

89-4

5988

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СССР
ПО ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКЕ
И ИНФОРМАТИКЕ

ВСЕСОЮЗНЫЙ МЕЖОТРАСЛЕВОЙ
НАУЧНО-УЧЕБНЫЙ ЦЕНТР
ПО ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКЕ
И ИНФОРМАТИКЕ

Г.Н.Волкова

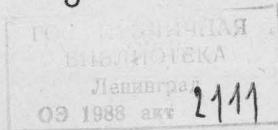
ОСНОВЫ ИНФОРМАТИКИ
И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
ДЛЯ ЭЭВМ "АГАТ"
(для учащихся 9-х классов)

Москва 1988

Практикум

© ВМНУЦ ВТИ, 1988

88-72909



ОБЩИЕ ПОЛОЖЕНИЯ

Преподавание курса "Основы информатики и вычислительной техники" введено во всех средних общеобразовательных школах в соответствии с принятым постановлением по обеспечению компьютерной грамотности учащихся средних учебных заведений и широкого внедрения электронно-вычислительной техники в учебный процесс в 1985-86 учебном году.

В соответствии с программой преподавание курса осуществляется в 9-х классах средних общеобразовательных школ в объеме 34 ч, т.е. один час в неделю. В 10-х классах в зависимости от возможности организовать практическую работу школьников на ЭВМ обучение по этому курсу должно вестись в следующем объеме: в школах, имеющих вычислительную технику или возможность организовать систематические занятия в вычислительных центрах, - 68 ч, т.е. 2 ч в неделю; в школах, не имеющих таких возможностей, - 34 ч, т.е. один час в неделю.

Настоящий практикум предназначен для проведения занятий с учащимися средних учебных заведений по курсу "Основы информатики и вычислительной техники" в учебных классах (кабинетах вычислительной техники) на базе персональной ЭВМ "Агат".

Материал раздела "Практические занятия" может быть использован в качестве раздаточного для учащихся на время выполнения практической работы, что повышает самостоятельное участие школьников в работе и облегчает труд преподавателя.

Практикум составлен в соответствии с действующей в настоящее время Программой курса для средних учебных заведений и рекомендаций органов народного образования.

Целью практических занятий является:

1. Закрепление полученных теоретических знаний:

- средств и методов описания алгоритмов;
- основных типов алгоритмов (линейный, разветвляющийся, циклический);

- программ как формы представления алгоритма для ЭВМ;
- основ программирования на языке Бейсик-Агат;
- принципов действия и устройств ЭВМ на примере персональной ЭВМ "Агат";
- применения и роли компьютеров в различных отраслях деятельности человека.

2. Приобретение учащимися практических навыков общения с ЭВМ на примере персональной ЭВМ "Агат":

- умение подготовить ЭВМ к работе;
- освоение клавиатуры;
- умение вводить в память ЭВМ числа и переменные;
- умение корректировать введенные в ЭВМ данные;
- умение вводить и отлаживать программу на ЭВМ;
- умение подключать к ЭВМ внешние устройства, управлять работой ЭВМ.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ПРОВЕДЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

В существующих в настоящее время методических рекомендациях по курсу содержатся только ориентировочные методические схемы. Применительно к конкретным условиям имеющаяся схема наполняется определенным содержанием, придавая ей соответствующую форму.

Содержание практических занятий данного практикума определено теоретическим материалом, изучаемым на занятиях, и возможностями персональной ЭВМ "Агат".

Все практические работы оформлены по единому плану и включают в себя следующие пункты:

- название темы работы;
- указание цели работы;
- продолжительность выполнения;
- краткие теоретические сведения;
- контрольные вопросы;
- порядок выполнения работы;
- требования к отчету;
- задания к работе (с вариантами);
- пример выполнения задания.

При изложении основных теоретических сведений во всех работах подчеркивается связь между конструкциями школьного алгоритмического языка и языка конкретной ПЭВМ Бейсик-Агат.

Контрольные вопросы могут быть использованы для фронтального опроса. Ответы на контрольные вопросы, не нашедшие прямого отражения в теоретической части, приведены в Приложении I. Теоретические сведения, помещаемые в практической работе, могут быть использованы для самоподготовки учащихся к выполнению данной практической работы и как справочный материал в случае возникновения трудностей в процессе ее выполнения.

Задания к практическим работам целесообразно выдавать с опережением, чтобы учащиеся могли хорошо продумать ход решения задачи, построить алгоритм, написать программу, а время, отведенное на практические занятия, использовать для отладки и выполнения программы на ПЭВМ.

Выдавать задания к практическим работам можно в качестве домашних работ при изучении теоретической части курса.

При выполнении заданий учащиеся обязаны составить требуемую программу на школьном алгоритмическом языке и на языке Бейсик-Агат. Необходимость составления алгоритма решаемой задачи в виде блок-схемы определяется преподавателем.

Практические занятия на ПЭВМ для учащихся 9-х и 10-х классов заканчиваются зачетными работами. Это самостоятельные работы учащихся, выполняемые под руководством преподавателя. Варианты зачетных работ могут быть даны учащимся после первого практического занятия с целью постепенного ее выполнения по мере закрепления теоретических знаний и приобретения навыков работы на ПЭВМ. Время, отведенное на зачетное занятие, должно использоваться для отладки, выполнения заданий на ПЭВМ, защиты зачетной работы у преподавателя.

Порядок выполнения работ может изменяться исходя из возможностей проведения занятий в кабинетах вычислительной техники и изучения теоретической части курса.

ПРАКТИЧЕСКИЕ ЗАНЯТИЯ

ПРАКТИЧЕСКАЯ РАБОТА 1

Тема: Знакомство с ПЭВМ.

Цель работы: Ознакомление с правилами техники безопасности при работе в кабинете вычислительной техники, основными блоками ПЭВМ, правилами включения-выключения ПЭВМ, правилами занесения информации с клавиатуры; редактирование текстов на экране видеоконтрольного устройства (ВКУ).

Продолжительность работы - 2 ч.

Теоретические сведения. Перед началом работы в кабинете вычислительной техники необходимо ознакомиться с правилами, соблюдение которых обеспечит безопасную для человека стабильную работу техники. Памятка с правилами находится на каждом рабочем месте. Текст ее приведен в Приложении 2.

Персональный компьютер предназначен для автоматической обработки информации и обеспечивает ввод программы и данных с клавиатуры или магнитного диска, редактирование программы, работу в диалоговом режиме, вывод программы и результатов обработки на экран ВКУ, печатающее устройство или магнитный диск.

ПЭВМ "Агат" состоит из следующих устройств и блоков:

- системного блока с встроенным накопителем на гибких магнитных дисках (НГМД);

- блока клавиатуры;

- видеоконтрольного устройства;

- мозаично-печатающего устройства.

Системный блок обеспечивает выполнение программ, хранение и обработку информации, управление всеми устройствами системы, организацию диалога с пользователем. Это основной блок, объединяющий в единую систему все составные части ПЭВМ. В системном блоке размещается и встроенный НГМД. Гибкие магнитные диски предназначены для долговременного хранения информации и позволяют оперативно изменять программы и данные в ПЭВМ.

Блок клавиатуры предназначен для ввода информации в ПЭВМ. При нажатии на клавишу замыкаются контакты, и в электронную схему попадает код символа соответствующей клавиши. На клавиатуре выделено три зоны: символьные клавиши, функциональные клавиши и клавиши управления











работой ПЭВМ.

Символьные клавиши содержат русские и латинские буквы, цифры, знаки препинания и арифметических действий и др. специальные символы. Клавиатура может работать либо только с латинским, либо только с русским набором букв с соответствующей световой индикацией. Перейти из одного режима в другой можно путем одновременного нажатия клавиш РЕГ и РУС, либо РЕГ и ЛАТ, что подтверждается загоранием соответствующего световказателя на блоке клавиатуры.

Цифры и символы, находящиеся на верхнем регистре, можно получить нажатием соответствующей клавиши независимо от того, какой регистр зафиксирован - РУС или ЛАТ. Символы, находящиеся на нижнем регистре, получают одновременным нажатием клавиши РЕГ и клавиши с нужным символом.

Функциональные клавиши выделены на клавиатуре в особую группу. В отличие от символьных за каждой функциональной клавишей закрепляется определенная функция, которая выполняется при ее нажатии.





Клавиши управления:

	- сброс		- управление
	- перевод строки		- управление курсором вниз
	- повторение		- управление курсором вверх
	- регистр		- управление курсором вправо
	- редактирование		- управление курсором влево

Назначение и использование клавиш управления приведены в Приложении 3.

Видеоконтрольное устройство цветного отображения обеспечивает диалоговый режим работы. Экран ВКУ предназначен для отображения вводимой и выводимой символьной и графической информации. Каждый символ располагается в определенном столбце и определенной строке (позиции) экрана. Одна из позиций отмечена определенным знаком - мигающим кур-






сором. Символ, вводимый с клавиатуры, отображается в позиции, отмеченной курсором. Курсор при этом смещается на одну позицию вправо по текущей строке до тех пор, пока не дойдет до ее края, и переходит на начало следующей строки.


Если вводимая информация не помещается на одной строке экрана, то новая вводимая строка занимает место нижней, а все ранее введенные строки смещаются на одну строку вверх. При этом самая верхняя строка исчезает с экрана. Перемещается курсор по экрану ВКУ с помощью клавиш , , ,  (их нажатие вызывает смещение курсора на одну позицию вправо, влево, вверх и вниз соответственно).









Мозаично-печатающее устройство, или принтер, не является обязательным устройством и может быть отключен от ПЭВМ без нарушения нормальной работы. Принтер предназначен для получения твердой копии информации, хранимой в ПЭВМ.

Основные операции редактирования. Для замены одного символа другим курсор подводится к заменяемому символу и нажимается клавиша, соответствующая вводимому символу. Удаление (стирание) символа в позиции курсора выполняется нажатием клавиши "Пробел" (самая длинная клавиша, находящаяся в нижнем ряду на блоке клавиатуры). Чтобы внести изменения в строку программы, находящейся в памяти ПЭВМ, нужно ввести номер этой строки и правильный текст. Строка с указанным номером заменится на вновь введенную.

Если изменения в строке незначительны, то целесообразно вводить правильный вариант не полной строки программы, а откорректировать старый текст. Для этого текст неверной строки должен быть на экране ВКУ. Курсор подводится к первому неверному символу строки и нажатием нужной клавиши заменяется новым.

Если нужно удалить лишние символы в середине строки, курсор устанавливается на первый удаляемый символ и нажимается клавиша . В позиции курсора появляется мигающая точка. Далее клавишей  проходят все удаляемые символы и, выведя мигающую точку на символ, не подлежащий удалению, нажимают клавишу . Оставшиеся символы проходят курсором до конца с помощью клавиши  или заменяют нажатием клавиши с нужным символом. Исправленная строка запоминается в оперативной памяти ПЭВМ при нажатии клавиши .

Если нужно убрать конец строки, то курсором проходят оставляемые в строке символы и нажимают клавишу .

Если строку нужно раздвинуть для вставки недостающих символов, то курсор устанавливается под ту позицию в редактируемой строке, где должен быть вставлен первый недостающий символ. Клавишей  или  курсор перемещается на чистое поле экрана и вводятся недостающие символы. Затем курсор возвращают на редактируемую строку клавишей  или . Устанавливаем режим редактирования, нажав клавишу . Вместо курсора появляется мигающая точка. Возвращаем курсор клавишей  на первый символ, следующий за вставкой. Выходим из режима редактирования, нажав клавишу . Курсор перемещается до конца редактируемой строки, нажимаем клавишу . ПЭВМ запоминает исправленную строку программы.

