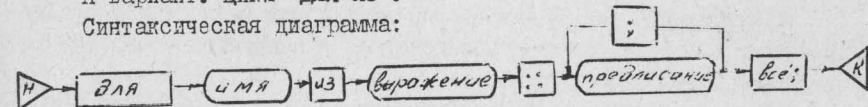
ДЛЯ X ОТ I ДО 30::ВЫВОД:X
ВСЕ;

```

Примечание. Результат будет тот же, что и в предыдущих примерах, но данный цикл наиболее естественным образом задает действия, которые мы специально организовывали в примерах I и 2.

II вариант. Цикл "ДЛЯ-ИЗ".

Синтаксическая диаграмма:



Предназначен для поэлементной обработки текстов. При выполнении этого предписания переменная цикла поочередно принимает значение каждой литеры текста (от первой к последней).

Пример 5.

О → СЧ;

ДЛЯ БУКВА ИЗ КНИГА::

ЕСЛИ БУКВА="А" ТО СЧ+1 → СЧ

ВСЕ;

ВСЕ;

ВЫВОД:"В ТЕКСТЕ КНИГА",СЧ,"БУКВ "А"";

Примечание. Подсчитывается количество букв "А" в тексте с именем КНИГА. При встрече буквы А к счетчику (СЧ) прибавляется единица.

Методические рекомендации

Циклические алгоритмы позволяют наиболее эффективно использовать ЭВМ и практически не существует задач, где бы не использовались циклы.

На Рапире существует большой набор средств для организации циклов в виде различных циклических предписаний. Нет необходимости сразу разбирать все виды предписаний. Можно начать с предписания "ПОВТОР", записав с его помощью программу для упражнения 2 (урок 4).

Упражнение 1.

ПРОЦ СНЕГ;

Ø → ОБЩ;

ПОВТОР 3 РАЗА::

ВЫВОД:"ВВЕДИТЕ РАДИУС";

ВВОД ДАННЫХ:Р;

4:ПМ-Рез3/3 → ОБ ; ОБ+ОБЩ → ОБЩ

ВСЕ;

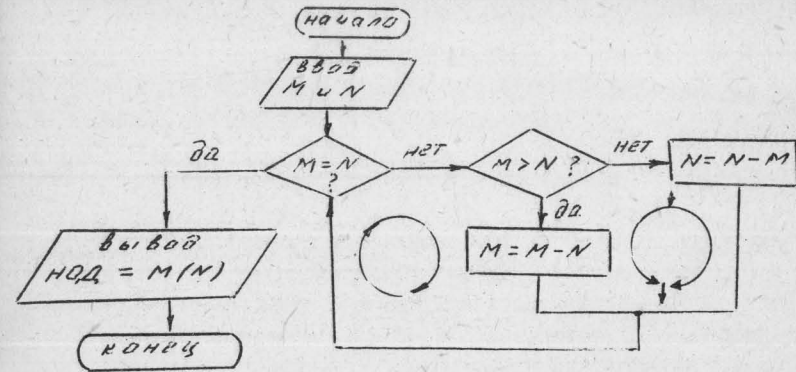
ВЫВОД:"ОБЪЕМ СНЕГОВИКА=",ОБЩ;

КНЦ;

Примечание. Объем каждого из трех шаров вычисляется и заносится в блок ОБ, содержимое которого суммируется с ОБЩ. Вначале в ОБЩ заносится ноль.

Затем можно перейти к предписанию "ПОКА". В качестве примера можно использовать алгоритм Евклида (определение наибольшего общего делителя НОД), который подробно разобран в [2].

Вначале составим блок-схему:



Упражнение 2.

