

PascalABC.NET 2023 – русский Паскаль

Михалкович С.С.

*Южный федеральный университет,
Институт математики, механики и компьютерных наук*
miks@sfedu.ru

Доклад на четвертой Всероссийской научно-методической конференции
«Использование системы программирования PascalABC.NET
в обучении программированию»
(Ростов-на-Дону, 29-30 марта 2023 г.)

PascalABC.NET – российский Паскаль

PascalABC.NET – **единственный** язык программирования **общего назначения** и система программирования, целиком **разработанные в России**

PascalABC.NET имеет свободную лицензию LGPL

PascalABC.NET широко используется в школах **России**

PascalABC.NET имеет мощную методическую поддержку: в компьютерной школе мехмата ЮФУ развиваются новые методики обучения современному программированию, оказывающие влияние на сам язык и его библиотеки

PascalABC.NET имеет постоянно действующий форум поддержки и Телеграм-канал на **русском**

PascalABC.NET имеет простую **русскоязычную** среду с сообщениями об ошибках и справкой на **русском** языке, а также позволяет писать программы с именами на **кириллице**

PascalABC.NET в 2022 году добавлен в **русские** Линуксы:

- Alt Linux Образование,
- Astra Linux Orel Special Edition,
- М ОС (Москва)

PascalABC.NET активно используется в ЕГЭ и **русских** олимпиадах по информатике

Миссия и области использования PascalABC.NET

Миссия PascalABC.NET

PascalABC.NET (2007 г.) – язык программирования Паскаль нового поколения, пришедший на замену «стандартному Паскалю»:

- язык для **обучения современному программированию**
- язык для **сферы образования и научных исследований**
- язык, **аккумулирующий лучшие современные языковые конструкции и библиотечные методы** из таких языков как C#, Python, Kotlin, Java, Haskell
- **лаконичный и понятный**, конкурируя по этим показателям с языком Python
- компилятор PascalABC.NET проверяет большинство ошибок до запуска программы, что повышает скорость и качество разработки
- **эффективный**: скорость программ примерно в 100 раз быстрее чем на Python и примерно на 20-30% медленнее, чем на C++
- обеспечивает легкий переход на промышленные языки C#, Java, Python, C++

Стандартный Паскаль (его коды можно увидеть на ЕГЭ) давно не развивается, сильно устарел и вызывает отторжение учеников

Области использования PascalABC.NET

- Для обучения программированию **школьников** на **базовом** уровне
- Для обучения программированию **школьников** на **углубленном** уровне
- В **дополнительном образовании** (DL Club – 2000 школьников на Stepik за год, компьютерная школа мехмата ЮФУ – 400 школьников очно каждый год, ...)
- В **олимпиадах** по программированию (**третий** по популярности язык). PascalABC.NET имеется практически во всех системах олимпиадной проверки
- При **подготовке к ЕГЭ** по информатике
- Для обучения **студентов ИТ направлений** (программа «Фундаментальная информатика и информационные технологии» на мехмате ЮФУ, набор 110 человек)
- В научных исследованиях (более 350 работ в РИНЦ: сайт elibrary.ru)
- Как средство программирования «**на каждый день**»

События 2022-23 гг., связанные с PascalABC.NET

PascalABC.NET – история развития

Основная причина создания – обучение современному программированию

2007 год

Выпуск первой версии PascalABC.NET.
Развитие методики преподавания
PascalABC.NET: полигон – **детская**
компьютерная школа мехмата ЮФУ
(выпуск – 700 учащихся в год)



2015-2021 гг.

Активное развитие языка PascalABC.NET.
Использование **лучших идей**
современных языков
программирования. Приближение по
краткости записи к языку Python.



2023 год

Модуль невидимой
автоматической проверки задач.
Четвёртая Всероссийская
конференция по использованию
PascalABC.NET в обучении



2015 год

Переход на свободную лицензию
LGPL. Начало активного
использования PascalABC.NET в
школах России и на олимпиадах
по программированию

2021-2022 гг.