Порядок включения-выключения ПЭВМ "Агат":

1. Включаем ВКУ, нажимая на передней панели клавишу.
2. Включаем системный блок тумблером на задней панели при закрытой крышке дисководов.
3. Вставляем гибкий магнитный диск в НГМД, накопитель закрывается, начинается загрузка дисковой операционной системы, о чем сообщает светящийся индикатор на НГМД. По завершении загрузки индикатор гаснет, на экране ВКУ появляются знак правой квадратной скобки и мигающий курсор. ПЭВМ готова к работе.

Выключение ПЭВМ осуществляется в обратном порядке.

Контрольные вопросы

1. Перечислить правила техники безопасности при работе в кабинете вычислительной техники.
2. Объяснить назначение основных блоков ПЭВМ.
3. Указать последовательность действий при включении и выключении ПЭВМ.
4. Объяснить назначение клавиш управления.
5. Определить последовательность нажатия клавиш для ввода текста:

375+18*S/3*R, если зафиксирован латинский регистр.

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задания 1, 2, 3;
- оформить отчет.

Отчет по практической работе I должен содержать:

- название и цель работы;
- перечень основных блоков ПЭВМ "Агат";
- описание порядка включения-выключения ПЭВМ;
- ответ на вопрос: в какой последовательности нужно нажать клавиши для ввода текста:

$$\text{СУММА } S=A^2+(4*B*C)$$

если зафиксирован латинский регистр?

Задание I выполняется с использованием обучающей программы "Знакомство с клавиатурой", которая выдает упражнения и контролирует правильность выполнения. Программа начинает работать при последовательном нажатии клавиш R U N F

Задание 2.

1. Введите предложение:

10_ ? "ЭВМ РЕДАКТИРУЕТ ТЕКСТ" F

2. Замените слово:

- а) "ЭВМ" словом "ОНА";
- б) "ТЕКСТ" словом "ТЕКСТЫ";
- в) "РЕДАКТИРУЕТ" словом "ИСПРАВЛЯЕТ".

Задание 3. С помощью символа "*" нарисуйте на экране прямоугольник, треугольник, ромб.

Пример выполнения задания.

R U N F

* * * ЗНАКОМСТВО С КЛАВИАТУРОЙ * * *

НАБЕРИТЕ: 1987

? 1987 F

ПРАВИЛЬНО!

НАБЕРИТЕ: 0.876

? 0,876 F

НЕПРАВИЛЬНО! ПРОВЕРЬТЕ, КАКИМ АЛФАВИТОМ И РЕГИСТРОМ ВЫ ПОЛЬЗУЕТЕСЬ!

НАБЕРИТЕ: 0.876

? _ и т.д.

После выполнения Вами всех упражнений программа выводит на экран ВКУ одно из трех сообщений:

ХОРОШО! (ОТЛИЧНО)

ВЫ ГОТОВЫ К РАБОТЕ С ПРОСТЕЙШИМИ ПРОГРАММАМИ НА ЯЗЫКЕ БЕЙСИК-АГАТ!

10

УДОВЛЕТВОРИТЕЛЬНО!

ВАМ НЕОБХОДИМО ПОРАБОТАТЬ С ДАННОЙ ПРОГРАММОЙ ЕЩЕ РАЗ!

ПЛОХО! НО НЕ БЕЗНАДЕЖНО!

БУДЬТЕ БОЛЕЕ ВНИМАТЕЛЬНЫ ПРИ РАБОТЕ С КОМПЬЮТЕРОМ!

Если Вы получили оценку "хорошо" или "отлично", поднимите руку и сообщите об этом преподавателю, а затем выполняйте следующее задание.

Если же Вы получили оценку "удовлетворительно" или "плохо", то программа предложит Вам повторить все упражнения еще раз.

ПРАКТИЧЕСКАЯ РАБОТА 2

Тема: Линейный алгоритм, выполнение его на ЭЕМ.

Цель работы: Ознакомление с основными системными командами, правилами записи линейного алгоритма и его выполнением на языке Бейсик.

Продолжительность работы - 2 ч.

Теоретические сведения. Для выполнения алгоритма, записанного на алгоритмическом языке, на ПЭВМ необходимо знание системных команд, приведенных ниже.

Команда	Выполняемое действие
NEW	Очистить память
RUN	Запустить программу на выполнение
LIST	Вывести на экран ВКУ текст программы
DEL	Стереть строки программы из памяти

Команда LIST имеет пять вариантов записи.

Запись	Выполняемое действие
LIST	Вывод на экран ВКУ всего текста программы

Запись	Выполняемое действие
LIST N1	Вывод на экран ВКУ текста программы с номером N1
LIST N1,N2	Вывод на экран ВКУ текста программы со строки с номером N1 по строку с номером N2
LIST N1,	Вывод на экран ВКУ текста программы со строки с номером N1 до конца текста
LIST,N2	Вывод на экран ВКУ текста программы сначала до строки с номером N2

Например, LIST,40 означает, что надо вывести на экран ВКУ текст программы, начиная с наименьшего встреченного номера строки программы по строку с номером 40.

Команда RUN имеет два варианта записи.

Запись	Выполняемое действие
RUN	Запуск программы на выполнение, начиная с наименьшего номера строки, встреченного в тексте программы
RUN N1	Запуск программы на выполнение, начиная с номера строки N1 (например, RUN 100)

Команда NEW не имеет параметров и записывается всегда однозначно.

Команда DEL записывается всегда с двумя параметрами, указанными через запятую. Но при этом значения параметров влияют на вариант выполнения.

Запись	Выполняемое действие
DEL N1,N2	Удаление из текста программы группы строк, начиная с номера N1 по номер N2
DEL N1,N1	Удаление из текста программы одной строки с номером N1

Возможен более простой способ удаления одной строки без использования оператора DEL. В этом случае после правой квадратной скобки указывается номер удаляемой строки и нажимается клавиша \square .

Чтобы выполнить на ПЭВМ программу, записанную на алгоритмическом языке, нужно переложить ее на язык, понятный машине. В данном случае мы будем говорить о языке Бейсик-Агат.

На клавиатуре ПЭВМ имеются клавиши, нажатие которых вызывает выполнение соответствующих арифметических операций.

Клавиша	Действие, выполняемое при нажатии клавиши
\square	Возведение в степень
\times	Умножение
\square	Деление
$+$	Сложение
$-$	Вычитание

В выражениях на языке Бейсик порядок выполнения действий может быть выражен таблицей:

Порядок выполнения операций	Выполняемая операция
1	Действия в скобках
2	Вычисление стандартных встроенных функций SQRT, SIN и др.
3	Возведение в степень
4	Одноместный минус, изменяющий знак стоящей за ним величины
5	Умножение, деление
6	Сложение, вычитание

Операции, имеющие равный приоритет, выполняются слева направо. При использовании скобок первым выполняется самое внутреннее выражение, затем внешнее по отношению к предыдущему и т.д. Другими словами, действуют те же правила, что и в обычных математических формулах.

В языке Бейсик используются три типа констант и переменных. В скобках указано соответствие типа обозначению на алгоритмическом языке (табл. 1).

Т а б л и ц а 1

Тип величины	Диапазон использования	Примеры записи	Определение
Вещественные (вещ)	$10^{-38} + 10^{+38}$	50.3 -148.12 42861 0.85 6.02E23 1.6E-19	Набор из одной или нескольких десятичных цифр, положительный или отрицательный, целый или дробный Целая часть от дробной отделяется точкой
Целые (цел), (нат)	-32768% + +32767%	18% -375% 0%	Набор десятичных цифр, оканчивающихся знаком %
Строковые, или литерные, или текстовые (лит)	0+255 символов	ПАРОХОД P=СУММЕ SIRIUS R20 A# BC N_(V)	Набор буквенно-цифровых или специальных символов. Имя строковой константы или переменной оканчивается знаком # (солнышко). Явно задаваемая строковая константа ограничивается с двух сторон двойными кавычками (" ")

Каждая строка программы на языке Бейсик должна начинаться номером. Номер строки необходим для:

- указания интерпретатору Бейсика порядка выполнения программы;
- изменения обычного порядка выполнения операторов при условной и безусловной передаче управления;
- изменения отдельных строк при отладке программы.

Номер строки - это положительное целое число в диапазоне 0 - 999. Рекомендуется использовать номера строк с шагом 10. Это позволяет при редактировании вставлять в программу пропущенные операторы. В каждой строке можно записать один или несколько операторов. Если в одной строке записывается несколько операторов, их надо разделить

двоеточием, например:

100 A=2: B=3: C=A+B

Одна строка программы может занимать на экране более одной строки, содержащей 32 символа. Максимально допустимая длина строки программы - 255 символов. Строки программы можно вводить в произвольном порядке (например, сначала строку 100, а потом строку 10). При размещении в памяти они автоматически упорядочиваются.

В языке Бейсик нет таких строгих правил оформления программы, как в алгоритмическом языке. Желательно в первых строках программы в комментариях дать короткое объяснение сущности алгоритма и используемых переменных. Для этого используется оператор REMark (заметка):

REM <строка символов>

Встретив REM, интерпретатор игнорирует все следующие за этим ключевым словом символы до конца физической строки экрана и переходит к выполнению следующей строки программы.

В отличие от алгоритмического языка в Бейсике нет ключевого слова, указывающего начало алгоритма, но есть оператор END (конец). Выполняя его, ПЭВМ переходит из программного режима в командный и ожидает ввода пользователем следующей команды, т.е. выполнение программы заканчивается. Он аналогичен ключевому слову кон, указывающему конец алгоритма.

В алгоритмическом языке нет аналогов операторам ввода-вывода, используемым в Бейсике. Для вывода на экран ВКУ информации используется оператор PRINT - печатать:

PRINT <список вывода>

Список вывода может содержать строковые константы, переменные всех типов и элементы массивов, разделенные запятой или точкой с запятой. Эти разделители управляют размещением информации на экране. Если разделителем является точка с запятой (;), то информация выводится подряд, без пробелов. Когда ";" стоит за последним элементом списка вывода, то это правило распространяется на первое значение следующего оператора PRINT, т.е. печать будет происходить в ту же строку. Иначе каждый следующий оператор PRINT осуществляет вывод с новой строки.

Разделитель "запятая" (,) задает печать в очередную зону экрана. Каждая зона кратна восьми разрядам. Стоящие подряд две запятые в списке вывода приведут к пропуску одной свободной зоны.

Ввод информации осуществляется с помощью оператора INPUT (ввод)

INPUT <список ввода>

При вводе нескольких величин их нужно перечислить через запятую. Для окончания ввода нажать клавишу **Enter**.

В списке ввода могут быть указаны имена констант или переменных целого, вещественного или литерного типа. После того как в программе начнет работать оператор INPUT, на экране ВКУ появятся знак "?" и мигающий курсор. Это означает, что программа ждет, чтобы с клавиатуры были введены значения. Но когда с программой работает не автор, а пользователь, ему трудно догадаться, ввода каких значений ожидает программа. Поэтому целесообразно использовать оператор ввода с запросом

```
INPUT " <запрос> "; <список ввода>
```

Запрос содержит информацию, которая дает пояснения о вводимых значениях, например:

```
50 INPUT "ВВЕДИТЕ ЗНАЧЕНИЯ ОСНОВАНИЯ И ВЫСОТЫ ТРЕУГОЛЬНИКА"; A, H
```

Как и в алгоритмическом языке, в Бейсике есть оператор присваивания. Это наиболее часто употребляемый оператор любого алгоритмического языка. Он единственный распознается транслятором по его форме, даже без названия. Полная форма оператора присваивания LET (давать):

$$\text{LET} \left\langle \begin{array}{l} \text{идентификатор} \\ \text{переменной или} \\ \text{элемента массива} \end{array} \right\rangle = \left\langle \begin{array}{l} \text{арифметическое} \\ \text{выражение} \end{array} \right\rangle$$

Сокращенная форма оператора присваивания опускает ключевое слово LET.

