

Отчет по опытно-экспериментальной работе в 2006-2007 учебном году

А.Б.Кондратович, начальник отдела компьютерных технологий
УО «Витебский областной государственный ИПК и ПРР и СО»
2007-08-23

Цели и задачи:

В ходе эксперимента было необходимо проанализировать следующие направления:

- Адаптация и продвижение Блэкбокса и др. Оберонов в систему образования — цель научно-образовательного проекта Информатика-21, стартовавшего в 2001 г., и «вышедшего в Сеть» осенью 2002 г.
- обеспечение подготовки учащихся с ориентацией на профили, углубленно изучающих методы и средства информационных технологий решения задач и программирование;
- реализация в наиболее полной мере возрастающего интереса учащихся к углубленному изучению программирования через совершенствование их алгоритмического и логического мышления;
- раскрытие основных приёмов и методов обработки информации разной структуры и организации с помощью разнообразных систем обработки информации и языка программирования;
- развитие у учащихся информационной культуры.
- Простота Компонентного Паскаля (в отличие от других «мощных» языков) позволяет быстро осваивать его учителям, не имеющих опыта серьезного программирования. Хорошая интеграция Блэкбокса с операционной системой позволяет использовать уже имеющиеся библиотеки и программы.

Реализация:

В учебных заведениях Витебской области эксперимент проводится с учащимися 5-9 классов 12 летней школы. В областной олимпиаде по информатике впервые были учащиеся, которые решали задачи на КП. В областной олимпиаде по программированию среди учащихся 5-6 классов «Информаша» 14 учащихся решали задачи (приложение 1) на КП.

Положительное и отрицательное

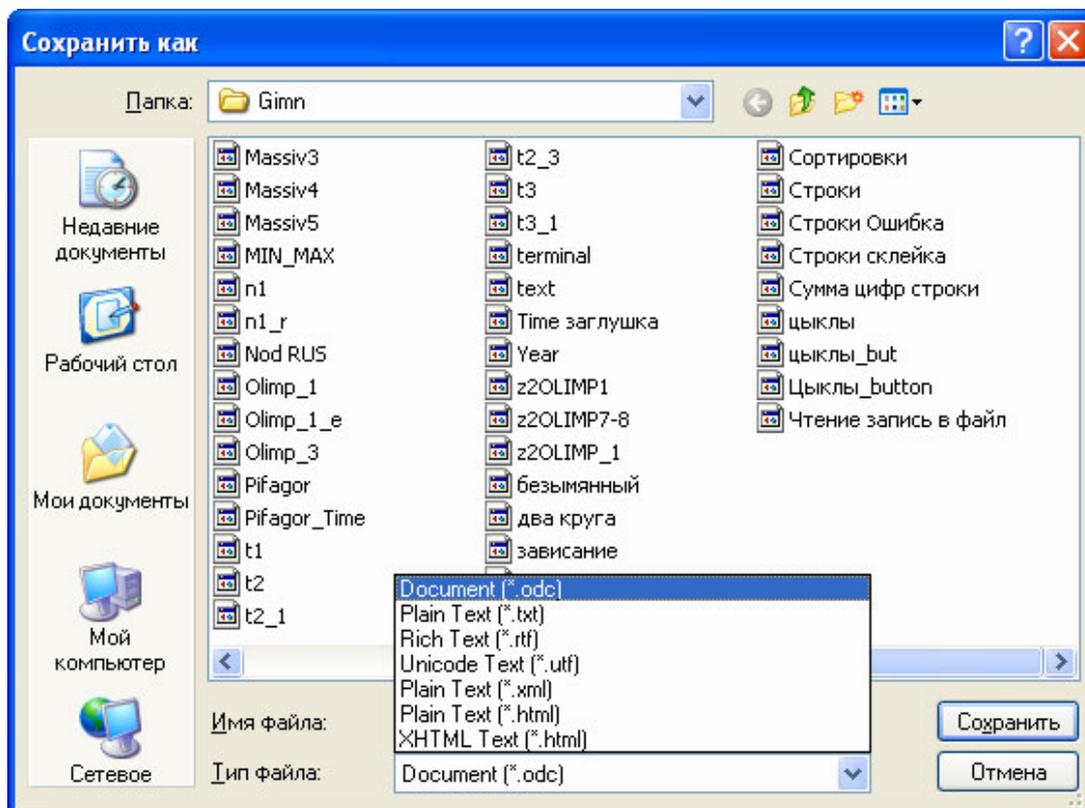
Учащиеся восприняли язык программирования Компонентный Паскаль в среде BlackBox достаточно оптимистически и только те, кто уже изучал язык программирования Паскаль — с некоторой настороженностью. Но эта настороженность достаточно быстро прошла.