ПРОЦ НОД;

ВЫВОД:"ВВЕДИТЕ ДВА НАТУРАЛЬНЫХ ЧИСЛА";

ВВОД ДАННЫХ: M, N;

ПОКА M/=N::

ЕСЛИ M > N ТО M - N → M

ИНАЧЕ N - M → N

ВСЕ;

ВСЕ;

ВЫВОД:"НОД=",M;

КНЦ;

Примечание. Цикл повторяется до тех пор, пока два числа не сравняются, а потом любое из них может быть принято за НОД. Здесь имеет смысл ввести ручную прокрутку, т.е. задать два натуральных числа и исполнить алгоритм, имитируя работу ЭВМ. Тем самым алгоритм становится более понятным и наглядным.

Предписание "ДЛЯ" можно рассмотреть на следующем примере.

Упражнение 3.

Составить программу для нахождения суммы и произведения M первых чисел натурального ряда.

$$S = 1+2+3+\dots+M$$

$$P = 1 \times 2 \times 3 \times \dots \times M$$

ВЫВОД:"ВВЕДИТЕ ЧИСЛО M";

ВВОД ДАННЫХ:M;

О → S ; I → P;

5  
5  
5  
1

ДЛЯ X ОТ I ДО M: S + X → S;  
P ← X → P;

ВСЕ;  
ВЫВОД: "СУММА РАВНА", ;  
ВЫВОД: "ПРОИЗВЕДЕНИЕ РАВНО", P;  
КНЦ;

Примечание. В блоки S и P засылаются начальные значения в S-0, в P (произведение) - I.

После разбора циклических предписаний необходимо дать учащимся возможность самостоятельно составить несколько программ с циклами и затем исполнить их на ЭВМ.

Упражнение 4.

Составить программу, при выполнении которой на экране дисплея будет печататься N раз имя любого ученика.

ПРОЦ ПЕЧИМЯ;  
ВЫОД: "ВВЕДИТЕ ВАШЕ ИМЯ";  
ВВОД ТЕКСТОВ: ИМЯ;  
ВЫВОД: "СКОЛЬКО РАЗ ПЕЧАТАТЬ?"  
ВВОД ДАННЫХ: N;  
ПОВТОР N РАЗ: :ВЫВОД: ИМЯ  
ВСЕ;  
КНЦ;

Упражнение 5.

Составить программу нахождения остатка от деления натуральных чисел A и B.

Алгоритм. Пока A > B последовательно вычитать B из A.  
ПРОЦ ОСТ;  
ВЫВОД: "ВВЕДИТЕ A И B (A > B)";  
ВВОД ДАННЫХ: A, B;  
ПОКА A > = B: :A - B → A  
ВСЕ;  
ВЫВОД: "ОСТАТОК=", A;  
КНЦ;

Упражнение 6.

Вычислить значения функции  $Y = 6X^2 + 1,5$  на отрезке X [1; 10] с шагом 0,1.

ПРОЦ ТАБ;  
ВЫВОД: "ФУНКЦИЯ  $Y = 6X^2 + 1,5$ ";  
ДЛЯ X ОТ 1 ДО 10 шаг 0,1: :  $6X^2 + 1,5$   
ВЫВОД: "X=", X, ", Y=", Y  
ВСЕ;

Защикливание

Во время исполнения циклов "ДЛЯ-ОТ" и "ПОКА" возможно так называемое защикливание, т.е. невыполнение условия окончания цикла. Причины этого разнообразны, но чаще всего они лежат в неверной организации цикла в программе.

Приведем примеры.

Пример 6.

ДЛЯ X ОТ 1 ДО I+IE-II ШАГ IE-I2: :.....  
ВСЕ;

Хотя цикл должен выполняться II раз, произойдет защикливание, потому что  $\frac{I.0000000000}{+0.0000000001}$  и после отсекаания двенадцати значащих  $\frac{I.0000000000}{I.0000000001}$

цифр очередное значение переменной окажется точно таким же, как до приращения, т.е. приращение переменной фактически оказалось нулем. Поэтому нужно быть осторожными с дробными числами, подобная ситуация может возникнуть в любом месте.

Пример 7.