Идентификатор — это имя переменной, состоящее из последовательности букв и цифр. Первой в имени обязательно должна стоять буква латинского алфавита.

Контрольные вопросы

1. Каково назначение основных системных команд?
2. Рассказать о выражениях на языке Бейсик и приоритете операций.
3. Какие типы величин используются в языке Бейсик?
4. В каких случаях используются операторы REM и END?
5. В каких случаях используются операторы PRINT и INPUT?
6. Как будет выведена на экран информация, если программа содержит следующие операторы:

```
90 V=3: T=40
```

```
100 PRINT "ОНА ИДЕТ СО СКОРОСТЬЮ-"; V, " ЗА ВРЕМЯ-"; T;
```

```
130 PRINT "МИН"
```

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет по практической работе 2 должен содержать:

- название и цель работы;
- список основных системных команд;
- таблицу порядка выполняемых действий в выражениях на языке

Бейсик;

- общий вид и краткое описание назначения операторов REM, END, PRINT, INPUT, LET;
- условие задания 1 или 2, текст составленной программы.

Задание 1. Составить программу с помощью изученных операторов, которая после команды RUN выдавала бы на экран:

```
ВВЕДИТЕ СВОЕ ИМЯ
```

```
?НАДЕЖДА
```

```
ПРИВЕТ, НАДЕЖДА!
```

```
МЕНЯ ЗОВУТ 'АГАТ',
```

```
РАД ПОЗНАКОМИТЬСЯ!
```

Отладить программу на ПЭВМ и получить результат.

Задание 2. Составить программу с помощью изученных операторов, которая после команды RUN выдавала бы на экран:

```
СКОЛЬКО СТОИТ ВАША ПОКУПКА?(РУБ)
```

```
?2.40
```

```
СКОЛЬКО ДЕНЕГ ВЫ ДАЛИ КАССИРУ?(РУБ)
```

```
?3.00
```

```
ВАША СЛАЧА-0.60
```

Отладить программу на ПЭВМ и получить результат.

Пример выполнения задания. Составить программу, используя изученные операторы, которая после команды RUN выдавала бы на экран:

```
СКОЛЬКО ВАМ ЛЕТ?
```

```
?_
```

```
СКОЛЬКО ЛЕТ ВАШЕЙ МАМЕ?
```

```
?_
```

СКОЛЬКО ЛЕТ ПАПЕ?

?

ВАМ ВМЕСТЕ ... ЛЕТ!

Программа на алгоритмическом языке:

алг сумма лет (цел А, В, С, Д)

арг А, В, С

рез Д

нач

задать вопрос: сколько вам лет?

А:=ответ

задать вопрос: сколько лет маме?

В:=ответ

задать вопрос: сколько лет папе?

С:=ответ

Д:=А+В+С

кон

Программа на языке Бейсик:

10 REM ОБЩАЯ СУММА ЛЕТ ЧЛЕНОВ СЕМЬИ

20 PRINT "СКОЛЬКО ВАМ ЛЕТ?"

30 INPUT A

40 PRINT: PRINT "СКОЛЬКО ЛЕТ ВАШЕЙ МАМЕ?"

50 PRINT: INPUT B

60 PRINT: PRINT "СКОЛЬКО ЛЕТ ПАПЕ?"

70 PRINT: INPUT C

80 D=A+B+C

90 PRINT: PRINT "ВАМ ВМЕСТЕ ";D;"ЛЕТ!"

100 END

Оператор PRINT без параметров выводит на экран ВКУ пустую строку. Это делается для лучшего восприятия выводимого на экран текста.

ПРАКТИЧЕСКАЯ РАБОТА 3

Тема: Разветвляющийся алгоритм (команда ветвления), выполнение его на ПЭВМ.

Цель работы: Изучение команды ветвления, составление алгоритмов с ее использованием, ознакомление с оператором безусловного перехода.

Продолжительность работы - 2 ч.

Теоретические сведения. Команда ветвления на алгоритмическом языке записывается следующим образом:

если условие

то серия1

иначе серия2

все

или в сокращенной форме:

если условие

то серия

все

Команда ветвления на языке Бейсик аналогична команде на алгоритмическом языке, но записывается по другим правилам. Слова, используемые в качестве ключевых, - на английском языке: IF (если), THEN (то), ELSE (иначе). Ключевое слово все не обязательно.

Общий вид оператора:

IF <условие> THEN <серия1> ELSE <серия2>

Выполняя оператор, ПЭВМ проверяет соблюдение условия. Если оно соблюдено, то выполняется <серия1>, если нет, то - <серия2>. Следует помнить, что <серия1> может состоять только из одного оператора.

В языке Бейсик также возможна и сокращенная форма оператора ветвления:

IF <условие> THEN <серия>

В этом случае при соблюдении условия выполняется оператор (один!), стоящий после THEN, а при несоблюдении - управление передается на следующую строку программы.

Отметим, что применение сокращенной формы оператора ветвления вполне достаточно для работы.

Рассмотрим теперь, какие условия можно употреблять в операторе ветвления. Элементарные условия - это обычные отношения между выражениями:

<выражение> <знак отношения> <выражение>

В качестве знаков отношений используются знаки той или иной операции сравнения:

<	Меньше	<=	Меньше или равно
>	Больше	>=	Больше или равно
=	Равно	<>	Не равно

Три знака в правом столбце - составные, состоящие из двух символов, так как на клавиатуре нет специальных клавиш \leq , \geq , \neq .

Оба выражения в условии должны быть одного типа: нельзя сравнивать число с символьной величиной! Выполняя оператор ветвления, ПЭВМ в первую очередь вычисляет значения обоих выражений, потом, сравнивая их, выясняет, соблюдено ли условие. Употребляя в программах числовые условия, особое внимание надо обращать на отношение равенства, так как может оказаться, что два числа, которые должны быть равными, все-таки немного отличаются (из-за ошибок округления).

Сравнение символьных строк на совпадение (=) или несовпадение (< >) не требует комментариев.

В версии Бейсик-Агат допускаются также составные условия, образованные из элементарных условий, соединенных ключевыми словами AND (и), OR (или) и NOT (не).

Нижеприведенный пример иллюстрирует соответствие записей на алгоритмическом языке (слева) записям на языке Бейсик (справа):

$x=0$ или $x=1$ $(X=0)$ OR $(X=1)$
не ($a=2$ и $b=3$) NOT ($A=2$ AND $B=3$)

Оператор ветвления должен разместиться в одной строке программы. Однако не всегда удается поместить в одну строку программы необходимые серии операторов. В таких случаях помогает оператор GOTO, который передает управление на строку, с которой начинается нужный фрагмент программы.

Общий вид оператора GOTO (идти к):
GOTO <номер строки>

В результате его выполнения следующим выполняемым оператором будет тот, номер которого указан за ключевым словом GOTO. Операторами GOTO следует пользоваться как можно реже, так как они делают программу трудночитаемой. В алгоритмическом языке этот оператор отсутствует.

Контрольные вопросы

1. Когда необходим оператор ветвления и как он выполняется?
2. Какие условия можно употреблять в операторе ветвления?
3. Как образуется составное условие?
4. Что произойдет, если пользователь введет следующую программу:
1Ø X=1
2Ø X=2*X
3Ø PRINT X
4Ø GOTO 2Ø

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет по практической работе 3 должен содержать:

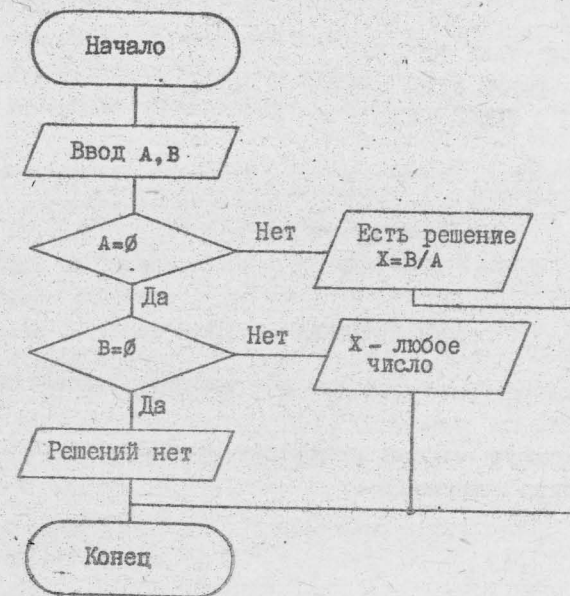
- название и цель работы;
- запись общего вида операторов ветвления и безусловного перехода;
- условие задания 1 или 2, блок-схему, программу, составленную на алгоритмическом языке и языке Бейсик, результат работы программы.

Задание 1. Написать программу, вычисляющую и показывающую на экране наибольший общий делитель (НОД) двух натуральных чисел M и N, введенных пользователем, используя алгоритм Евклида (НОД пары различных натуральных чисел совпадает с НОД пары, полученной из предыдущей заменой большего числа разностью большего и меньшего чисел).

Задание 2. Написать программу, вычисляющую и выводящую на экран большее из двух чисел (БМД).

Пример выполнения задания. Написать программу, реализующую алгоритм решения линейного уравнения $A \cdot X = B$, где A и B - любые вещественные числа.

Блок-схема



Программа на алгоритмическом языке:

алг ЛУР (вещ а, в, вещ х, лит у)

арг а, в

рез х, у

нач

если а ≠ 0

то у := "есть решение"

х := в/а

иначе

если в = 0

то у := "х-любое число"

иначе у := "решений нет"

все

все

кон

Программа на языке Бейсик:

10 REM ПРОГРАММА РЕШЕНИЯ ЛИН. УР-Я АХ=В

20 INPUT "ВВЕДИТЕ ЗНАЧЕНИЯ А И В": А, В

30 IF А=0 THEN GOTO 70

40 PRINT "ЕСТЬ РЕШЕНИЕ"

50 Х=В/А

60 PRINT "Х="; Х: GOTO 100

70 IF В < > 0 THEN GOTO 90

80 PRINT "РЕШЕНИЙ НЕТ" : GOTO 100

90 PRINT "Х - ЛЮБОЕ ЧИСЛО"

100 END

ПРАКТИЧЕСКАЯ РАБОТА 4

Тема: Циклический алгоритм (команда повторения), выполнение его на ПЭВМ.

Цель работы: Изучение команды повторения, составление алгоритмов с ее использованием.

Продолжительность работы - 2 ч.

Теоретические сведения. Команда повторения имеет особое значение для построения сравнительно коротких алгоритмов, предписывающих совершение длинной последовательности действий. Команда повторения записывается следующим образом:

пока условие

нп

серия

кп

В языке Бейсик оператор повторения записывается с помощью ключевых слов FOR (для), TO (до), STEP (шаг) и NEXT (следующий). Оператор повторения имеет заголовок цикла, тело цикла и признак конца цикла.