Преимущества КП стали очевидны.

Первое, что выдвинуло КП вперёд — это среда BlackBox в которой достаточно легко работать, а редактирование документа организовано достаточно удобно.

Многооконная система ничем не отличается от стандартных окон операционной системы Windows. Работа с раскладкой клавиатуры также ничем не отличается от стандартной.

Сохранение файла не ограничивается восемью символами, как в Турбо Паскале (далее – ТП), и типы сохранения файла достаточно разнообразны:



Набор программы можно частично автоматизировать. Например, в Компонентном Паскале постоянно возникает необходимость печатать ключевые слова, состоящие только из заглавных букв. Кроме того, очень удобно иметь возможность сразу получать скелет сложных синтаксических конструкций типа IF .. THEN .. ELSE .. END и т.п. Обе эти возможности добавлены в виде команд, которые вызываются либо при нажатии клавиш F5 или Shift+F5, либо, соответственно, через меню: Текст → Заглавные и раскрыть или Текст → Заглавные.

На первом этапе изучения КП возникли трудности (особенно у тех, кто изучал язык программирования Паскаль): в системе программирования нет отладчика. Этот пункт требует пояснения.

Пошаговый отладчик — обязательный элемент традиционного "наивного" подхода к программированию. Этот подход заключается в том, чтобы написать первый вариант программы более или менее наугад, а потом "отлаживать" её, ставя программные заплатки на основе наблюдений за поведением программы с помощью пошагового отладчика.

С профессиональной точки зрения неадекватность подхода "начнем как попало, потом отладим" осуществляется в значительной степени наугад (хорошо известно выражение "агония отладки" — "the agony of debugging").

Мнение Н.Вирта: "... Нет другого, более сильного стимула для аккуратного программирования, чем отсутствие отладчика.

Альтернативой наивному подходу являются систематические методы, в которых программа выводится из требуемых пред- и постусловий. Такое проектирование опирается на систематическую верификацию (доказательство, проверку) логических свойств программы, которые должны удовлетворяться в ключевых точках программы (пред- и постусловия для процедур, инварианты циклов и т.п.). Тогда и "отладка" в идеале сводится к устранению синтаксических ошибок и мелких описок.

Для такого систематического программирования более чем достаточны средства, предлагаемые в Блэкбоксе (при аварийной остановке — например, если нарушено логическое условие в ASSERT — Блэкбокс дает возможность исследовать состояние локальных переменных всех процедур в цепочке вызовов, приведших к аварийной остановке, а также глобальных переменных всех модулей).

Пример:

In.String(s); ASSERT(In.Done);

In.LongInt(a); ASSERT(In.Done);

Контролирует чтение переменной длинного целого числа.

In.Int(b); ASSERT(b>0);

Контролирует чтение переменной целого (натурального) числа.

Отсутствие пошагового отладчика дисциплинирует процесс программирования подобно тому, как невозможность применить GOTO стимулирует использование методов, приводящих к программам, имеющим хорошую структуру.

Организация ввода-вывода в КП не включена в язык. Этот факт поначалу вызывал некоторое раздражение у учеников, уже владеющих навыками программирования в ТП, но и здесь в итоге проявились преимущества системы Блэкбокс. При многократном выполнении программы в ТП приходилось многократно вводить одни и те же данные. В КП это не требуется: достаточно один раз написать командер для выполнения программы (модуля) вместе с входными данными. В дальнейшем командер остается неизменным, а данные можно менять по необходимости.