Компьютерный ЕГЭ - усложнение задач.
Вторая и третья Всероссийские конференции
по использованию PascalABC.NET в обучении
программированию.
Создание Jupyter Notebook PascalABC.NET

Воскресная компьютерная школа мехмата ЮФУ

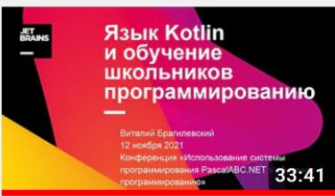
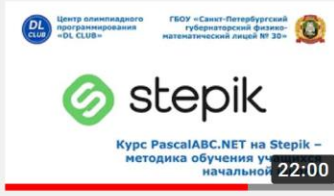
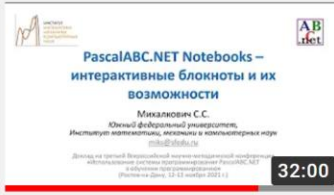
Воскресная компьютерная школа мехмата ЮФУ является основной площадкой для совершенствования системы программирования PascalABC.NET (методика, сервисы)

Каждый год ВКСШ выпускает около **700 школьников** с 6 по 11 класс. В группах программистов **PascalABC.NET** – основной язык программирования, используются также **Python** и **C#**.



III научно-практическая конференция PascalABC.NET

Более 150 участников, 16 докладов, 2 дня конференции, трансляция в интернет

 PascalABC.NET 2021 Михалков С.С. Южный федеральный университет, Институт математики, механики и компьютерных наук msh@yandex.ru Доклад на третьей Всероссийской научно-практической конференции «Использование систем программирования PascalABC.NET в обучении программированию» (Росток-на-Дону, 12-13 ноября 2021 г.) 49:43	 Презентация курса на Stepik «PascalABC.NET: современный код» Осипов А.В. Конференция «Использование систем программирования PascalABC.NET в обучении программированию» (Росток-на-Дону, 12-13 ноября 2021 г.) 38:50	 СТРАТЕГИИ ПОДГОТОВКИ К ЕГЭ ПО ИНФОРМАТИКЕ 2022 И РЕШЕНИЯ НЕКОТОРЫХ ЗАДАЧ Алексей Богданов, г. Архангельск VK: https://vk.com/danov Группа М. ЕГЭ Информатика – Инновации https://vk.com/stepik_danov YouTube канал – Alex Danov https://www.youtube.com/c/AlexDanov 37:37	 Язык Kotlin и обучение школьников программированию Виталий Бегинцевский 12 ноября 2021 Конференция «Использование систем программирования PascalABC.NET в обучении программированию» 33:41	 Обучение младших школьников 2D/3D графике и анимации на языке программирования PascalABC.NET Центр олимпиадного программирования «DL CLUB» ГБОУ «Санкт-Петербургский губернаторский физико-математический лицей № 30» 23:03
PascalABC.NET 2021 163 просмотра • 4 месяца назад	Презентация курса на Stepik «PascalABC.NET:... 78 просмотров • 4 месяца назад	Стратегии подготовки к ЕГЭ по информатике 2022... 62 просмотра • 4 месяца назад	Язык Kotlin и обучение школьников... 41 просмотр • 4 месяца назад	Обучение младших школьников 2D/3D график... 43 просмотра • 4 месяца назад
 Конференция «Использование систем программирования PascalABC.NET в обучении программированию» 2021 Динамические массивы в языке PascalABC.NET Как и чему учить? Прушин Максим Валентинович Старший преподаватель каф. прикладной математики и программирования ЮФУ mvalentin@yandex.ru 45:53	 DL CLUB ГБОУ «Санкт-Петербургский губернаторский физико-математический лицей № 30» stepik Курс PascalABC.NET на Stepik – методика обучения учащихся начальной школы Михалков С.