Общий вид заголовка цикла:

FOR $\left\langle \begin{array}{l} \text{ИМЯ} \\ \text{переменной} \end{array} \right\rangle = \langle \text{выражение1} \rangle$ TO $\langle \text{выражение2} \rangle$ [$\text{STEP} \langle \text{выражение3} \rangle$]

Часть оператора, заключенная в квадратные скобки, не обязательна. По умолчанию принимается стандартное значение шага, равное единице. Все выражения должны быть числовыми.

При выполнении цикла ПЭВМ сначала отмечает указанное имя счетчика цикла, присваивает в качестве начального значение выражения1, запоминает значение выражения2 (конечное значение счетчика). Если в заголовке цикла присутствует ключевое слово STEP, то запоминает значение выражения3 (шаг). Если STEP отсутствует, шаг получает значение "единица". Затем ПЭВМ выполняет подряд следующие операторы до тех пор, пока не достигнет строки

NEXT $\langle \text{ИМЯ переменной} \rangle$

в которой дано запомненное имя счетчика. При выполнении этого оператора к значению счетчика добавляется шаг и проверяется, не вышло ли новое значение счетчика из заданного интервала. Если значение счетчика превысило конечное значение, то работа с циклом закончена, и ПЭВМ выходит из цикла - "забывает" имя счетчика, конечное значение и шаг и приступает к выполнению оператора, следующего за оператором NEXT.

Если значение счетчика еще не достигло конечного значения, то ПЭВМ снова начинает выполнять тело цикла, начиная с оператора, следующего сразу за заголовком цикла, и все повторяется сначала.

В любом случае тело цикла выполняется хотя бы один раз.

По отношению к остальной части программы тело цикла замкнуто — не следует "входить" в него, минуя заголовок цикла, и не следует "выходить" из него, минуя NEXT. Из цикла принято "выходить" только тогда, когда при проверке обнаружено, что счетчик "перешел" конечное значение. Если надо выйти из цикла раньше, то следует присвоить счетчику значение, превышающее конечное, и по оператору GOTO выйти на NEXT.

Контрольные вопросы

1. В каких случаях используется оператор цикла?
2. Что такое счетчик цикла?
3. Что произойдет, если не указан шаг?
4. Что напечатает нижеследующая программа:

```

10 A=0
20 FOR L=15 TO 100 STEP 10
30 A=A+L
40 NEXT L
50 PRINT A
60 END
  
```

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет по практической работе 4 должен содержать:

- название и цель работы;
- запись общего вида оператора цикла;
- условие задания 1 или 2, блок-схему, программу на алгоритмическом языке и на языке Бейсик, результат работы программы.

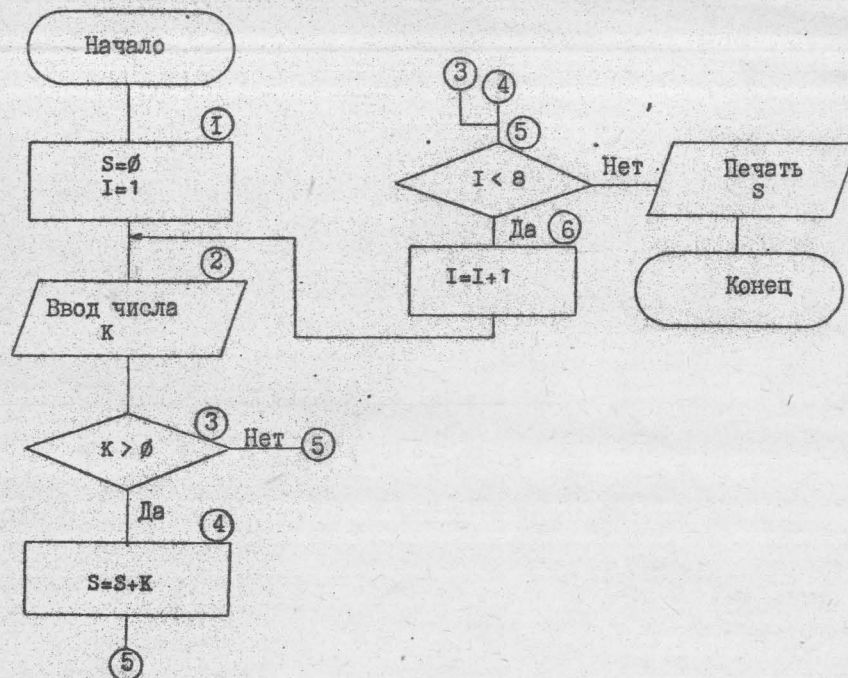
Задание 1. Составить программу, подсчитывающую $n!$. Значение n вводится пользователем. Результат напечатать.

Задание 2. Составить программу, подсчитывающую сумму квадратов i^2 . Результат напечатать.

24

Пример выполнения задания. Составить программу, суммирующую восемь положительных чисел.

Блок-схема



Программа на алгоритмическом языке:

```

алг сумма положительных чисел
  арг k
  рез s
нач  цел i
  пока i < 8
  ни
    если k < 0
      то (перейти на к1)
    иначе s=s+k
  все
к1 (печать s)
кон
  
```

Программа на языке Бейсик:

```
10 REM СУММА 8 ПОЛОЖИТЕЛЬНЫХ ЧИСЕЛ
20 S=0
30 FOR I=1 TO 8
40 INPUT "ВВЕДИТЕ ЧИСЛО";K
50 IF K<=0 THEN GOTO 70
60 S=S+K
70 NEXT I
80 PRINT S
90 END
```

ПРАКТИЧЕСКАЯ РАБОТА 5

Тема: Алгоритм, использующий табличные величины, выполнение его на ПЭВМ.

Цель работы: Ознакомление с оператором описания массивов, переменными с индексами и применение их при решении задач.

Продолжительность работы - один час.

Теоретические сведения. Кроме простых переменных используются и переменные с индексами, являющиеся элементами массивов (или табличными величинами). Линейные таблицы - это одномерные массивы (векторы). Им соответствуют переменные с одним индексом.

Прямоугольные таблицы - это двумерные массивы (матрицы). Им соответствуют переменные с двумя индексами: первый индекс - номер строки, второй - номер столбца.

Индексом может являться целая константа, или переменная, или любое выражение, которое в результате вычисления дает целое значение.

Индексная переменная записывается в виде имени, за которым следует одна или две индексные величины, заключенные в круглые скобки. Если индексных величин две, они разделяются запятой, например:

A(5,7); A(M,K); B(T+P),A1(8),...

Все массивы должны быть описаны, т.е. необходимо указать размерность массива с помощью оператора DIM (размер):

DIM <имя> (размер)

В одном операторе DIM могут быть указаны несколько массивов через запятую, например:

20 DIM A(30,40), B(25)

Контрольные вопросы

1. Рассказать о двух типах переменных с индексами.
2. Что может являться индексом?
3. Какой командой описываются массивы?
4. Каким будет результат работы программы:

```
10 REM МАССИВ
20 DIM C(100):REM ВВОД МАССИВА
30 PRINT "ВВЕДИТЕ N <=100"
40 INPUT N
50 FOR I=1 TO N
60 INPUT "ЗНАЧЕНИЕ ЭЛЕМЕНТА МАССИВА";C(I)
70 NEXT I
80 S=0
90 FOR I=1 TO N
100 S=S+C(I)
110 NEXT I
120 PRINT "V=";S/(N+1)
130 END
```

если N=4, C(1)=5, C(2)=10, C(3)=8, C(4)=14.

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет по практической работе 5 должен содержать:

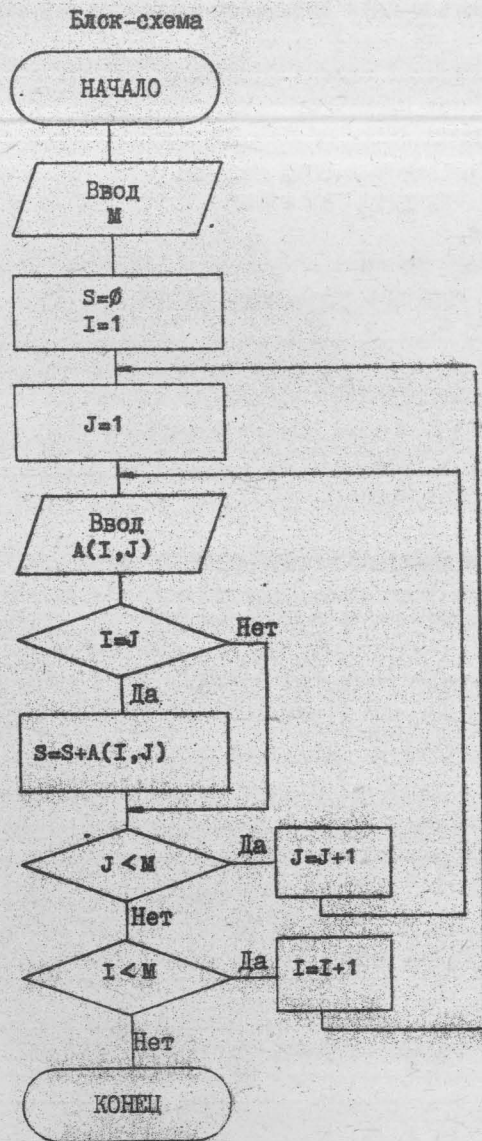
- название и цель работы;
- ответ на вопрос 4;
- запись общего вида команды описания массива;
- условие задания 1 или 2, блок-схему алгоритма, составленную программу на алгоритмическом языке и на языке Бейсик, результат ее решения.

Задание 1. Составить программу ввода двумерной таблицы чисел. Число строк и столбцов задается пользователем. Каждый вводимый элемент печатается.

Задание 2. Составить программу для подсчета среднесуточной температуры. Замеры температуры делаются через 4 ч. Значения температур

составляют таблицу.

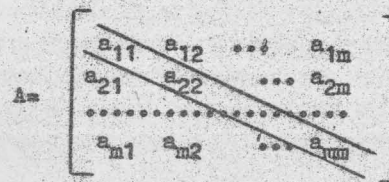
Пример выполнения задания. Составить программу нахождения суммы элементов главной диагонали квадратной матрицы $A(m \times m)$.



Программа на алгоритмическом языке:

```

алг сумма (вещ таб а [1:M,1:M], вещ S)
  арг а
  рез S
  нач цел i, j
    s:=0
    i:=1
    пока i <= M
      ни
        j:=1
        пока j <= M
          ни
            если i=j
              то s:=s+A [i,j]
              j:=j+1
            все
          ни
        i:=i+1
      ни
    кон
  
```



Элементы главной диагонали:

$a_{11}, a_{22}, a_{33}, a_{44}, \dots, a_{mm}$

Программа на языке Бейсик:

```

10 REM СУММА ЭЛЕМЕНТОВ ДИАГОНАЛИ
15 REM МАТРИЦЫ
20 INPUT "ВВЕДИТЕ КОЛИЧЕСТВО ЭЛЕМЕНТОВ МАТРИЦЫ";M
30 DIM A(M)
40 S=0
50 FOR I=1 TO M
60 FOR Y=1 TO M
70 INPUT A(I,Y)
80 IF I=Y THEN S=S+A(I,Y)
90 NEXT Y
100 NEXT I
110 PRINT "СУММА=";S
120 END
    
```

ПРАКТИЧЕСКАЯ РАБОТА 6

Тема: Алгоритм, использующий вспомогательный алгоритм, выполнение его на ПЭВМ.

Цель работы: Ознакомление со стандартными вспомогательными алгоритмами, программами и их использованием.