Пример. Для задачи даны координаты центров окружностей и их радиусы необходимо выяснить их взаиморасположения. Командеры будут выглядеть следующим образом:

- ❗Размещение. Окружность 0 0 1 0 0 1
(*Совпадают*)
- ❗Размещение. Окружность 0 0 2 0 0 1
(*Не пересекаются*)
- ❗Размещение. Окружность 0 1 3 0 0 1
(*Не пересекаются*)
- ❗Размещение. Окружность 0 0 2 0 1 1
(*Касаются*)
- ❗Размещение. Окружность 0 0 2 0 1 2
(*Пересекаются*)
- ❗Размещение. Окружность 0 0 2 0 2 1
(*Пересекаются*)
- ❗Размещение. Окружность 0 0 2 0 4 2
(*Касаются*)
- ❗Размещение. Окружность 0 0 2 0 6 2
(*Не пересекаются*)

Текст модуля удобно помещать в начале документа. В Блэкбоксе обычно компилятор игнорирует все, что находится после текста модуля (т.е. после точки, закрывающей модуль), так что пространство после текста модуля можно использовать по своему усмотрению (помещать там команды вызова процедур модуля, а также хранить фрагменты программных текстов).

Отметим **некоторые особенности**:

Управляющие операторы записываются в лаконичной форме ОПЕРАТОР ...; ...; ...END в отличие от двух неудачных форм, которые использовались в старом Паскале: сокращенной ОПЕРАТОР ...; и полной ОПЕРАТОР begin ...; ...; ... end.

Язык стал чувствительным к регистру — имена *x* и *X* теперь являются различными. Все ключевые слова записываются заглавными буквами, что улучшает читаемость программ и позволяет использовать выделение жирностью и цветом для других целей — прежде всего для расстановки смысловых акцентов.

Окончания процедур записываются в виде END ИмяПроцедуры, что облегчает поиск потерянного END.

Управляющие операторы КП

Условный оператор.

```
IF ... THEN ... END;  
IF ... THEN ... ELSE ... END;  
IF ... THEN ... ELSIF ... THEN ... END;
```

Оператор выбора

```
CASE ... OF  
|... : ...  
END;  
CASE ... OF  
|... : ...  
ELSE ...  
END;
```

Оператор цикла «Пока»

```
WHILE ... DO  
...  
END;
```

Оператор цикла «Повторять»

```
REPEAT  
...  
UNTIL ... ;
```

Оператор цикла «Для»

```
FOR g :=... TO ... DO  
...  
END;  
FOR i := ... TO ... BY -... DO  
...  
END;
```

Безусловный цикл

```
LOOP  
  
END;
```

Массивы нумеруются только от 0, объявляются в виде ARRAY размер OF... Поддерживается передача в процедуру по ссылке открытых массивов (неопределенного на этапе компиляции размера), что особенно полезно - многомерных: PROCEDURE ... (VAR a: ARRAY OF ARRAY OF ...).