С. Южный федеральный университет, Институт математики, механики и компьютерных наук msh@yandex.ru Доклад на третьей Всероссийской научно-практической конференции «Использование систем программирования PascalABC.NET в обучении программированию» (Росток-на-Дону, 12-13 ноября 2021 г.) 22:00	 Сравнение Python и PascalABC.NET Михалков С.С. Южный федеральный университет, Институт математики, механики и компьютерных наук msh@yandex.ru Ваш вариант или самый быстрый, или самый надежный, а что из этого вышло? – А.В. Вайс 31:58	 Модуль School базовых алгоритмов информатики Осипов А.В. Конференция «Использование систем программирования PascalABC.NET в обучении программированию» (Росток-на-Дону, 12-13 ноября 2021 г.) 49:09	 Вся правда о лямбда-выражениях Данилов И.М. 31:48
Динамические массивы в языке PascalABC.NET 215 просмотров • 4 месяца назад	Курс PascalABC.NET на Stepik – методика обучен... 34 просмотра • 4 месяца назад	Сравнение Python и PascalABC.NET 219 просмотров • 4 месяца назад	Модуль School базовых алгоритмов информатики 169 просмотров • 4 месяца назад	Вся правда о лямбда-выражениях 73 просмотра • 4 месяца назад
 Конференция «Использование систем программирования PascalABC.NET в обучении программированию» 2021 Написание Telegram-бота в PascalABC.NET Прушин Максим Валентинович Старший преподаватель каф. прикладной математики и программирования ЮФУ mvalentin@yandex.ru 34:08	 PascalABC.NET Notebooks – интерактивные блокноты и их возможности Михалков С.С. Южный федеральный университет, Институт математики, механики и компьютерных наук msh@yandex.ru Доклад на третьей Всероссийской научно-практической конференции «Использование систем программирования PascalABC.NET в обучении программировании» (Росток-на-Дону, 12-13 ноября 2021 г.) 32:00	 3. Знаете ли вы с другими языки программирования? Если да, то есть ли с ними точки соприкосновения и можно ли их использовать для решения задач ЕГЭ? А для дальнейшего обучения? 25:14	 Плагин PascalABC.NET для интегрированной среды Visual Studio Code Косикова Оксана Южный федеральный университет, Институт математики, механики и компьютерных наук или ИТ-ИТ okosikova@yandex.ru 25:49	 Использование электронного задачника при решении задач ЕГЭ на обработку файлов М. Э. Абрамян Южный федеральный университет, Институт математики, механики и компьютерных наук или ИТ-ИТ mabramyan@yandex.ru Восковая научно-методическая конференция «Использование систем программирования PascalABC.NET в обучении программированию» (Росток-на-Дону, 10-12 ноября 2021 г.) 28:11
Написание Telegram-бота в PascalABC.NET 198 просмотров • 4 месяца назад	PascalABC.NET Notebooks – интерактивные блокноты ... 39 просмотров • 4 месяца назад	Дискуссия «PascalABC.NET глазами студентов,... 12 просмотров • 4 месяца назад	Плагин PascalABC.NET для интегрированной среды... 201 просмотр • 4 месяца назад	Использование электронного задачника... 41 просмотр • 4 месяца назад