Продолжительность работы - один час.

Теоретические сведения. Наиболее часто используемыми вспомогательными алгоритмами являются стандартные математические функции, приведенные в табл. 2.

Т а б л и ц а 2

Имя функции	Выполняемое действие
SIN(X) ¹	Вычисляет синус аргумента X
COS(X)	Вычисляет косинус аргумента X
TAN(X)	Вычисляет тангенс аргумента X
ATN(X)	Вычисляет арктангенс аргумента X

¹ Для тригонометрических функций аргумент X задается в радианах.

Имя функции	Выполняемое действие
SQR(X)	Вычисляет положительный квадратный корень аргумента X
INT(X)	Определяет целую часть аргумента X
ABS(X)	Определяет абсолютное значение аргумента X
EXP(X)	Вычисляет показательную функцию e^X ($e=2,71228$; $x < = 37$)
LOG(X)	Вычисляет натуральный логарифм аргумента X
SGN(X)	Принимает значения: +1 при $X > 0$ -1 при $X < 0$ 0 при $X = 0$
RND(X)	Принимает псевдослучайное значение в интервале от 0 до 1. При $X > 0$ - элемент случайной последовательности. При $X < 0$ - запуск псевдослучайной последовательности (для каждого X). При $X=0$ - повторение последнего результата

Для обращения к функции необходимо набрать имя функции и аргумент, заключенный в скобки. Функции используются в выражениях так же, как константы и переменные.

Функция вычисляет результат и возвращает его значение. Интерпретатор продолжает вычислять значение выражения, как если бы вместо функции указали результат ее вычисления.

В тригонометрических функциях аргумент X задается в радианах. Пользователю предлагается использовать следующую формулу преобразования:

$$\text{значение в радианах} = \frac{\text{значение в градусах} \cdot 3,1416}{180}$$

Кроме встроенных математических функций в качестве вспомогательных алгоритмов могут выступать подпрограммы. Применение подпрограмм целесообразно при составлении сложных программ, когда определенный фрагмент программы используется неоднократно.

Переход на подпрограмму осуществляется оператором GO TO SUBroutine (идти к подпрограмме):

GOSUB <номер строки >

Выполняя этот оператор, ПЭВМ запоминает "адрес возврата" – номер строки, следующий за данным оператором GOSUB, и приступает к выполнению операторов, записанных со строки с указанным в GOSUB номером. Только встретив оператор RETURN (возвратиться), ПЭВМ возвращается назад к запомненному адресу. Таким образом, оператор RETURN служит признаком конца подпрограммы.

Одна подпрограмма может вызвать другую, другая – третью и т.д.

Передачу значений параметров подпрограмме и передачу результатов работы подпрограммы главной программе должен обеспечивать программист.

Контрольные вопросы

1. Какие вспомогательные алгоритмы используются в языке Бейсик?
2. Для каких целей необходимы операторы GOSUB и RETURN?
3. В чем отличие операторов GOSUB и GOTO?
4. Что напечатает программа:

```

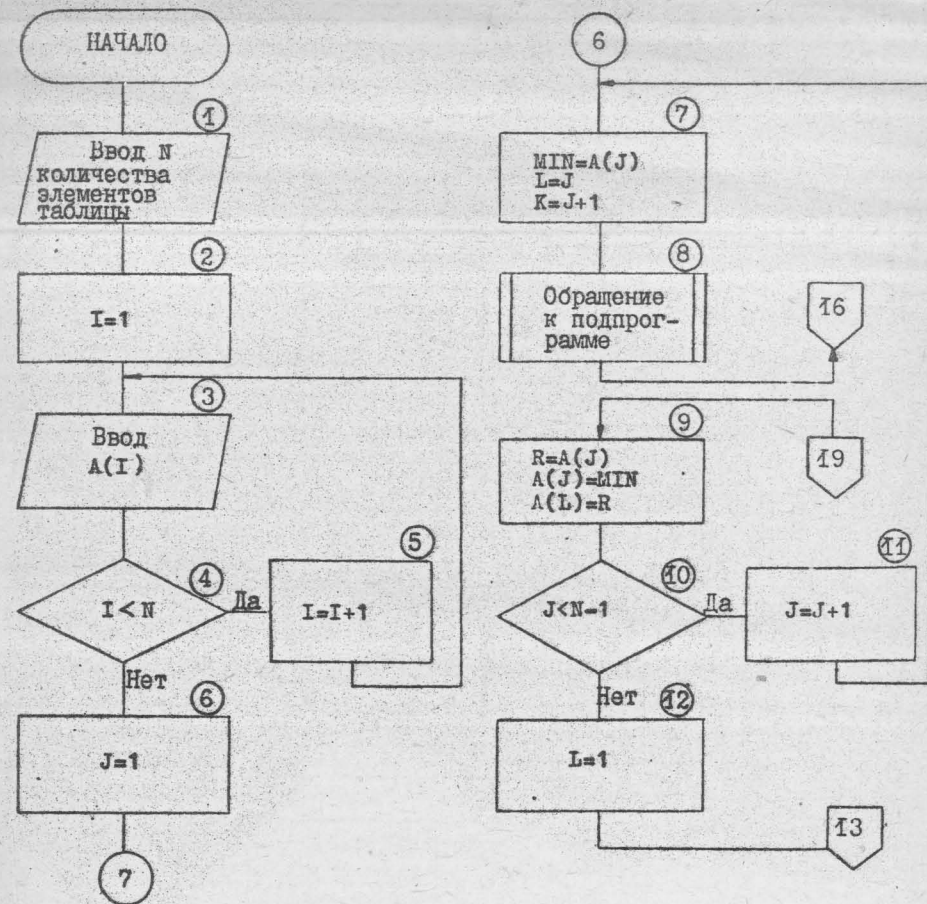
10 PRINT "ЖИВУТ В РИГЕ"
20 GOSUB 100
30 PRINT "Я СНОВА ДОМА"
40 END
100 PRINT "ПРИЕХАЛ В МОСКВУ"
110 GOSUB 200
120 PRINT "ВОЗВРАТИЛСЯ В РИГУ"
130 RETURN
200 PRINT "БЫЛ В КОМАНДИРОВКЕ"
210 RETURN
    
```

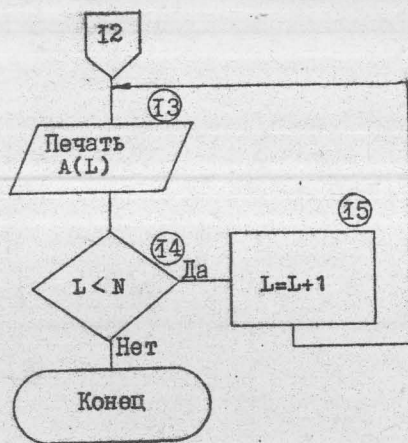
Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

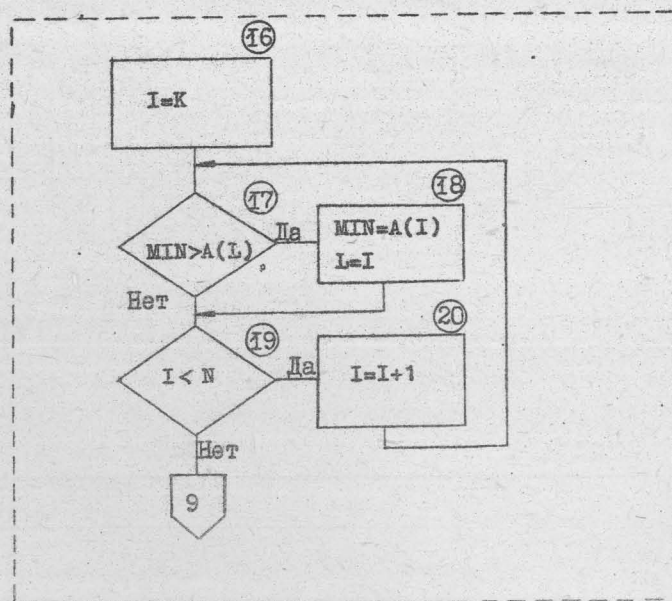
Отчет по практической работе 6 должен содержать:

- название и цель работы;
- таблицу встроенных математических функций;
- запись общего вида оператора перехода на подпрограмму и оператора возврата в основную программу;





Подпрограмма нахождения минимального элемента таблицы



- условие задания 1 или 2, блок-схему алгоритма, составленную программой на алгоритмическом языке и на языке Бейсик, и результат ее работы.

Задание 1. Написать программу для вычисления числа сочетаний из N элементов по M . Формула для расчета: $C_n^m = \frac{n!}{m!(n-m)!}$. Для вычисления факториала использовать подпрограмму. N и M задаются пользователем.

Задание 2. Написать программу, которая для данных l, m, n составляет таблицу значений функции $f(x) = (\sum_{j=1}^n \sqrt{x^2 + jx})^2 + (\sum_{i=1}^m \sqrt{x^2 - ix})$, где $x = 0, 1, 2, \dots, l$. Значение функции для каждого x вывести на печать. l, m, n задаются пользователем. Вычисление суммы оформить подпрограммой.

Пример выполнения задания. Упорядочить в порядке возрастания линейную таблицу чисел. Поиск минимального элемента таблицы оформить в виде подпрограммы.

Программа на алгоритмическом языке:

алг УТЧ (цел N , вещ таб $A[1:N]$)

арг N, A

рез A

нач цел J, L вещ MIN, R

$J := 1$

пока $J < N$

ни

$MIN := A[J]$

$L := J$

$K := J + 1$

МИНЭЛЕМЕНТ (K, N, L, A)

$R := A[J]$

$A[J] := MIN$

$A[L] := R$

$J := J + 1$

кц

кон

алг МИНЭЛЕМЕНТ (цел K, N, L , вещ таб $A[K:N]$)

арг A, K, N

рез L

```

нач цел I, вещ MIN
  I=K
  пока I < N
  ни
    если MIN > A[I]
      то
        MIN:=A[I]
        L:=I
      все
    I=I+1
  кн
кон

```

Программа на языке Бейсик:

```

10 REM УПОРЯДОЧЕНИЕ ЛИНЕЙНОЙ ТАБЛИЦЫ ЧИСЕЛ
20 REM В ПОРЯДКЕ ВОЗРАСТАНИЯ (С ОБРАЩЕНИЕМ
30 REM К ПОДПРОГРАММЕ)
35 INPUT "ВВЕДИТЕ ЧИСЛО ЭЛ-ОВ ТАБЛИЦЫ N ";N
40 DIM A(N)
50 FOR I=1 TO N:INPUT A(I) : NEXT I
60 FOR J=1 TO N-1
70 MIN=A(J): L=J
80 K=J+1
90 GOSUB 160
100 R=A(J)
110 A(J)=MIN
120 A(L)=R
130 NEXT J
140 FOR L=1 TO N : PRINT A(L): NEXT L: REM ПЕЧАТЬ РЕЗУЛЬТАТА
150 END
160 REM ПОДПРОГРАММА ПОИСКА
161 REM МИНИМАЛЬНОГО ЭЛЕМЕНТА
162 REM ТАБЛИЦЫ
170 FOR I=K TO N
180 IF MIN <=A(I) THEN 200
190 MIN=A(I): L=I
200 NEXT I
210 RETURN

```

ПРАКТИЧЕСКАЯ РАБОТА 7

Тема: Решение задач из курса математики.

Цель работы: Приобретение навыков решения математических задач, закрепление пройденного материала.

Продолжительность работы - один час.