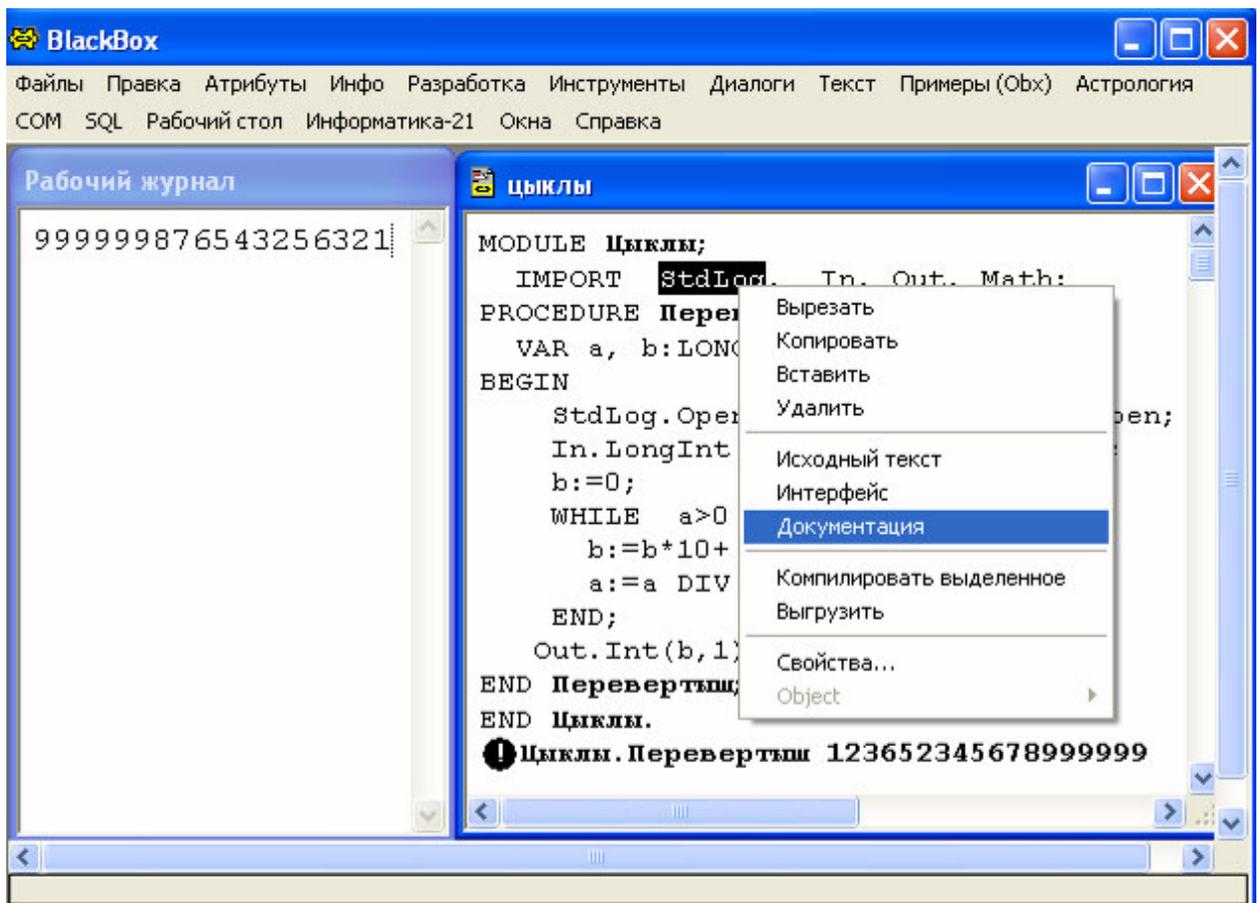
Важным требованием КП является то, что имя любой сущности из импортированного модуля всегда указывается в полном виде: Модуль.Имя (StdLog.Open; StdLog.Clear). Это позволяет при чтении кода программы сразу