PascalABC.NET – статистика сайта

Система программирования PascalABC.NET

В XXI веке растёт спрос на языки программирования, позволяющие писать высокоуровневый код в лёгкой, компактной и понятной форме. Современные реализации языка Паскаль, такие как PascalABC.NET, здесь могут многое предложить.

Новости

07.02.23. Опубликовано сравнение конструкций и стандартной библиотеки языков PascalABC.NET и C#.

13.07.22. Вышла бета-версия интегрированной среды PascalABC.NET

Рейтинг > Компьютеры > Программирование

НАЗВАНИЕ САЙТА	Посетители
1. Хекслет, практические курсы программирования	29,821
2. Ru.Board - Компьютерный портал	7,246
3. Освой программирование играючи!	961
4. Мир MS Excel Всё об Excel	924
5. Система программирования PascalABC.NET	661

- Посещаемость – **40000** посетителей в месяц
- Пользователи из Беларуси, Молдовы, Казахстана, США, Узбекистана
- В рейтинге mail.ru входит в **Топ-5** в категории Программирование

ГЕОГРАФИЯ	Январь 2023		ПОСЕТИТЕЛИ
▶ ☒ Россия	29,821	72.27%	
▶ ☒ Беларусь	7,246	17.56%	
▶ ☒ Республика Молдова	961	2.33%	
▶ ☒ Украина	924	2.24%	
▶ ☒ Казахстан	661	1.60%	
▶ ☐ Соединенные Штаты Америки	184	0.45%	
☐ Узбекистан	149	0.36%	
☐ Киргизия	144	0.35%	
▶ ☐ Германия	116	0.28%	
☐ Нидерланды	110	0.27%	

Количество скачиваний PascalABC.NET по данным официальных сайтов

2023 год: **6 миллионов**
скачиваний с начала проекта,
1500 скачиваний в день



Сегодня Вчера **Неделя** Месяц Квартал Год 6 — 12 фев 2023

Сегмент: 1 условие × Точность: 100% ▾

Визиты, в которых Роботность: Только люди × + для людей, у которых +

1. PascalABCNETSetup ⬇

Конверсия	50,8 %
Достижения цели	9 781
Целевые визиты	7 864

2. PascalABCNETMiniSetup ⬇

Конверсия	7,68 %
Достижения цели	1 421
Целевые визиты	1 189

3. PascalABCNETLinux ⬇

Конверсия	1,54 %
Достижения цели	297
Целевые визиты	238

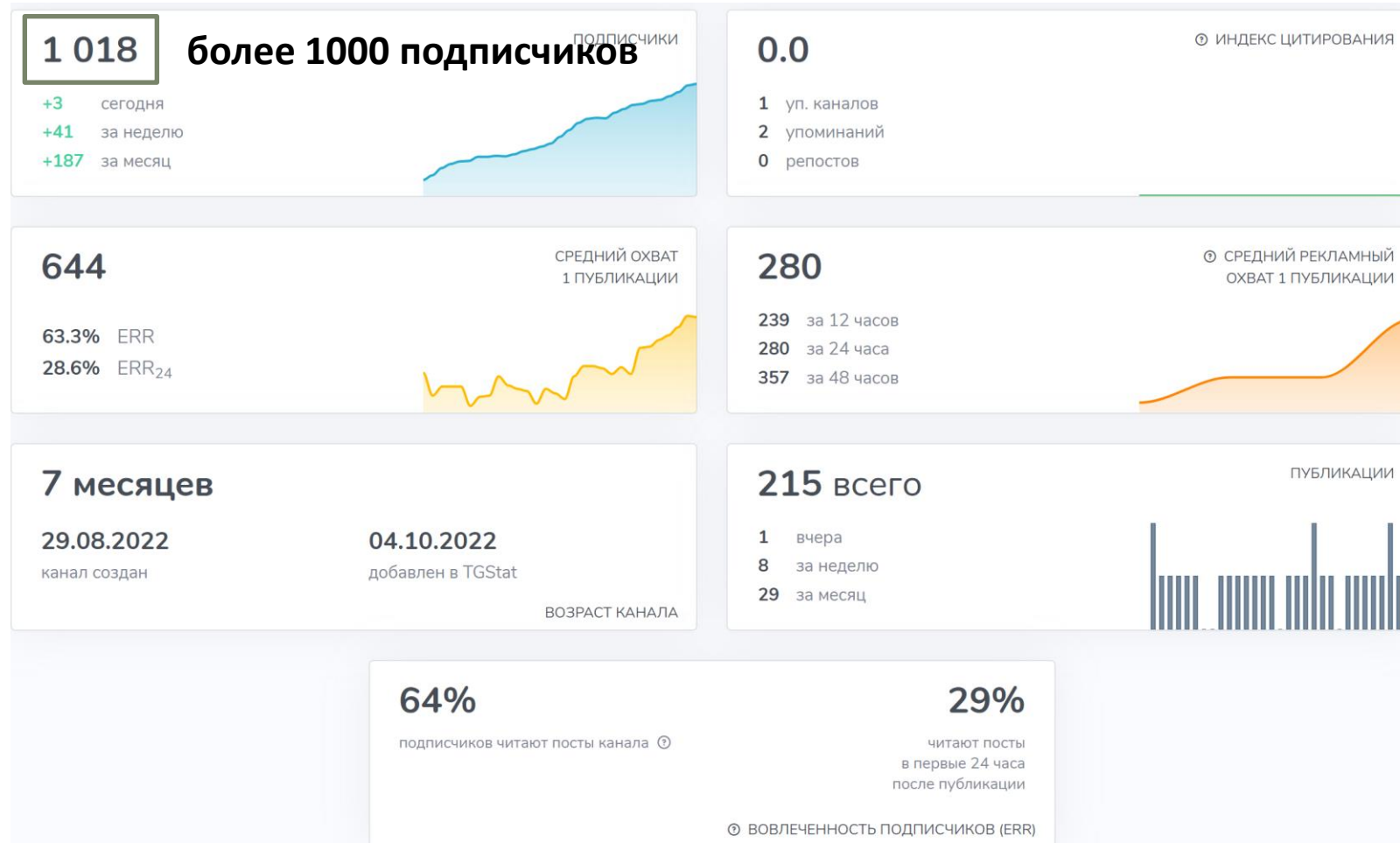
Данные
Яндекс-метрики:
11400 скачиваний
в неделю