Теоретические сведения. При изучении школьных курсов математики, физики, химии часто приходится сталкиваться с решением уравнений. К решению уравнений сводится реализация многих практических задач. Решить уравнение с одной переменной - это значит определить количество его корней и вычислить значение этих корней с заданной точностью.

На практике широко используются приближенные методы решения уравнений. При этом прежде всего требуется умение анализировать уравнение по его виду, по возможности оценивать границы участков, содержащих корни, определять корни уравнения, т.е. определять наиболее тесные промежутки, содержащие по одному корню. Решения всех этих задач опирается на знание свойств элементарных функций:

- если непрерывная функция $f(x)$ имеет на концах отрезка значения разных знаков, то на этом отрезке она обязательно имеет корень;
- если к тому же непрерывная функция $f(x)$ только убывает (или только возрастает) на отрезке, то она имеет на этом отрезке единственный корень.

Итак, проверяя знаки значений функции $f(x)$ на концах различных отрезков из области определения функции, можно отделить корни уравнения $f(x)=0$ методом вычислений. Этот прием трудоемок, поэтому в данном случае эффективно применение ЭВМ. Пользуясь этим же вычислительным приемом, можно не только локализовать корни уравнения на отрезках заданной длины, но и уточнить корни с любой желаемой степенью точности, используя метод половинного деления, рассмотренный в [2, § 7].

Контрольные вопросы

1. С помощью какого неравенства можно определить, что непрерывная функция $f(x)$, определенная на некотором отрезке $[a, b]$, имеет значения разных знаков.

2. Когда следует прекратить процесс вычисления корня уравнения методом половинного деления, если задана точность вычисления ϵ ? Какое значение будет значением корня?

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет к практической работе 7 должен содержать:

- название и цель работы;
- ответы на вопросы 1 и 2;
- условие задания 1 или 2, составленную программу, результат

просчета контрольного примера.

Задание 1. Составить алгоритм и написать программу отделения корней уравнения $\cos x - \lg x = 0$, все корни которого находятся на отрезке $[1; 7]$. В результате все найденные координаты концов отрезков записать в таблицу значений.

Задание 2. Составить алгоритм и написать программу уточнения положительного корня уравнения $f(x) = \cos x - x^2$ на отрезке $[0; 1]$ методом половинного деления с точностью $\varepsilon = 0,05$.

ПРАКТИЧЕСКАЯ РАБОТА 8

Тема: Решение задач из курса физики.

Цель работы: Приобретение навыков решения физических задач с помощью вычислительных методов, закрепление пройденного материала.

Продолжительность работы - один час.

Теоретические сведения. Многие физические явления могут быть описаны с помощью математической модели. По исходным данным задачи, производя определяемые моделью вычисления, можно найти требуемый ответ. Вычисления, как это обычно и бывает в практических задачах, имеют довольно большой объем, поэтому их целесообразно решать, используя ЭВМ.

Рассмотрим вычислительные модели, не поддающиеся описанию по явной формуле, по которой можно сразу найти требуемые значения искомых величин. Вместо этого есть алгоритм, позволяющий приближенно вычислять эти значения. При этом вычислении промежутки времени, интервалы длины, участки поверхности разбиваются на мелкие доли, фрагменты. Вычисления производятся отдельно для каждого участка.

Важной задачей, возникающей в процессе физических исследований, является определение значений параметров по результатам измерений.

Зная линейную зависимость некоторого явления, выраженную формулой $y = \frac{1}{x} \cdot z$, и имея результаты измерения, занесенные в таблицу

z									
y									

по методу наименьших квадратов, рассчитываем угловой коэффициент искомой прямой по формуле

$$k = \frac{z_1 \cdot y_1 + z_2 \cdot y_2 + \dots + z_n \cdot y_n}{z_1^2 + z_2^2 + z_3^2 + \dots + z_n^2}$$

где z, y - значения из таблицы.

Если измерения проведены точно, то получаем

$$k = \frac{1}{x} \cdot \frac{(z_1^2 + z_2^2 + \dots + z_n^2)}{z_1^2 + z_2^2 + \dots + z_n^2} = \frac{1}{x}$$

т.е. формула дает ожидаемый результат.

Контрольные вопросы

1. Объяснить, как получают значения, заносимые в таблицу?
2. Как с помощью графика определить точность вычисления?

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет по практической работе 8 должен содержать:

- название и цель работы;
- ответы на вопросы 1 и 2;
- условия задания 1 или 2, составленную программу, результат

контрольного просчета, график отклонения экспериментальных точек от найденной прямой.

Задание 1. Составить алгоритм и написать программу нахождения толщины нити, если были сделаны 10 измерений длины участков карандаша, на который виток к витку намотана нить. Результаты измерений приведены в таблице.

Количество витков на участке	10	20	30	40	50	60	70	80	90	100
Длина участка, мм	3	7	10	13	16	19	23	26	29	32

Алгоритм строится с помощью разобранного выше метода.

Задание 2. Составить алгоритм и написать программу нахождения толщины листа бумаги. Результаты измерений, приведенные в нижеследующей таблице, получены путем складывания его в несколько раз и измерения толщины полученной стопки.

Число листов в стопке	26	52	104	208	416	832	1664	3328
Толщина стопки, мм	3,5	7,1	10,1	28,2	56,5	113,0	226,1	452,1

ПРАКТИЧЕСКАЯ РАБОТА 9

Тема: Работа с графической информацией на ПЭВМ.

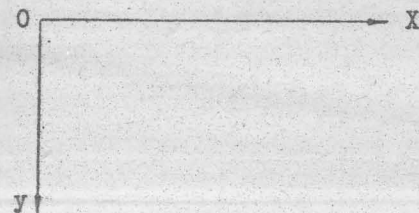
Цель работы: Ознакомление с общими принципами вывода графической информации на ПЭВМ, основными операторами графики в языке Бейсик.

Продолжительность работы - один час.

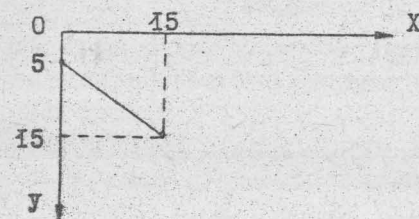
Теоретические сведения. Если графическая задача решается с помощью ПЭВМ "Агат", то графическая информация выводится на экран ВКУ. В отличие от алгоритмического языка, в котором существуют команды "Назад", "Вперед", "Вправо", "Влево", "Рисуй", "Не рисуй", в языке Бейсик существует лишь один оператор, заменяющий все вышеперечисленные, - PLOT (график):

PLOT <координата1> [TO <координата2>]

Часть оператора, заключенная в квадратные скобки, не обязательна. Если в операторе PLOT указана только одна координата, то на экране ВКУ будет изображена точка, место расположения которой определяется значением <координаты>. В отличие от системы координат, применяемой в математике, в языке Бейсик система координат имеет точку отсчета из верхнего левого угла.



Если в операторе PLOT указать две координаты, то будет начерчена прямая линия от точки с <координатой1> до точки с <координатой2>. Под координатой понимается пара значений, соответствующих положению точки относительно осей X и Y, записанных через запятую.



Например, по оператору PLOT 0,5 TO 15,15 на экране будет начерчена линия, как показано на рисунке.

Следует отметить, что работа программы с графикой начинается с включением графического режима. В языке Бейсик-Агат предусмотрено три графических режима (табл. 3).

Т а б л и ц а 3

Обозначение графического режима	Назначение графического режима	Диапазоны допустимых значений координат	Диапазоны номеров страниц
GR	Включение графики низкого разрешения (64x64)	0 + 63	2 + 31
MGR	Включение графики среднего разрешения (128x128)	0 + 127	1 + 7
HGR	Включение графики высокого разрешения (256x256)	0 + 255	1 + 7

Команда включения графического режима имеет вид:

GR= <N> . или MGR= <N> , или HGR= <N> ,

где N - номер страницы графического режима.

Чтобы графическая информация была видна на экране, задается цвет точки или линии, которую мы рисуем, с помощью оператора COLOR (цвет)
COLOR= <номер цвета >

Соответствие номеров цвету:

1 - красный	5 - фиолетовый
2 - зеленый	6 - голубой
3 - желтый	7 - белый
4 - синий	8 - черный

В графике низкого разрешения (GR) возможно только черно-белое отображение, а в графике среднего и высокого разрешения - цветное отображение.

Оператор PLOT может чертить не только прямую, но и ломаную линию. Ограничивает оператор PLOT размер строки - 255 символов.

Например, чтобы начертить квадрат красного цвета, нужно составить программу:

1Ø MGR=3

2Ø REM ЧЕРТИТЬ КВАДРАТ

3Ø COLOR=1 : REM ЦВЕТ - КРАСНЫЙ

4Ø PLOT 1,1 TO 1,127 TO 127, 127 TO 127,1 TO 1,1

5Ø TEXT=15 : REM ВЫХОД В ТЕКСТОВЫЙ РЕЖИМ

Последним оператором в этой программе является оператор TEXT (текст):

TEXT=<номер страницы > ,

устанавливающий текстовый режим. Диапазон изменения номеров страниц 2 + 31.

Соединение текстовой и графической страницы недопустимо. Включение графического режима всегда очищает экран, однако после перехода к текстовому режиму этого не происходит, т.е. текстовая информация наслаивается на предыдущую. При использовании двух режимов работы (текстового и графического), а также в случаях, когда количество выводимой информации превышает 32 строки экрана, возникает вопрос: как задержать информацию на экране? Ведь если не будет предусмотрена задержка каждой страницы выводимой информации, то она промелькнет очень быстро, и мы увидим лишь последнюю страницу. Для этой цели служит оператор GET (извлечь):

GET <имя литерной переменной >

Дойдя до этого оператора, выполнение программы приостанавливается и ПЭВМ ждет сигнала с клавиатуры. Для подачи этого сигнала нужно нажать на любую клавишу. Введенный символ на экране не отображается. После ввода символа выполнение программы продолжится.

Контрольные вопросы

1. В какой последовательности должны располагаться операторы графического режима?
2. Как начертить ломаную линию?
3. Что будет выведено на экран ВКУ в результате работы следующей программы:

1Ø MGR=3

2Ø PLOT 1Ø, 1Ø TO 1Ø, 4Ø

3Ø COLOR=2

4Ø PLOT 1Ø, 4Ø TO 4Ø, 4Ø TO 4Ø, 1Ø TO 1Ø, 4Ø

4. В каких случаях используется оператор GET?

Порядок выполнения практической работы:

- ознакомиться с теоретическими сведениями;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет по практической работе 9 должен содержать:

- название и цель работы;
- ответ на вопрос 3;
- таблицу графических режимов и таблицу цветов;
- запись общего вида операторов включения текстового и графического режимов, COLOR, PLOT и GET.

Задание 1. Написать программу вывода в графическом режиме MGR трех вложенных друг в друга квадратов разного цвета.

Задание 2. Написать программу вывода в графическом режиме HGR двух треугольников разного цвета, расположенных в виде песочных часов.

Пример выполнения задания. Написать программу вывода в графическом режиме MGR квадрата, заштрихованного внутри красным цветом.