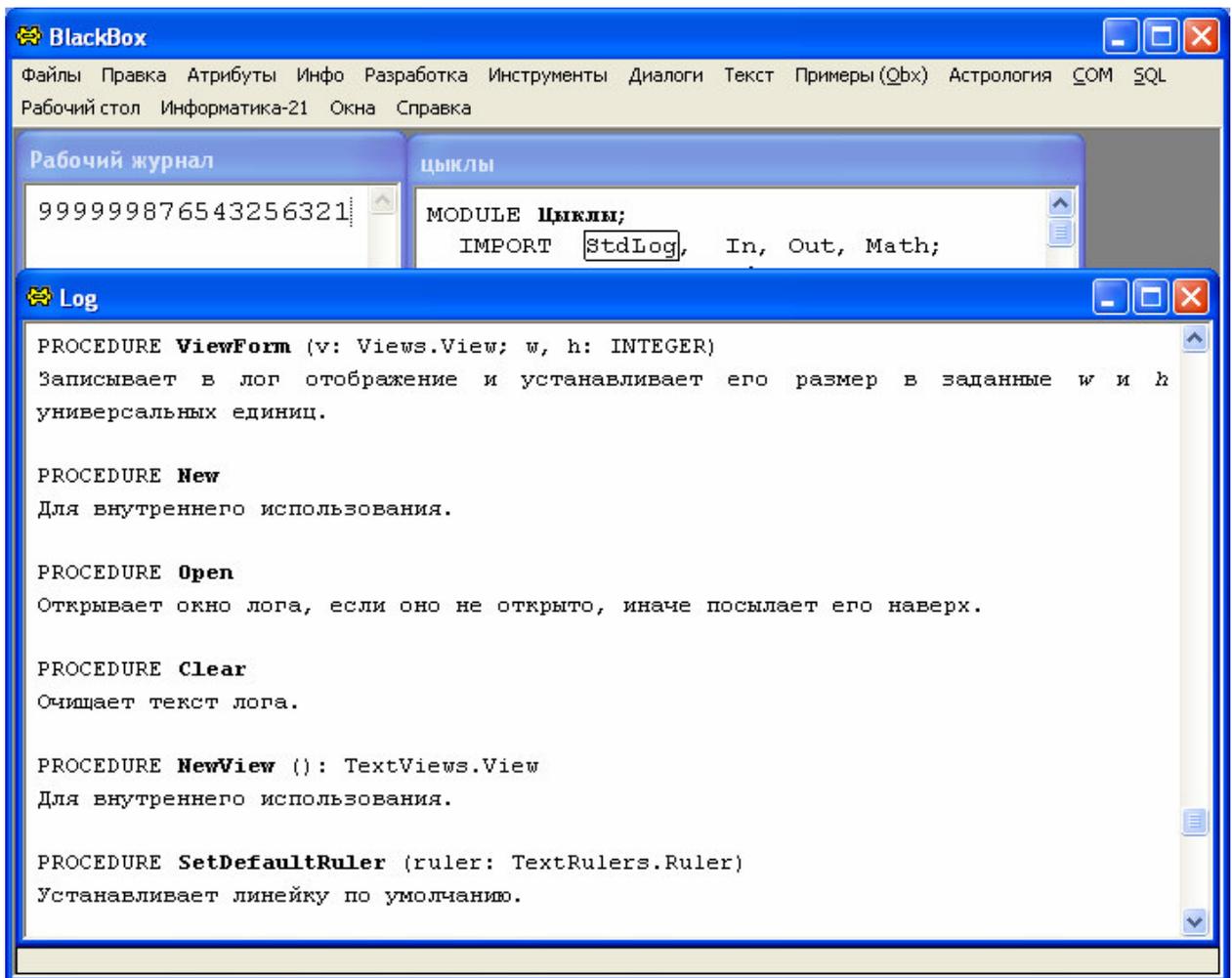
видеть, к какому из модулей относится тот или иной тип/переменная/процедура. Если имя модуля слишком длинное, можно назначить псевдоним: `IMPORT L:= StdLog` и далее работать с ним: `L.Open; L.Clear`.

Неоспоримым преимуществом КП является фиксированный диапазон значений основных типов

Тип	Диапазон значений
BOOLEAN	FALSE, TRUE
SHORTCHAR	0X .. 0FFX
CHAR	0X .. 0FFFFX
BYTE	-128 .. 127
SHORTINT	-32768 .. 32767
INTEGER	-2147483648 .. 2147483647
LONGINT	-9223372036854775808 .. 9223372036854775807
SHORTREAL	-3.4E38 .. 3.4E38, INF (32-битный формат IEEE)
REAL	-1.8E308 .. 1.8E308, INF (64-битный формат IEEE)
SET	подмножество из 0 .. 31