Продвижение PascalABC.NET 2022-23гг.

- В 2022 году мы зафиксировали версию: **PascalABC.NET 3.8.3**
- Версия 3.8.3 в настоящее время рекомендуется к установке в большинстве пунктов ЕГЭ
- Версия 3.8.3 установлена в большинстве олимпиадных проверяющих систем, в т.ч. в новейшей системе **Сириус-курсы**
- Версия 3.8.3 интегрирована в репозитории российских Линуксов:
 - **Alt Linux** Образование 10.1
 - **Astra Linux** Orel Special Edition
 - **М ОС** (школы Москвы)
- **Министерство просвещения** рассматривает вопрос о включении во ФГОС PascalABC.NET как отдельного языка на профильный уровень

Открытие официального телеграм-канала PascalABC.NET

В сентябре 2022 г. был открыт официальный Телеграм-канал PascalABC.NET
https://t.me/pascalabc_official. На скриншоте – статистика TGStat:



Telegram-группа для обсуждения PascalABC.NET

В месте с каналом открыта группа для обсуждения, содержащая ряд тематических подгрупп:

<https://t.me/PABCOfficial>

PascalABC.NET
229 участников

Конференция "Использование систем...
Andrey: Мой вопрос был не про Zoom. Ег...

Общие вопросы, обсуждение статей
Черновик: Написание Telegram-бота в Па...

P PascalABC NET под Linux
Nikolay: <https://www.youtube.com/watch?v=nw...>

B Болталка
Сергей: то есть в случае HashSet типо проход...

R PascalABC.NET для начинающих
Александр: Нашёл ещё примеры использова...

D Для учителей и преподавателей
Вы: Доклад о сравнении PascalABC.NET и ...

V Вопросы разработчикам
Сергей: function _Each<TKey,TValue>(se...

P Помогите решить задачу!
Александр Викторович: Там же есть какая-то ...

M Мнимые вирусы в PascalABC.NET
Александр Викторович: Ну и что это? Ни к...

Конференция "Использование системы программирования PascalABC
68 сообщений

Закреплённое сообщение
29 – 30 марта 2023 г. Институт математики механики и компьютерных н

Запланированные трансляции

Использование системы программирования PascalABC.NET в обучении программированию
День первый (29.03.2023), часть 1 (10:00 – 13:00)

10:00 – 10:05 Вступительное слово организатора конференции
10:05 – 10:20 Представление спонсоров мероприятия, спонсорство, спонсоры и спонсируемые лица
10:20 – 10:30 Михаил С.С. Пучкин: доклад «Математика и информатика»
10:30 – 10:40 Михаил С.С. Пучкин: доклад «Паскаль»
10:40 – 10:50 Сергей Д. Ионов: доклад «PascalABC.NET на YouTube»
10:50 – 11:00 Александр А.А. (Ульяновский, СПбГУ) Видео ЕГЭ 2023 по информатике на PascalABC.NET
11:00 – 11:10 Дмитрий К.М. Захаров: доклад «PascalABC.NET в старшей школе»
11:10 – 11:20 Виктор Ю.А. (Южурал-Ойл, УРТО) PascalABC.NET в старшей школе