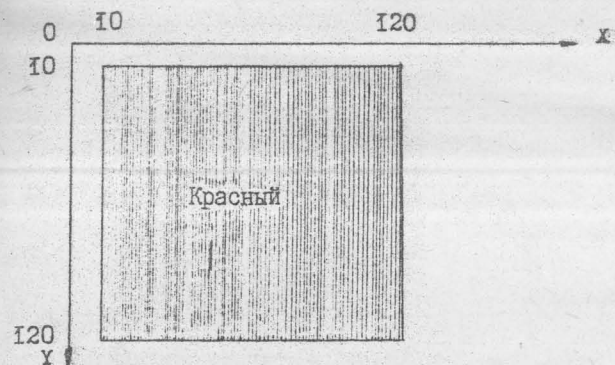
Программа на алгоритмическом языке:

```
алг квадрат (вещ N)
  арг N
нач цел I
  I:=0
  не рисуй
  вперед (10); направо (90); вперед (10)
  налево (90)
  рисуй
  пока I ≤ (N-10)/2
  ни
    вперед (N); направо (90); вперед (1)
    направо (90); назад (N); налево (90);
    вперед (1); налево (90)
    I:=I+1
  ки
кон
```

Программа на языке Бейсик:

```
10 MGR=3
20 COLOR=1 : REM ЦВЕТ - КРАСНЫЙ
30 INPUT N
40 IF N > 127 THEN PRINT " ЗНАЧЕНИЕ N БОЛЬШЕ ДОПУСТИМОГО ЗНА-
  ЧЕНИЯ КООРДИНАТЫ"
50 FOR I=10 TO N
60 PLOT I,10 TO I,N
70 NEXT I
80 END
```

Квадрат будет закрашиваться сверху вниз.
Результат работы программы при N=120:



ПРАКТИЧЕСКАЯ РАБОТА 10

Тема: Знакомство с примерами обучающих и игровых программ.

Цель работы: Ознакомление с построением алгоритмов простых обучающих и игровых программ.

Продолжительность работы - один час.

Теоретические сведения. В связи с распространением вычислительной техники появляется возможность обучения с помощью ЭВМ. Это освобождает преподавателя от лишних нагрузок и дает возможность одновременно с обучением контролировать усвоение материала. Такие программы называются обучающе-контролирующими. Но есть и программы, проводящие опрос по определенному курсу и оценивающие знания отвечающего. Такие программы называются контролирующими.

Рассмотрим наиболее простой пример контролирующей программы по физике.

Программа на алгоритмическом языке:

```
алг контролирующая программа по физике
  (лит у, х, а, z, цел s)
  арг х, а, z, s
  рез у
нач s:=0
  х:= "назовите единицу измерения силы тока"
  а:= "ответ"
  пока а ≠ "ампер" или а ≠ "А"
  ни
    z:= "неверно! попробуйте еще раз"
```

```

s:=s+1
a:= "ответ"
КЦ
z:= "правильно!"
x:= "назовите единицу измерения напряжения"
a:= "ответ"
ПОКА a# "вольт" или a# "В"
  НЦ z:= "неверно! попробуйте еще раз!"
  s:=s+1
  a:= "ответ"
КЦ
z:= "правильно!"
ЕСЛИ s=0
  ТО y:= "удовлетворительно!"
  ИНАЧЕ y:= "неудовлетворительно! Вам необходимо выучит материал"

```

Все

Все

кон

Программа на языке Бейсик:

```

10 REM КОНТРОЛИРУЮЩАЯ ПРОГРАММА ПО ФИЗИКЕ
15 S=0
20 PRINT "НАЗОВИТЕ ЕДИНИЦУ ИЗМЕРЕНИЯ СИЛЫ ТОКА:"
30 INPUT AX
40 IF AX = "АМПЕР" OR AX="А" THEN 70
50 PRINT "НЕВЕРНО! ПОПРОБУЙТЕ ЕЩЕ РАЗ"
60 S=S+1 : GOTO 20
70 PRINT "ПРАВИЛЬНО!"
80 PRINT "НАЗОВИТЕ ЕДИНИЦУ ИЗМЕРЕНИЯ НАПРЯЖЕНИЯ:"
90 INPUT AX
100 IF AX= "ВОЛЬТ" OR AX="В" THEN 125
110 PRINT "НЕВЕРНО! ПОПРОБУЙТЕ ЕЩЕ РАЗ!"
120 S=S+1 : GOTO 80
125 PRINT "ПРАВИЛЬНО!"
130 IF S=0 THEN PRINT "ОЦЕНКА - ОТЛИЧНО!"
140 IF S=1 THEN PRINT "УДОВЛЕТВОРИТЕЛЬНО!"
150 IF S >= 2 THEN PRINT "НЕУДОВЛЕТВОРИТЕЛЬНО! ВАМ НЕОБХОДИМО ВУЧИТЬ МАТЕРИАЛ!"
160 END

```

Переменная S служит для подсчета количества ошибок при ответах. Начиная с оператора 130 и кончая оператором 150, анализируется счетчик ошибок и выставляется оценка экзаменуемому.

Конечно, это наиболее простой пример контролирующей программы, но он дает представление об алгоритме построения аналогичных программ.

Математические игры занимательны и интересны не только детям, но и взрослым. Они развивают способности, необходимые для решения инженерных задач, проблем управления и экономики. Такие игры более интересны по сравнению с играми, в которых необходима лишь быстрая реакция играющего.

Приведем пример наиболее простой математической игры. Это игра Баше. Алгоритм ее приведен в [2, с. 19,28]. Алгоритм выигрыша первого игрока имеет следующий вид:

- 1) взять два предмета;
- 2) второй и последующий ходы делать так, чтобы количество предметов, взятых вместе с соперником за очередной ход, в сумме составляло 4.

Количество предметов определяется формулой $7+4*N$ ($N=0,1,2,\dots$).

Программа на алгоритмическом языке:

```

ал игра Баше
нач
  взять два предмета
  пока осталось больше четырех предметов
  НЦ
    дать противнику сделать ход
    запомнить число x взятых противником предметов
    взять 4 - x предметов
  КЦ
  дать противнику сделать ход

```

кон

Программа на языке Бейсик:

```

10 REM ИГРА БАШЕ
20 REM КОМПЬЮТЕР ХОДИТ И ВЫИГРЫВАЕТ
30 PRINT "ХОТИТЕ СЫГРАТЬ В ИГРУ БАШЕ С КОМПЬЮТЕРОМ?"
40 PRINT : INPUT AX
50 IF AX = "НЕТ" THEN 200
60 PRINT : PRINT "ВВЕДИТЕ ЦЕЛОЕ ЧИСЛО"
70 PRINT : INPUT N

```



```

80 N=7+4*N : REM N - КОЛ-ВО ПРЕДМЕТОВ
90 PRINT : PRINT "ЕСТЬ"; N; "ПРЕДМЕТОВ"
100 PRINT : PRINT "ЗА ОДИН ХОД МОЖНО ВЗЯТЬ 1,2"
110 PRINT : PRINT "ИЛИ 3 ПРЕДМЕТА. ПРОИГРЫВАЕТ ТОТ",
120 PRINT : PRINT "КТО БЕРЕТ ПОСЛЕДНИЙ ПРЕДМЕТ"
130 PRINT : PRINT "БУДЕМ ХОДИТЬ ПО ОЧЕРЕДИ!"
140 PRINT : PRINT "Я НАЧИНАЮ! МОЙ ПЕРВЫЙ ХОД : 2"
150 Y=2 : REM Y - ХОД КОМПЬЮТЕРА
160 N=N-Y
170 IF N=1 THEN 270
180 PRINT : PRINT "ОСТАЛОСЬ "; N; " ПРЕДМЕТОВ"
190 PRINT : PRINT "ВАШ ХОД?"
200 PRINT : INPUT X : REM X- ХОД ИГРОКА
210 IF X<1 OR X>3 THEN 260
220 N=N-X
230 PRINT : PRINT "ПОСЛЕ ВАШЕГО ХОДА ОСТАЛОСЬ ";
N; "ПРЕДМЕТОВ" : Y=4-X
240 PRINT : PRINT "МОЙ ХОД "; Y
250 GOTO 160
260 PRINT : PRINT "НЕВЕРНЫЙ ХОД!": GOTO 190
270 PRINT : PRINT "ОСТАЛСЯ 1 ПРЕДМЕТ. ВЫ ПРОИГРАЛИ..."
280 END

```

Оператор PRINT без параметров задается для пропуска пустой строки, так как строки на экране ВКУ располагаются очень близко друг к другу и плохо воспринимаются при чтении.

Контрольные вопросы

1. Как строится алгоритм контролирующей программы?
2. Почему операторы 100 - 140 в программе игры Баше записаны не как один оператор?
3. В операторе с каким номером осуществляется контроль за ответом игрока?
4. Что произойдет, если игрок сделает неверный ход, играя с компьютером по приведенной программе игры Баше?
5. Какой ответ и в какой форме должен дать испытуемый ученик, работающий с контролирующей программой по физике, когда появится первый знак вопроса?

6. Что будет в счетчике ошибок при работе с контролирующей программой по физике, если Вы дважды на один и тот же вопрос ответите неверно? Какой будет оценка в этом случае при условии, что на второй вопрос был дан сразу правильный ответ?

Порядок выполнения практической работы:

- разобрать программы, приведенные в теоретической части;
- ответить на контрольные вопросы;
- выполнить задание 1 или 2 по вариантам;
- оформить отчет.

Отчет по практической работе 10 должен содержать:

- название и цель работы;
- тексты программ, приведенных в работе на алгоритмическом языке и на языке Бейсик, результат работы программы.

Задание 1. Ввести и отладить контролируемую программу по физике, приведенную в работе. Поработать с ней.

Задание 2. Ввести и отладить программу игры Баше, приведенную в работе. Поработать с ней.

ПРАКТИЧЕСКАЯ РАБОТА 11

Тема: Зачетная работа.

Цель работы: Проверка знания учащимися пройденного материала.

Продолжительность работы - 2 ч.

Теоретические сведения. Зачетная работа проводится для оценки знаний учащихся и выставления итоговой оценки за практический курс.

В зачетную работу включены как теоретические вопросы, так и практическое задание.

Порядок выполнения практической работы:

- выполнить задание зачетной работы по вариантам;
- оформить отчет.

Отчет по практической работе 11 должен содержать:

- название и цель работы;
- номер варианта зачетной работы;
- номер вопроса и ответ на него.

ЗАДАНИЕ

Вариант I

1. Выбрать целые константы:

+15 -137. +1.0 125 0

2. Написать на языке Бейсик арифметические выражения:

а) $\frac{6,03}{\frac{1}{x_1} + \frac{1}{x_2} + \frac{1}{x_3}}$; б) $(\operatorname{tg} x + 1)$; в) $1 - \cos^2 4x$

3. Какой вид будет иметь строка результатов при печати перемен-

ных:

10 A=2.6 : B=-11.25: C=8.4 : FX ="РЕЗУЛЬТАТ"

60 PRINT FX

70 PRINT A,,B;

80 PRINT C

4. Задан массив A (7). Присвоить значения элементам массива, используя оператор цикла.

Вариант II

1. Выбрать правильную запись вещественной константы:

137.1A 1.03E+15 -1E15 -0.01E2.5 137

2. Написать на языке Бейсик арифметические выражения:

а) $\frac{(a+b)^2}{2,3x+1}$; б) $\cos 2\alpha \cdot \sin^2 \beta \cdot \cos \sqrt{2\alpha + \beta}$

3. Какие значения будут присвоены переменным в результате работы программы:

10 A=SQR(16)

20 B=8A+17

30 S=A+8B

4. Составить программу, выводящую на экран в режиме средней графики слиям цветом прямоугольник пунктирными линиями



Вариант III

1. Какие из приведенных констант не могут использоваться в языке Бейсик:

"ПРОГРАММА"; 6.25; "GAMMA"; -63,856; 0,34E-4;

"1234"; 6.7.8; A*B^A^C; -0.01E-2.5

2. Записать на языке Бейсик арифметические выражения:

а) $\frac{1}{2} \sqrt{b^2 + c^2 + 2bc \cos \alpha}$; б) $\frac{x_1 + x_2 + x_3}{30}$

3. Возможна ли такая запись:

10 MGR=3

20 COLOR=7

30 PLOT 3,4

40 PLOT TO 10,26

Если возможна, то что будет изображено на экране?