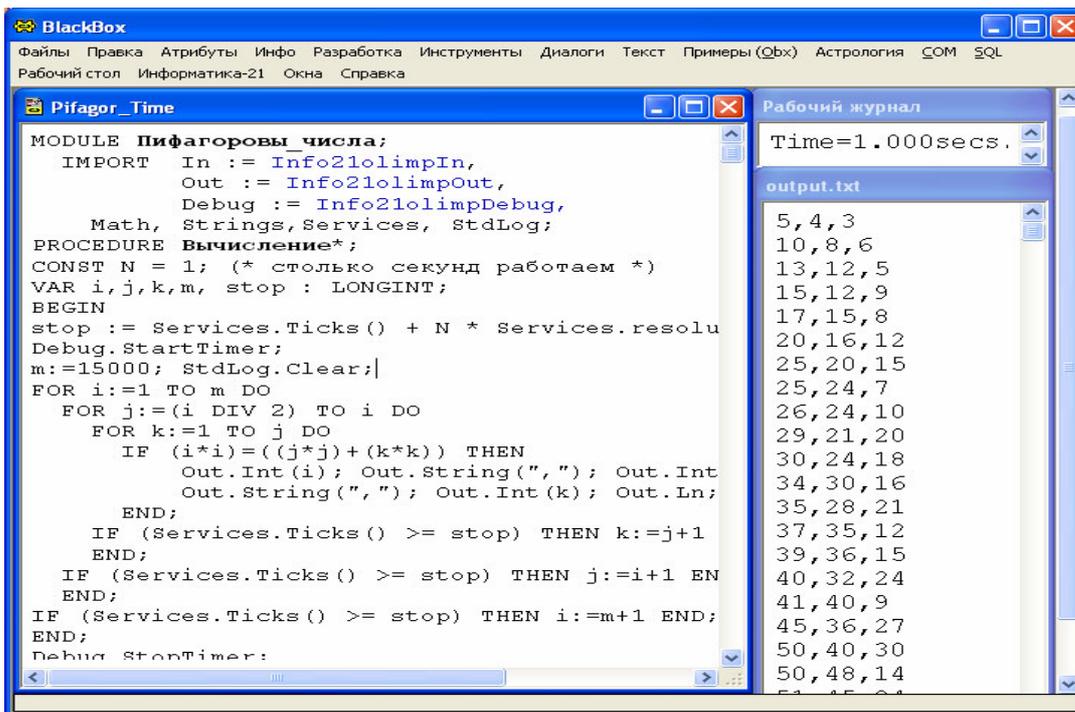
Справочная система построена в виде гиперссылок и доступности. Достаточно выделить модуль, процедуру или функцию, нажать правую клавишу мышки и выбрать «документация».





При необходимости можно самостоятельно написать, дописать, изменить любой файл справочной системы.

Работа с файлами предусматривает использование модуля Info21olimp или Info21sysFiles, предоставленного в рамках проекта «Информатика-21». Что заметно упрощает изучение темы.



При программировании в КП имеется возможность использовать ранее созданные модули и процедуры. Для этого достаточно описать их в строке **IMPORT**.