14:00 – 14:05 Михаил С.С. Пучкин: доклад «Математика и информатика»
14:05 – 14:10 Сергей Д. Ионов: доклад «PascalABC.NET на YouTube»
14:10 – 14:20 Александр А.А. (Ульяновский, СПбГУ) Видео ЕГЭ 2023 по информатике на PascalABC.NET
14:20 – 14:30 Дмитрий К.М. Захаров: доклад «PascalABC.NET в старшей школе»
14:30 – 14:40 Виктор Ю.А. (Южурал-Ойл, УРТО) PascalABC.NET в старшей школе

Использование системы программирования PascalABC.NET в обучении программированию
День второй (30.03.2023), часть 1 (14:10 – 16:00)

14:10 – 14:15 Михаил С.С. Пучкин: доклад «Математика и информатика»
14:15 – 14:20 Сергей Д. Ионов: доклад «PascalABC.NET на YouTube»
14:20 – 14:30 Александр А.А. (Ульяновский, СПбГУ) Видео ЕГЭ 2023 по информатике на PascalABC.NET
14:30 – 14:40 Дмитрий К.М. Захаров: доклад «PascalABC.NET в старшей школе»
14:40 – 14:50 Виктор Ю.А. (Южурал-Ойл, УРТО) PascalABC.NET в старшей школе

Конференция
«PascalABC.NET 2023» — ...

Планируемая дата публикации:
29.03.2023, 10:00

Уважаемые коллеги!

Завтра в 10.00 мы начинаем конференцию. Ссылки для входа в Zoom были разосланы по электронной почте.

Ссылки на трансляцию на YouTube:

День 1 (29.03.2023), часть 1: Карякин М.И., Михалкович С.С., Осипов А.В., Богданов А.А., Демяненко Я.М., Ипатов Ю.А.:
<https://www.youtube.com/live/IV3zy4Cjwb0>

День 1 (29.03.2023), часть 2: Михалкович С.С., Пучкин М.В., Кобзарь Д.:
<https://www.youtube.com/live/1xyHEwicxOE>

День 2 (30.03.2023), часть 1: Михалкович С.С., Николаев А.В., Стриженков Р.В., Костин К.К., Дженжер В.О.:
<https://www.youtube.com/live/-orBUe52NIO>

Конференция
«PascalABC.NET 2023» — ...

Планируемая дата публикации:
29.03.2023, 14:10

Уважаемые коллеги!

Завтра в 10.00 мы начинаем конференцию. Ссылки для входа в Zoom были разосланы по электронной почте.

Ссылки на трансляцию на YouTube:

День 1 (29.03.2023), часть 1: Карякин М.И., Михалкович С.С., Осипов А.В., Богданов А.А., Демяненко Я.М., Ипатов Ю.А.:
<https://www.youtube.com/live/IV3zy4Cjwb0>

День 1 (29.03.2023), часть 2: Михалкович С.С., Пучкин М.В., Кобзарь Д.:
<https://www.youtube.com/live/1xyHEwicxOE>

День 2 (30.03.2023), часть 1: Михалкович С.С., Николаев А.В., Стриженков Р.В., Костин К.К., Дженжер В.О.:
<https://www.youtube.com/live/-orBUe52NIO>

Основные языки в школе сегодня:

Python, PascalABC.NET, C++

Сравнение

PascalABC.NET, Python и C++ в школе сегодня

- В школы активно внедряется язык Python (возник в 1990 г.) как высокоуровневый современный язык. Основной мотив – Python широко распространен.
- Старый язык Pascal утратил свои позиции из-за бедности конструкций.
- **PascalABC.NET – новый язык** (возник в 2007 г.). Он предлагает самые современные конструкции языков программирования, адаптированные к школьной информатике.
- PascalABC.NET, C++ – **компилируемые** языки, Python – **интерпретируемый**. Это – разные языки со своими особенностями и проблемами.
- Python – сверхвысокоуровневый, широко рекламируемый, лаконичный. **Но:** скорость работы программ – в 100 раз медленнее PascalABC.NET, ошибки диагностируются лишь при выполнении программ, что приводит к формированию плохого стиля
- C++ – сложный профессиональный язык, основное применение – для олимпиад
- PascalABC.NET – баланс простоты, ранней проверки ошибок и скорости
- **PascalABC.NET и Python – основные языки в школе**, C++ также используется в олимпиадном программировании высокого уровня

