Урок 1. Знакомство со средой программирования Кумир. Понятие величины

Данная программная среда позволяет получить практические навыки построения и выполнения основных алгоритмических конструкций, применяемых в различных языках высокого уровня.

После загрузки среды КУМИР на экране можно увидеть следующее окно Рис 1

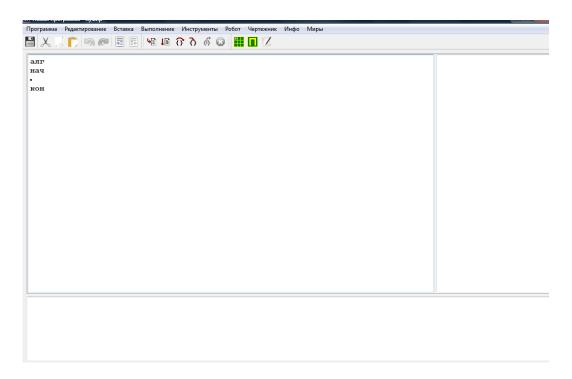
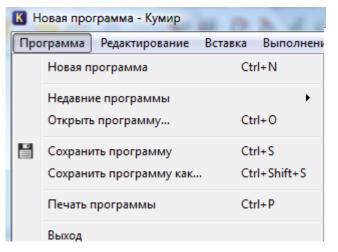


Рис 1 окно Новая программа Кумир



В этом окне видна структура будущего алгоритма.

Рис 2 Состав строки меню Программа

алг	название (заголовок)		
нач	тело алгоритма		

В верхней части экрана имеется строка меню Программа (рис. 2)

С помощью команд данного меню можно создавать новые программы, открывать ранее созданные, сохранять, переименовывать, выводить тексты программ на принтер.

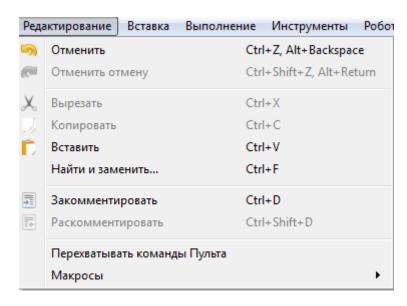


Рис 3 Состав команды редактирование

Следующий пункт основного меню Редактирование (Рис.3).

С помощью команд данного пункта меню производится редактирование текста алгоритма. Можно отменять последние действия, вырезать, копировать, перемещать удалять текстовые фрагменты. Если алгоритм достаточно большой, то поиск и замену фрагмента можно автоматизировать с помощью соответствующих команд **Найти и заменить**.

Вставка Выполнен	ие Инструме	нты Робот
алг-нач-кон	(ESC, A)	Esc, F
если-то-все	(ESC, E)	Esc, T
выбор-при-все	(ESC, B)	Esc, D
иначе	(ESC, VI)	Esc, B
нц-раз-кц	(ESC, P)	Esc, H
нц-для-кц	(ESC, Д)	Esc, L
нц-пока-кц	(ESC, Π)	Esc, G
нц-кц	(ЕЅС, Ц)	Esc, W
исп-кон_исп	(ESC, C)	Esc, C
вверх		Esc, Up
вправо		Esc, Right
вниз		Esc, Down
влево		Esc, Left
закрасить		Esc, Space
использовать Р	обот	Esc, 1
использовать Ч	использовать Чертежник	
использовать Ф	айлы	Esc, 3

Рис 4 Состав пункта Вставка

В раскрывающемся списки пункта Вставка (Рис.4)), можно найти структуру готовых конструкций, используемых в Кумире.

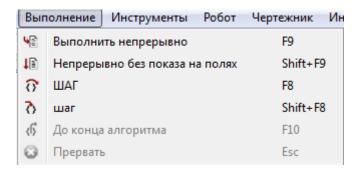
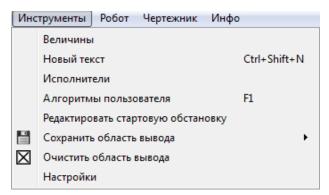


Рис 5 Состав пункта Выполнение

С помощью списка команд меню **Выполнение** (Рис. 5), можно просматривать работу готовых алгоритмов в различных режимах. Команда **Выполнить непрерывно** позволяет увидеть сразу готовый результат. Выполнение производиться с показом на полях промежуточных значений вычислений на специальном поле, которое расположено справа от основной рабочей области.

