**Алгоритмы работы с величинами**

**Компьютер как исполнитель алгоритмов**

Вам уже известно, что всякий алгоритм составляется для конкретного исполнителя. *Теперь в качестве исполнителя мы будем рассматривать компьютер, оснащенный системой программирования на определенном языке.*

Компьютер-исполнитель работает с определенными данными по определенной программе. Данные - это множество величин.

**Величины: константы и переменные**

Компьютер работает с информацией, хранящейся в его памяти. Отдельный информационный объект (число, символ, строка, таблица и пр.) называется величиной.

Всякая обрабатываемая программой величина занимает свое место (поле) в памяти ЭВМ. Значение величины - это информация, хранимая в этом поле памяти.

Существуют *три основных типа величин*, с которыми работает компьютер: ***числовой, символьный*** и ***логический***. Изучая базы данных и электронные таблицы, вы уже встречались с этими типами. В данной главе мы будем строить алгоритмы, работающие с числовыми величинами.

Числовые величины в программировании, так же как и математические величины, делятся на переменные и константы (постоянные). Например, в формуле (а2 - 2аb + b2) а, b - переменные, 2 - константа.

**Константы** записываются в алгоритмах своими десятичными значениями, например: 23, 3.5, 34. Значение константы хранится в выделенной под нее ячейке памяти и остается неизменным в течение работы программы.

**Переменные** в программировании, как и в математике, обозначаются символическими именами. Эти имена называют **идентификаторами** (от глагола "идентифицировать", что значит "обозначать", "символизировать"). Идентификатор может быть одной буквой, множеством букв, сочетанием букв и цифр и Т. д. Примеры идентификаторов: А, X, ВЗ, prim, r25 и т. п.

**Система команд**

Вам известно, что всякий алгоритм строится исходя из системы команд исполнителя, для которого он предназначен.

Независимо от того, на каком языке программирования будет написана программа, алгоритм работы с величинами составляется из следующих команд:

***- присваивание***;

***- ввод***;

***- вывод***;

***- обращение к вспомогательному алгоритму***;

***- цикл***;

***- ветвление***.

**Команда присваивания**

Команда присваивания - одна из основных команд в алгоритмах работы с величинами. Записывать ее мы будем так:

<переменная> := < выражение>

Значок ":=" читается "присвоить". Например:

Z :=X + Y

Компьютер сначала вычисляет выражение, затем результат присваивает переменной, стоящей слева от знака ":=".

Если до выполнения этой команды содержимое ячеек, соответствующих переменным X, Y, Z, было таким:

|  |
| --- |
| http://www.5byte.ru/9/images/alg6.gif |

то после выполнения команды оно станет следующим:

|  |
| --- |
| http://www.5byte.ru/9/images/alg7.gif |

Прочерк в ячейке Z обозначает, что начальное число в ней может быть любым. Оно не имеет значения для результата данной команды.

Если слева от знака присваивания стоит числовая переменная, а справа - математическое выражение, то такую команду называют ***арифметической командой присваивания***, а выражение - арифметическим.

В частном случае арифметическое выражение может быть представлено одной переменной или одной константой. Например:

Х:= 5

Y:=X

**Команда ввода**

Значения переменных, являющихся исходными данными решаемой задачи, как правило, задаются вводом.

Команда ввода в описаниях алгоритмов будет выглядеть так:

ввод <список переменных>.

Например:

ввод А, В, С

На современных компьютерах ввод чаще всего выполняется в режиме диалога с пользователем. По команде ввода компьютер прерывает выполнение программы и ждет действий пользователя. Пользователь должен набрать на клавиатуре вводимые значения переменных и нажать клавишу <ВВОД>. Введенные значения присвоятся соответствующим переменным из списка ввода, и выполнение программы продолжится.

Вот схема выполнения приведенной выше команды.

1. Память до выполнения команды:

|  |
| --- |
| http://www.5byte.ru/9/images/alg8.gif |

2. Процессор компьютера получил команду ввод А, B, С, прервал свою работу и ждет действий пользователя.

3. Пользователь набирает на клавиатуре:

1       3       5

и нажимает клавишу <ВВОД> (<Еnter>).

4. Память после выполнения команды:

|  |
| --- |
| http://www.5byte.ru/9/images/alg8.gif |

5. Процессор переходит к выполнению следующей команды программы.

При выполнении пункта 3 вводимые числа должны быть отделены друг от друга какими-нибудь разделителями. Обычно это пробелы.

Из сказанного выше можно сделать вывод:

Переменные величины получают конкретные значения в результате выполнения команды присваивания или команды ввода.

Если переменной величине не присвоено никакого значения (или не введено), то она является неопределенной. Иначе говоря, ничего нельзя сказать, какое значение имеет эта переменная.

**Команда вывода**

Результаты решения задачи сообщаются компьютером пользователю путем выполнения команды вывода.

**Команда вывода** в алгоритмах будет записываться так:

вывод <список вывода>

Например:

вывод X1, Х2

По этой команде значения переменных X1 и Х2 будут вынесены на устройство вывода (чаще всего это экран).

О других командах, применяемых в вычислительных алгоритмах, вы узнаете позже.

**Вопросы и задания**

1. Что такое величина? Чем отличаются переменные и постоянные величины?

2. Чем определяется значение величины?

3. Какие существуют основные типы величин в программировании?

4. Как записывается команда присваивания?

5. Что такое ввод? Как записывается команда ввода?

6. Что такое вывод? Как записывается команда вывода?

7. В схематическом виде (как это сделано в параграфе) отразите изменения значений в ячейках, соответствующих переменным А и В, в ходе последовательного выполнения команд присваивания:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1) | А:=1В:=2 А:=А+В В:= 2хА | 2) | А:=1В:=2С:=АА:=В | 3) | А:=1В:=2А:=А+ВВ:=А-ВА:=А-В |

8. Вместо многоточия впишите в алгоритм несколько команд присваивания, в результате чего должен получиться алгоритм возведения в 4-ю степень введенного числа (дополнительные переменные, кроме А, не использовать):

ввод А . . . вывод А