Сравнение PascalABC.NET и Python

PascalABC.NET

- Возник в **2007 г.**, изменения с **2015 г.** нацелены на обучение современному программированию
- частично **основан на C#** - 5 язык в рейтинге ЯП
- простой и **компактный синтаксис** (блоки)
- **компилируемый**, высокая скорость выполнения
- **полный контроль ошибок** при компиляции – **хорошо** для начинающих
- простая и мощная **среда программирования** под **Windows и Linux** (intellisense и автоформатирование)
- возможности языка нацелены на **обучение современному программированию**
- среда и сообщения об ошибках на **русском**
- **лёгкий переход** на C#, Java, C++ в силу совпадения языковых конструкций
- встроенные средства для обучения: графика, исполнители, автоматически проверяемые задания

Python

- Возник в **1990 г.**, изменения в языке за последние 10 лет не имеют отношения к обучению
- первый язык в рейтинге ЯП
- **лаконичный синтаксис** (отступы)
- **интерпретируемый**, низкая скорость выполнения
- **нет контроля** семантических ошибок до запуска программы – **плохо** для начинающих
- базовая среда программирования IDLE аскетична, **установка PyCharm сложна** для начинающих
- те возможности, которые появляются в новых версиях, - **не для обучения**
- среда и сообщения об ошибках на **английском**
- **тяжелый переход** на C#, Java, C++ в силу различия языковых конструкций
- нет встроенных средств для обучения

PascalABC.NET и Python – время работы программ

Python

Т.к. Python является интерпретатором, время выполнения программ может быть очень большим.

```
import time
start = time.time()
n = 20000
s = 0.0
for i in range(1,n+1):
    for j in range(1,n+1):
        s += 1.0/(i*j)
end = time.time()
print(s, end - start)
```

43,1 с



PascalABC.NET

Компилятор генерирует эффективный код, сравнимый по быстродействию с кодом на C#, C++

```
##
var n := 20000;
var s := 0.0;
for var i:=1 to n do
for var j:=1 to n do
    s += 1.0/(i*j);
Println(s,Milliseconds);
```

0,37 с



Слайд конференции
2021 г.

В 110 раз
быстрее!

Процессор Intel Core I7-8700K CPU 3.7 ГГц, ОЗУ 16 Гб

Михалкович С.С. Презентация «PascalABC.NET 2023 – российский Паскаль»

Слайд 18

PascalABC.NET и Python – компактность кода

Ввод

```
a = int(input())
```

```
var a := ReadInteger;
```

```
x,y = map(int,input().split())
```

```
var (x,y) := ReadInteger2;
```

Вывод

```
for x in range(10):  
    print(x,end = ' ')
```

```
for var i:=0 to 9 do  
    Print(i);
```

```
for x in range(10):  
    print('*')
```

```
loop 10 do  
    Println('*');
```

Кортежи

```
a, b = b, a
```

```
(a,b) := (b,a);
```

Массивы и списки

```
a = [1,2,3,4,5,6,7,8,9]
```

```
var a := Arr(1..9);
```

```
c = [int(input()) for i in range(10)]
```

```
var c := ReadArrInteger(10);
```

```
a = [2*i+1 for i in range(10)]
```

```
var a := ArrGen(10,i->2*i+1);
```

```
a = [12,3,5,2,41,92,28,35]  
print([x for x in a if x % 2 == 0])
```

```
var a = |12,3,5,2,41,92,28,35|;  
a.Where(x->x.IsEven).Print;
```

Матрицы

```
a = [[(i+1)*(j+1) for i in range(3)] \  
      for j in range(3)]  
print(a)
```

```
c := MatrGen(3,3,(i,j)->(i+1)*(j+1));  
c.Println;
```