Выполнение по шагам необходимо в тех ситуациях, когда сложно найти ошибку в готовом алгоритме и надо просматривать каждый шаг последовательно для нахождения неверного действия.



В пункте меню **Инструменты** (Рис. 6), а также в двух последующих возможен выбор других исполнителей, которые в данной главе не рассматриваются

Рис 6 Состав пункта Инструменты

И последний пункт **Инфо** (Рис. 7) позволяет получить любую справку по работе в данной среде программирования.

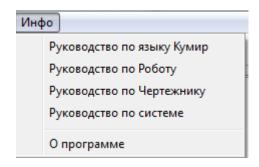
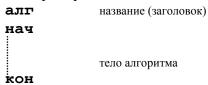


Рис 7 Состав пункта Инфо

Ниже пунктов меню можно видеть заготовку структуры будущего алгоритма, которой надо обязательно придерживаться.



Около служебного слова алг должно указываться имя алгоритма. Как правило, это название решаемой задачи. Все служебные слова выделяются жирным шрифтом. и располагаются строго друг под другом, остальные команды рекомендуется записывать правее.

Между слов нач и кон_размещаются команды для исполнителя.

Величина (имя, значение, тип)

Часто при составлении программ приходится сначала вычислять необходимые числовые параметры, а затем использовать их.

Для этих целей в алгоритмических языках используются Величины.

Этот термин взят из математики.

Величины, которые могут изменяться в процессе выполнения программы, называются переменными.

Величины, которые не изменяются - называются константами.

Любая величина имеет имя, значение и тип.

Общаясь между собой люди на вопрос «кто это?» или «что это такое?» отвечают одним словом, которое обозначает данный предмет или человека. Чтобы сообщить информацию о предмете или явлении, его нужно обозначить – дать ему имя.

Имя служит для обозначения величины. **Имя величины** образуется из одного или нескольких символов (букв латинского алфавита или цифр), причем первый символ должен быть обязательно буквой.

Например: A, ANC, LI2, A1, SDY, VFR57, SUMMA

Если для человека имя связано объектом, то для компьютера это определенная ячейка оперативной памяти. При трансляции составляется таблица, в которой каждому имени ставится в соответствие конкретный числовой адрес ячейки памяти.

Каждая величина должна иметь **значение.** С помощью значения величины можно сравнивать и отмечать изменения, которые с ними происходят, например, изменение веса, места жительства, количество отсутствующих на уроке, изменение цвета.

Значение величин выражаются числами или словами, символами. Например, длину можно определить числом, а цвет крыши— словом. Соответственно переменные и константы бывают числовыми и символьными.

Значение символьной величины записывается в кавычках. Так, например, если мы видим такую запись в программе: **123**, то понимаем, что это числовая величина, а если написано так: "**123**", то это величина символьная.

Над числовыми величинами можно производить различные математические операции. С символьными величинами математических операций производить нельзя. Но некоторые специальные действия с ними выполняют, а какие именно разберем позднее.

Числовые величины могут быть целыми и вещественными.

Действия над целыми числами всегда выполняются в компьютере точно, а над вещественными, как правило, приближенно. Кроме того, целые числа в памяти компьютера при хранении занимают меньше места, следовательно, полезно иметь

возможность выделить в программе те величины, которые точно будут целыми, и сэкономить на этом ресурсы компьютера.

Вещественные переменные имеют две формы представления: основную и экспоненциальную.

Основная форма отличается от обычной записи дробных десятичных чисел только тем, что целая и дробная части отделяются точкой, а не запятой.

При выводе целых и вещественных чисел в основной форме выделяются 9 позиций:

одна - для знака числа (знак + не печатается),

одна - для десятичной точки и

семь позиций для цифр числа.

Например: 23.78, -1.789, 45, -5000000

Числа с большей целой частью или маленькие дроби невозможно представить в этой форме.

Например, числа 123456780000 и 0,0000000012

Для расширения диапазона и сокращения записи чисел используется экспоненциальная форма вещественного числа. С такой формой записи числа вы сталкивались на уроках физики и химии для представления очень маленьких и очень больших чисел.

В этой форме можно любое число представить умножением на степень числа десять. Так число 1, 2 можно представить различными способами:

$$1,2=1,2*10^{-0}=12*10^{-1}=120*10^{-2}=0,12*10^{-1}=0,012*10^{-2}$$

Существует множество представлений одного и того же числа в экспоненциальной форме. Но есть стандартная форма записи числа. Число с нулевой целой частью и ненулевой первой цифрой дроби называют нормализованным.