I → X;  
ПОКА X < = 30: :ВЫВОД X  
ВСЕ;

Совершенно очевидно, что условие цикла всегда будет истинным, потому что в цикле X не получает приращения и программа защикливается.

Здесь нужно следить за тем, чтобы выражения, входящие в условие цикла "ПОКА", изменялись в теле цикла и в определенный момент приводили к нарушению условия.

Цикл "ДЛЯ-ИЗ" защиклиться не может, так как перебираемый текст имеет конечную мощность.

Цикл "ПОВТОР" тоже не может защиклиться, потому что число проходов - конечное число.

Чтобы прервать исполнение защиклившейся программы, надо нажать функциональную клавишу "F I" на клавиатуре ЭВМ "АГАТ".

3  
5  
5  
1



ПРИЛОЖЕНИЕ

ТЕМАТИЧЕСКИЙ ПЛАН  
по курсу "Основы программирования на языке Рапира"  
(10 класс, 34 часа, 2 ч/нед.)

Т е м а	Количество часов	
	Теория	Практика
1. Введение Язык программирования "Рапира". Объекты и правила языка	I	-
2. ПЭВМ "Агат". Система "Школьница". Правила работы на ПЭВМ	-	I
3. Предписание присваивания. Имена. Представление чисел	I	-
4. Составление и исполнение линейных программ в диалоговом режиме	2	2
5. Тексты и операции над ними	2	2
5. Предписания ввода-вывода	2	-
6. Процедура. Составление и исполнение диалоговых программ в режиме редактирования	2	2
7. Ветвление. Условное предписание. Составление программ	2	2
8. Предписание выбора. Составление и исполнение программ	2	2
9. Циклы. Циклические предписания. Составление и исполнение программ	3	3
10. Составление и отладка программ из школьного курса математики, физики, химии и т.д.	-	3
	I7	I7

Всего на курс 34 часа

СПИСОК ЛИТЕРАТУРЫ

1. Звенигородский Г.А. Первые уроки программирования. - М.: Наука, 1985. - 206 с.
2. Основы информатики и вычислительной техники. Учебное пособие в двух частях. Под ред. А.П.Ершова и В.М.Монахова. - М.: Просвещение, 1986.
3. Пакет прикладных программ автоматизации школьного учебного процесса. Руководство программиста. АН СО ВЦ, 1985. - 228 с.

СОДЕРЖАНИЕ

	<u>Стр.</u>
ВВЕДЕНИЕ .....	3
УРОК 1. ПЭВМ "Агат". Система "Школьница".....	5
УРОК 2. Графика ПЭВМ "Агат".....	8
УРОК 3. Представление чисел. Вычисление числовых вы- ражений.....	11
УРОК 4. Язык программирования "Рапира". Основные пра- вила. Предписание присваивания.....	14
УРОК 5. Исполнение простейших линейных программ на ЭЭМ. Диагностика ошибок и их исправление в диалоговом режиме.....	18
УРОК 6. Тексты и операции над ними.....	20
УРОК 7. Режим запоминания. Оформление, вызов и редак- тирование процедуры.....	22
УРОК 8. Предписания ввода и вывода. Диалоговые проце- дуры.....	25
УРОК 9. Ветвление. Условное предписание. Предписание выбора.....	29
УРОК 10. Циклы. Циклические предписания.....	37
СПИСОК ЛИТЕРАТУРЫ.....	45
ПРИЛОЖЕНИЕ. Тематический план по курсу "Основы програм- мирования на языке "Рапира".....	44

Подписано в печать 20.05.88.  
Формат 60x84 1/16  
Бумага белая оберточная.  
Печать оперативная.  
Объем печатных листов  
Тираж 100.  
Заказ № 3551.  
Бесплатно.

Куйбышевский областной институт усовершенствования учителей,  
г. Куйбышев, Московское шоссе, 125а

Отпечатано на ротационной типографии имени Мяги КПО,  
г. Куйбышев, ул. Венника, 60