

# ***Программирование на Паскале – первый уровень***

*Простые (линейные) программы.*

*Операторы ввода-вывода.*

*Переменные величины.*



Учитель информатики  
МОУ Юрцовская СОШ Егорьевский район  
Сафонова Ольга Викторовна

Программа на Паскале состоит из команд, как дом из кирпичей. Прежде чем строить дом, нужно узнать, как выглядят кирпичи. Команды, из которых состоит программа на Паскале и многих других языках, называются **операторами**. Многие операторы на Паскале являются **обращениями к процедурам**.

# Процедура вывода **Write**

Практически любая программа кроме всего прочего показывает на экране какие-нибудь числа или слова. Оператор, с которым мы познакомимся, приказывает компьютеру изобразить на экране монитора ту или иную информацию, состоящую из символов. Пусть мы хотим, чтобы компьютер изобразил на экране число **1999**. Эту задачу выполняет процедура **Write**. К ней обратимся так: **Write(1999)**. То, что нужно изобразить, положено заключать в круглые скобки. По-английски слово **Write** означает «пиши», читается — «**райт**».

А теперь поучимся на примерах.

# Примеры.

*Как пишем обращение к процедуре*

*Что мы видим в результате на экране*

**Write(– 500)**

**– 500**

**Write(3 + 2)**

**5**

**Write(3 \* 2)**

**6**

**Write(2 \* 2 – 1)**

**3**

**Write('Хорошо!')**

**Хорошо!**

**Write('Я изучаю Паскаль')**

**Я изучаю Паскаль**

**\*** – знак обозначает умножение.

Текст, в отличие от чисел и выражений, нужно брать в одинарные кавычки.

# Первая программа на Паскале

Рассмотрим, как устроена простая программа на Паскале:

**BEGIN**

**Write**('Начали');

**Write**(8 + 1);

**Write**(5 – 2)

**END.**

Программа состоит из трёх операторов **Write**, записанных друг за другом. Кроме них, программы на Паскале содержат «знаки препинания», в нашем случае это служебные слова **BEGIN** и **END**, точка с запятой и точка. Слова **BEGIN** и **END** не являются, в отличие от **Write**, приказами на выполнение каких-либо действий по исполнению задания.

# Правила расстановки «знаков препинания»

- **BEGIN** будем ставить в начале программы, чтобы компьютеру было видно, откуда она начинается. **BEGIN** переводится как **НАЧАЛО**, читается — «би'гин».
- **END** с точкой ставится в конце программы, чтобы компьютеру было видно, где она кончается, **END** переводится как **КОНЕЦ**, читается — «энд».
- Точкой с запятой необходимо отделять операторы друг от друга. Служебные слова **BEGIN** и **END** от операторов точкой с запятой отделять не нужно.

# Выполняем первую программу на компьютере

1. **Запустите Паскаль.** Наверху экрана возникает меню, а под ним синее (обычно) окно во весь экран с мигающим курсором. Можно вводить программу. Если окно не появилось, то нажмите клавишу **F4**, а затем в меню слова **File** и **New**. (**File → New**).
2. **Введите в окно программу** как обычный текст в обычном текстовом редакторе.
3. Если получится, **сохраните программу** на жёстком диске. Для этого **File → Save**, а затем в открывшемся диалоговом окне выберите каталог и введите имя файла, в который будете сохранять программу.

4. Выполните программу. Для этого выполните Run → Run, а чтобы увидеть результаты, нажмите (Ctrl+F9), а затем Alt-F5, что означает: удерживая нажатой клавишу Alt, щёлкните по клавише F5. Выполнив программу в первый раз, поэкспериментируйте: измените содержимое операторов Write и выполните программу ещё раз.
5. Если в вашей программе Паскаль заметил ошибку, он ставит на неё (или недалеко от неё) курсор и сообщает о ней золотыми буквами на красном фоне.



# *Распознаём сообщения компьютера об ошибках*

Составим следующую программу:

**BEGIN**

**Write('Начали!');**

**Write(78 + 15)**

**END.**

Выполните программу.

Она напечатала следующий результат:

**Начали!93**

Введём в нашу программу ошибки и будем наблюдать за реакцией Паскаля.

# Сообщения компьютера об ошибках

- Сотрите точку после **END**. Теперь запустите программу. На экране появится сообщение **Unexpected end of file**, которое переводится как **Неожиданный конец файла**. Паскаль нашёл эту ошибку в программе и сообщает о ней, поставив курсор в строку, содержащую ошибку. Уберите сообщение компьютера клавишей **Esc**.
- Исправьте эту ошибку и введите другую – сотрите точку с запятой после **Write('Начали!')**. На этот раз сообщение такое: **“;” expected**, что значит **Ждал точку с запятой**. Однако курсор стоит совсем не в том месте, где ошибка, а в начале следующего оператора. Паскаль не всегда точно определяет место ошибки.

# Сообщения компьютера об ошибках

- Исправьте эту ошибку и введите другую – напишите само имя оператора с ошибкой – `Wite('Начали!')`. Реакция Паскаля – `Unknown identifier`, что означает **Неизвестное имя**. Имеется в виду имя процедуры `Write`.
- Исправьте эту ошибку и введите другую – сотрите правую кавычку в операторе `Write('Начали!')`, чтобы получилось `Write('Начали!)`. Реакция Паскаля – `String constant exceeds line`. Это намёк на то, что **раз кавычку открыли, то надо её закрывать**.
- Теперь сотрите левую кавычку. Реакция Паскаля – `Syntax error`, что значит **Синтаксическая ошибка**. Паскаль в затруднении: он знает, где ошибка, но в чём она состоит – не знает.

# Сообщения компьютера об ошибках

□ Исправьте ошибки и введите ещё одну – сотрите правую скобку в операторе `Write('Начали!')`, чтобы получилось `Write('Начали!'`.

Реакция Паскаля – `“)” expected`, что значит **Ждал скобку**.

□ Исправьте ошибку и введите другую – сотрите левую скобку в операторе `Write('Начали!')`, чтобы получилось `Write'Начали!')`.

Реакция Паскаля – `“;” expected`, что значит **Ждал точку с запятой**, причём курсор стоит на букве **Н**. Вот здесь

Паскаль не прав (просто нельзя учесть все возможные причины ошибки). Паскаль иногда неправильно определяет характер ошибки.

# *Используемая литература*

## **1. Лукин Н.С.**

Турбо-Паскаль 7.0 . Самоучитель для начинающих – 2-е изд., испр. И допол. – М.: «Диалог-МИФИ», 2005. – 400 с.