Дробная часть нормализованного числа называется **мантиссой** числа, а показатель степени десяти - **порядком** числа.

В рассмотренном примере нормализованным является число 0,12. При этом порядок равен 1, а мантисса 0,12.

При записи числа в экспоненциальной форме на компьютере знак умножения на 10 опускается, число 10 заменяется на букву Е (от англ. Exponent - показатель степени), а далее в той же строке записывается порядок числа со знаком. Кроме того КуМир разрешает в записи числа использовать заглавную букву Е и строчную букву е и при выводе использует строчную.

Например,

число **123456780000** будет представлено так: **0.1234568E+12** или **0.1234568E+12**, а число **0,000000012** будет выглядеть так: **0.12E-7 или 0.12e-7.**

Если в отведенное количество позиций мантисса не помещается, число округляется.

Каждая величина должна иметь значение. С помощью значения величины можно сравнивать и отмечать изменения, которые с ними происходят, например, изменение веса, места жительства, количество отсутствующих на уроке, изменение цвета. Значение величин выражается числами или словами. Например, длину можно определить числом, а окрас собаки — словом.

Значение переменной меняется с помощью команды присваивания, в результате которой в ячейку памяти, отведенную для данной переменной, записывается значение выражения, указанного в правой части команды.

Обозначается присваивание знаком « := ». Имя величины, которой присваивается значение, должно стоять всегда слева от знака присваивания, а выражение, определяющее значение переменной, - справа. В частном случае выражением может быть просто имя числовой или символьной величины.

```
Например:
```

```
      A := 87
      - переменной с именем A присвоено значение 87;

      X := A
      - переменной с именем X присвоено значение переменной A;

      B := 43+A
      - переменной с именем B присвоено значение 43+A;

      C1 := - 3.5
      - переменной с именем C присвоено значение — 3,5;

      Y := 'мышь'
      - переменной с именем Y присвоено значение мышь.
```

При присваивании нового значения переменной, старое уничтожается в памяти и на его месте записывается новое.

```
A := 5;

A := 7;

A := -1;
```

В результате такой последовательности действий в ячейке памяти, определяющей значение величины A, будет -1.

Величине можно присваивать и арифметическое выражение, в которое входит имя этой величины. Компьютер сначала вычисляет выражение справа от знака «присвоить». Затем полученное значение записывает в ячейку памяти, соответствующую величине находящейся слева от знака «присвоить».

Поэтому в информатике правомерно, например, такое выражение

$$n := n + 1$$

В этом случае выполняются следующие действия:

- извлекается из ячейки памяти значение переменной **n**,
- выполняется сложение,
- новое значение переменной записывается вместо старого.

Описание типа величины можно производить в строке **нач** или в последующих строках. Для описания целых чисел используется слово **цел**. Например : **нач цел** A, SUMMA. Это экономит компьютерную память, т.к. под целую величину отводится места меньше, чем под вещественные числа.

Вещественные величины описываются с помощью слова вещ.

Рассмотрим пример программы

Программа алг тест нач цел A, B, C | A := 2; B := 7 | C := A + B | A := A+C кон

В КуМире алгоритм обязательно должен иметь имя

Рекомендуется при наборе программы в КуМире установить флаг «Выделять курсивом латинские символы в именах»; это позволяет отличать символы кириллицы и латиницы одинакового начертания.

В данном примере переменной А присваивается значение 2, а переменной В - значение 7. Значение переменной С складывается из значений А и В и равно 9. Затем изменяется значение А. Таким образом: из ячейки извлекается число 2, к нему добавляется значение С, равное 9, получается 11, затем результат записывается в ячейку, определяющую значение А.

В результате переменные будут иметь следующие значения: А=11, В=2, С=9.

🖎 Выполните упражнения

TT 6		T/			
чему оудет р	авна переменная	к после	выполнения	следующих	деиствии

K:=5

$$\hat{S} := K + 2$$

$$K:=S+2$$

$$K := K + 10 + S$$

1. Чему равна переменная S в результате выполнения следующих действий:

$$S:=10$$
 $K:=10/2+4$

$$S := S/2$$

$$K:=S+5-K$$

$$F := + C + D$$

3. Чему равна символьная переменная L после следующих действий:

4. Запишите следующие числа в экспоненциальной форме:

- a) 326, 09871
- б) -0,0007584
- в) 23467594, 76

5. Преобразуйте числа из экспоненциальной формы в обычную:

- a) 0.456748E-5
- б) 0.23456Е7
- B) -0.456E+4