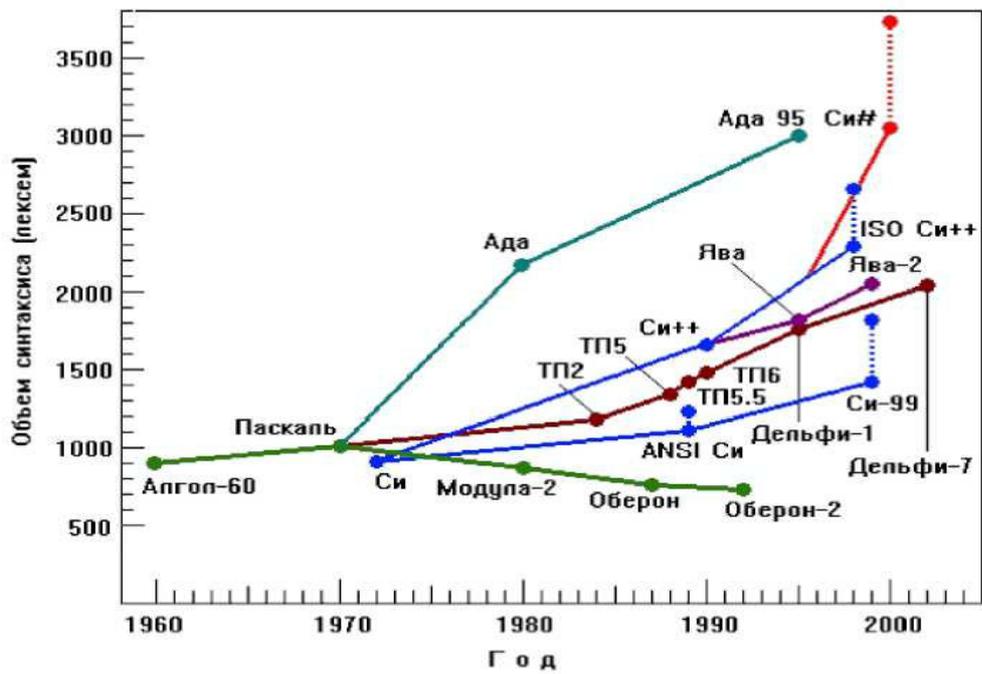
Выводы

Компонентный Паскаль — усовершенствованная версия языка Оберон-2, разработанная компанией Oberon microsystems.

Освоение языка программирования КП произошло достаточно быстро и безболезненно. Учащиеся легко освоили принцип построения модуля, его компиляцию, исправление ошибок и выполнение.

Уже на начальном этапе изучения языка программирования были оценены его преимущества перед другими языками программирования.

Схема, отображающая сложность (число лексем) в основных ЯП (из доклада Свердлова С. З. на II Всероссийском совещании по ИТ-образованию).



Из схемы видно, что меньшей сложности не имеет ни один язык программирования

Компонентный паскаль имеет все шансы завоевать образовательное пространство, и стать языком программирования школьного курса «Информатика»

Приложение 1.

УПРАВЛЕНИЕ ОБРАЗОВАНИЯ ВИТЕБСКОГО ОБЛИСПОЛКОМА
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«ВИТЕБСКИЙ ОБЛАСТНОЙ ГОСУДАРСТВЕННЫЙ ИПК И ПРР И СО»
ОТДЕЛ ОБРАЗОВАНИЯ АДМИНИСТРАЦИИ ПЕРВОМАЙСКОГО РАЙОНА г.
ВИТЕБСКА
СРЕДНЯЯ ШКОЛА №45 г. ВИТЕБСКА

**Областная олимпиада
учащихся 5 - 6 классов
«Информаша»**

22 мая 2007, г. Витебск

Задача № 1 Подорожание и подешевление

Некий товар стоимостью M рублей подорожал на $A\%$, потом на $A\%$ подешевел. Определить, когда цена его была ниже: до подорожания или после подешевления?

Задача № 2 Буратино

По окончании каждого месяца Буратино приносит папе Карло свой заработок в X золотых монет. Папа Карло за следующий месяц тратит ровно половину имеющихся на его начало денег. В начале срока у них было Y золотых монет. Составить программу, выводящую номера месяцев и суммы денег, имевшихся у Буратино и Карло на начало и конец месяца.

Задача № 3 Царевна-лягушка

Царевна-лягушка съедает ежедневно на 20% комаров больше, чем в предыдущий день и ещё M комаров. Написать программу, определяющую, через сколько дней количество съеденных комаров превысит N , если в первый день было съедено A комаров.

Задача № 4 Числа

Найти все трёхзначные числа, сумма цифр которых делится на M .

Задача № 5 Хочу мороженого

У первоклассника Пети m рублей. Мороженое стоит k рублей. Петя решил наесться досыта мороженого, для этого он покупал по одному мороженому и съедал ее до тех пор, пока ему хватало денег. Как Пете узнать, сколько денег останется у него в конце концов? Учтите, что Петя делить еще не умеет, а умеет только вычитать и складывать. Сколько мороженого он может съесть?

Задача № 6 И опять числа

Даны два двузначных числа A и B . Из этих чисел составили два четырехзначных числа: первое число получили путем написания сначала числа A , а затем B ; для получения второго сначала записали B , а потом A . Найти числа A и B , если известно, что первое четырехзначное число нацело делится на 99 , а второе – на 49 .