4. Задан массив A (10). Составить программу нахождения наименьшего элемента массива и распечатать весь массив и наименьший элемент отдельно: MIN=...

Вариант IV

1. Записать числа на языке Бейсик:

0,5; 8,374; 3,172589312; $1,003 \cdot 10^5$; 0,000000987; $74 \frac{3}{4}$

2. Записать на языке Бейсик арифметические выражения:

$1 + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{3}} + \frac{1}{\sqrt{4}}$; $\frac{A^2 \sin A \cdot \sin B}{2 \sin^2 A}$; $\operatorname{tg} \frac{x}{2}$

3. Какую фигуру мы увидим на экране в результате работы программы:

10 MGR=3

20 PLOT 40,100 TO 65,30 TO 90,100 TO 40,100

4. Составить программу, задающую массив A (5), состоящий из последовательности чисел натурального ряда, используя оператор цикла. Просуммировать все элементы массива и найти среднее арифметическое.

Вариант V

1. Найти ошибки в записи чисел на языке Бейсик и написать их верно:

-2; .01; +31,5; -2835130.128; 18.2E; 245.0; 125,3E10

18.24287-E5; -18E0981

2. Записать на языке Бейсик арифметические выражения:

а) $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$; б) $\frac{3 \sin^2 5x}{2 \cos(3x)^2}$

3. Будет ли выполняться оператор PRINT:

```
20 A=5 : GOTO 40
30 PRINT A
40 INPUT N
```

4. Составить программу, выводящую на экран ВКУ в графическом режиме среднего разрешения красную пятиконечную звезду.

Вариант VI

1. Записать числа на языке Бейсик:

```
2 7/8; 0,14193; -0,002726; -6,47 * 10^-4; 7, ;
4 * 10^27; 2,34455628921
```

2. Записать на языке Бейсик арифметические выражения:

а) $x:y \cdot a$; б) $1 + \frac{a+b \cdot c}{1+2,3 \cdot d}$; в) $\operatorname{tg} \frac{x}{2}$.

3. Сократить запись программы:

```
10 MGR=3
20 COLOR=2
30 PLOT 10,10 TO 40,10
40 PLOT 10,50 TO 40,50
50 PLOT 10,10 TO 10,50
60 PLOT 40,50 TO 40,10
```

4. Составить программу нахождения нечетных чисел среди ряда натуральных чисел от 1 до 30 и вывода их на экран ВКУ.

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ
К ПРАКТИЧЕСКИМ РАБОТАМ ДЛЯ 9-ГО КЛАССА

Номер практической работы	Номер вопроса	Ответ											
1	5	<table border="1" style="display: inline-table; text-align: center;"> <tr> <td>3</td><td>7</td><td>5</td><td>+</td><td>PEE</td><td>I</td><td>8</td><td>s</td><td>/</td><td>3</td><td>R</td> </tr> </table> <p style="text-align: center;">Одновременно</p>	3	7	5	+	PEE	I	8	s	/	3	R
3	7	5	+	PEE	I	8	s	/	3	R			
2	6	ОНА ИДЕТ СО СКОРОСТЬЮ - 3 ЗА ВРЕМЯ - 40 МИН											
3	4	Программа будет печатать 2,4,16,65536,... до тех пор, пока очередное значение x не станет слишком большим - тогда произойдет переполнение, и программа закончит работу											
4	4	Число 495											
5	4	Число 7.4											
6	4	Киву в Риге приехал в Москву был в командировке возвратился в Ригу я снова дома											
7	1	$f(\alpha) \cdot f(\beta) < 0$											
7	2	Процесс будет продолжаться до тех пор, пока длина отрезка, содержащего корень, не станет меньше 2ϵ . Значением корня является середина отрезка											
8	1	Значения получают в результате измерений											
8	2	Если отклонение экспериментальных точек от найденной прямой мало, то вычисление произведено достаточно точно											
9	3	Прямоугольный треугольник зеленого цвета											

Номер практической работы	Номер вопроса	Ответ
10	2	Чтобы текст на экране ВКУ был в виде, более удобном для восприятия
10	3	210
10	4	Будет дано соответствующее сообщение (оператор 260) и предоставлена возможность сделать ход еще раз
10	5	АМПЕР или А (без кавычек)
10	6	S=2; неудовлетворительно

ПРАВИЛА РАБОТЫ УЧАЩИХСЯ В КАБИНЕТЕ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

1. В кабинете вычислительной техники установлена дорогостоящая, сложная и требующая осторожного и аккуратного обращения аппаратура - компьютеры (ПЭВМ), принтер и др. технические средства. Чтобы избежать ее повреждения, спокойно, не торопясь, не задевая столы, входите в кабинет и занимайте отведенное Вам место, ничего не трогая на столах.

2. На Вашем рабочем месте размещены составные части ПЭВМ - системный блок, клавиатура и ВКУ, которые работают под высоким напряжением. Неправильное обращение с аппаратурой, кабелями и ВКУ может привести к тяжелым поражениям электрическим током, вызвать загорание аппаратуры.

Поэтому строго запрещается:

- трогать разъемы соединительных кабелей;
- прикасаться к питающим проводам и устройствам заземления;
- прикасаться к экрану и тыльной стороне ВКУ, клавиатуры;
- включать и отключать аппаратуру без указания преподавателя;
- класть диск, книги, тетради на ВКУ и клавиатуру;
- работать во влажной одежде и влажными руками.

3. При появлении запаха гари немедленно прекратите работу, выключите аппаратуру и сообщите об этом преподавателю, в случае необходимости окажите ему помощь в тушении огня.

4. Перед началом работы:

- убедитесь в отсутствии видимых причин повреждений аппаратуры на рабочем месте;

- сядьте так, чтобы линия зрения приходилась в центр экрана, на расстоянии 60-70 см от него, чтобы, не наклоняясь, работать на клавиатуре и воспринимать передаваемую на экран ВКУ информацию;

- разместите на столе тетрадь и учебное пособие так, чтобы они не мешали работе на ПЭВМ;

- хорошо разберитесь в особенностях применяемого в работе оборудования;

- начинайте работу только по указанию преподавателя.

5. Во время работы:

- строго выполняйте все перечисленные выше правила, а также указания преподавателя; все операции при работе на ПЭВМ выполняйте в строгой последовательности, изложенной преподавателем. При необходи-

мости обращайтесь к преподавателю;

- следите за исправностью аппаратуры и немедленно прекращайте работу при появлении необычного звука или самопроизвольного отключения аппаратуры. Немедленно докладывайте об этом преподавателю;
- плавно нажимайте на клавиши, не допуская резких ударов;
- не пользуйтесь клавиатурой, если не подключено напряжение;
- работайте на клавиатуре чистыми руками;
- никогда не пытайтесь самостоятельно устранять неисправности в работе аппаратуры;
- не вставайте со своих мест, когда в кабинет входят посетители.

6. Работа на ПЭВМ требует большого внимания, четких действий и самоконтроля.

Поэтому нельзя работать:

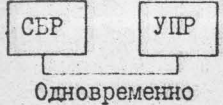

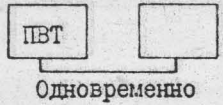
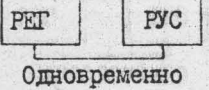
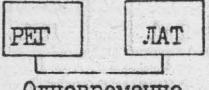

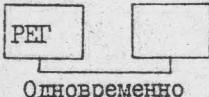
- при недостаточном освещении;
- при плохом самочувствии (в этом случае надо обратиться к врачу).






7. Вы должны хорошо знать и строго придерживаться настоящих правил, точно следовать указаниям преподавателя, чтобы:

- избежать несчастных случаев;
- успешно овладеть знаниями, умениями, навыками;
- сберечь государственное имущество - вычислительную технику и оборудование.

Вы отвечаете за состояние рабочего места и сохранность размещенного на нем оборудования. Невыполнение правил - грубейшее нарушение порядка и дисциплины.

КЛАВИШИ УПРАВЛЕНИЯ

Обозначение клавиши	Действие, вызываемое нажатием клавиши
	Прекращение выполнения любой программы и ввод ПЭВМ в определенное начальное состояние (обычно, в состоянии приглашения к диалогу, когда ПЭВМ ждет директив от пользователя, высвечивая слева на экране знак "I" и мигающий курсор)
	Ввод директив, набранных на клавиатуре, после чего ПЭВМ воспринимает набранную директиву и исполняет ее
	Повторная выдача символа, нажатого одновременно с клавишей ПВТ. Действие прекращается, когда отпускается одна из нажатых клавиш
	Переключение в русский набор букв
	Переключение в латинский набор букв
	Выдача на экран ВКУ знака "!" (нажатие одновременно с клавишей РЕГ любой из клавиш, содержащих цифры и специальные знаки, выдаст символ, находящийся на <u>нижнем</u> регистре)
	При фиксированном русском регистре выдаст символ на латинском регистре, соответствующий нажатой клавише, а при фиксированном латинском регистре - символ на русском регистре, соответствующий нажатой клавише

Обозначение клавиши	Действие, вызываемое нажатием клавиши
	Ввод в режим редактирования. Подробнее о работе клавиши см. в практической работе 1
	Перемещение курсора на одну строку вниз
	Перемещение курсора на одну строку вверх
	Перемещение курсора на одну позицию вправо
	Перемещение курсора на одну позицию влево

ЛИТЕРАТУРА

1. Основы информатики и вычислительной техники. Программа для средних учебных заведений. - М.: Просвещение, 1985.
2. Основы информатики и вычислительной техники. Пробное учебное пособие для средних учебных заведений. Ч. I / Под ред. А.П.Ершова, В.М.Монахова. - М.: Просвещение, 1985.
3. Основы информатики и вычислительной техники. Пробное учебное пособие для средних учебных заведений. Ч. II / Под ред. А.П.Ершова, В.М.Монахова. - М.: Просвещение, 1986.

ОГЛАВЛЕНИЕ

Общие положения	3
Методические рекомендации по проведению практических занятий	4
Практические занятия	6
Практическая работа 1. Знакомство с ПЭВМ	6
Практическая работа 2. Линейный алгоритм, выполнение его на ПЭВМ	11
Практическая работа 3. Разветвляющийся алгоритм (команды ветвления), выполнение его на ПЭВМ.	18
Практическая работа 4. Циклический алгоритм (команда повторения), выполнение его на ПЭВМ.	22
Практическая работа 5. Алгоритм, использующий табличные величины, выполнение его на ПЭВМ.	26
Практическая работа 6. Алгоритм, использующий вспомогательный алгоритм, выполнение его на ПЭВМ	30
Практическая работа 7. Решение задач из курса математики	37
Практическая работа 8. Решение задач из курса физики	38
Практическая работа 9. Работа с графической информацией на ПЭВМ.	40
Практическая работа 10. Знакомство с примерами обучающих и игровых программ	45
Практическая работа 11. Зачетная работа.	49
Приложения	53
Литература	59