Условная операция

```
y = 0 if x < 5 else 1
```

```
y := if x < 0 then 0 else 1;
```

Диапазоны

```
if 3 < x < 7:
```

```
if x in 3..7 then
```

Функции

```
def Add(a,b):  
    return a + b
```

```
function Add(a,b: integer) := a + b;
```

Срезы

```
a = [1,2,3,4,5];  
print(a[1:-1])
```

```
var a = |1,2,3,4,5|;  
Print(a[1:^1]);
```

PascalABC.NET в ЕГЭ. Презентации К.Ю. Полякова

Петя строит все возможные пятибуквенные слова из букв В, Е, Н, О, К. Сколько он может построить слов, в которых буквы Н и К встречаются ровно по два раза?

```
##  
'БЕНОК'.Cartesian(5)  
    .Where( w -> (w.CountOf('Н') = 2) and  
                (w.CountOf('К') = 2) )  
    .Count.Println;
```

90

Артур составляет 5-буквенные коды перестановкой букв слова АРЕАЛ. При этом нельзя ставить рядом две гласные. Сколько различных кодов может составить Артур?

```
## 'АРЕАЛ'.Permutations  
    .Distinct  
    .Where( w->w.Pairwise.All(  
        p-> p[0]+p[1] not in 'ААЕЕА' ) )  
    .Count.Println;
```

6

Автор школьных учебников по информатике К.Ю. Поляков в своих презентациях рассказывает учителям, как эффективно применять PascalABC.NET для сдачи КЕГЭ.

<https://kpolyakov.spb.ru/school/doklad.htm>

Решение «демо-2023».15

```
##  
function f(x, A: integer) :=  
    (x.Divs(2) <= x.notDivs(3)) or (x + A >= 100);  
  
(1..1000)  
    .Where( A -> (1..1000).All( x -> f(x,A) ) )  
    .First  
    .Println;
```

94

Python в ЕГЭ. Презентации К.Ю. Полякова

Петя строит все возможные пятибуквенные слова из букв В, Е, Н, О, К. Сколько он может построить слов, в которых буквы Н и К встречаются ровно по два раза?

```
from itertools import product
words = [ w for w in product('ВЕНОК', repeat=5)
          if w.count('Н') == 2 and
            w.count('К') == 2 ]
print( len(words) )
```

90

Артур составляет 5-буквенные коды перестановкой букв слова АРЕАЛ. При этом нельзя ставить рядом две гласные. Сколько различных кодов может составить Артур?

```
from itertools import permutations
pairsArr = ( zip(w, w[1:])
              for w in set( permutations('АРЕАЛ') ) )
print( sum( 1 for pairs in pairsArr
            if all( p[0]+p[1] not in 'АЕЕЕА'
                    for p in pairs ) ) )
```

6

Автор школьных учебников по информатике К.Ю. Поляков в своих презентациях рассказывает учителям, как эффективно применять Python для сдачи КЕГЭ.

<https://kpolyakov.spb.ru/school/doklad.htm>

Решение «демо-2023».15

```
def f(x, A):
    return ((x % 2 == 0) <= (x % 3 != 0))
           or (x + A >= 100)
print( [
    A for A in range(1,1001)
        if all( f(x,A) for x in range(1,1001))
    ][0] )
```

94

Диагностика ошибок

Python

Перед запуском проверяется соответствие грамматике. Контроль области видимости переменных и контроль типов осуществляются на этапе выполнения (для обучения **это плохо** – лучше чтобы как можно больше ошибок отлавливались автоматически)

```
2.py - C:\Python\Python37-32\2.py (3.7.1)
File Edit Format Run Options Window Help
a = 1
if a < 2:
    print ('ok')
elif bla( bla )[bla]:
    bla [bla:bla:bla, bla:bla]
    print(b)
elif bla.bla().bla:
    bla.bla ( bla )
    print(c)
else:
    prnt('Bad')
```

Ln: 8 Col: 19

Этот код сработает
без ошибок 😞

PascalABC.NET

Слайд конференции
2021 г.

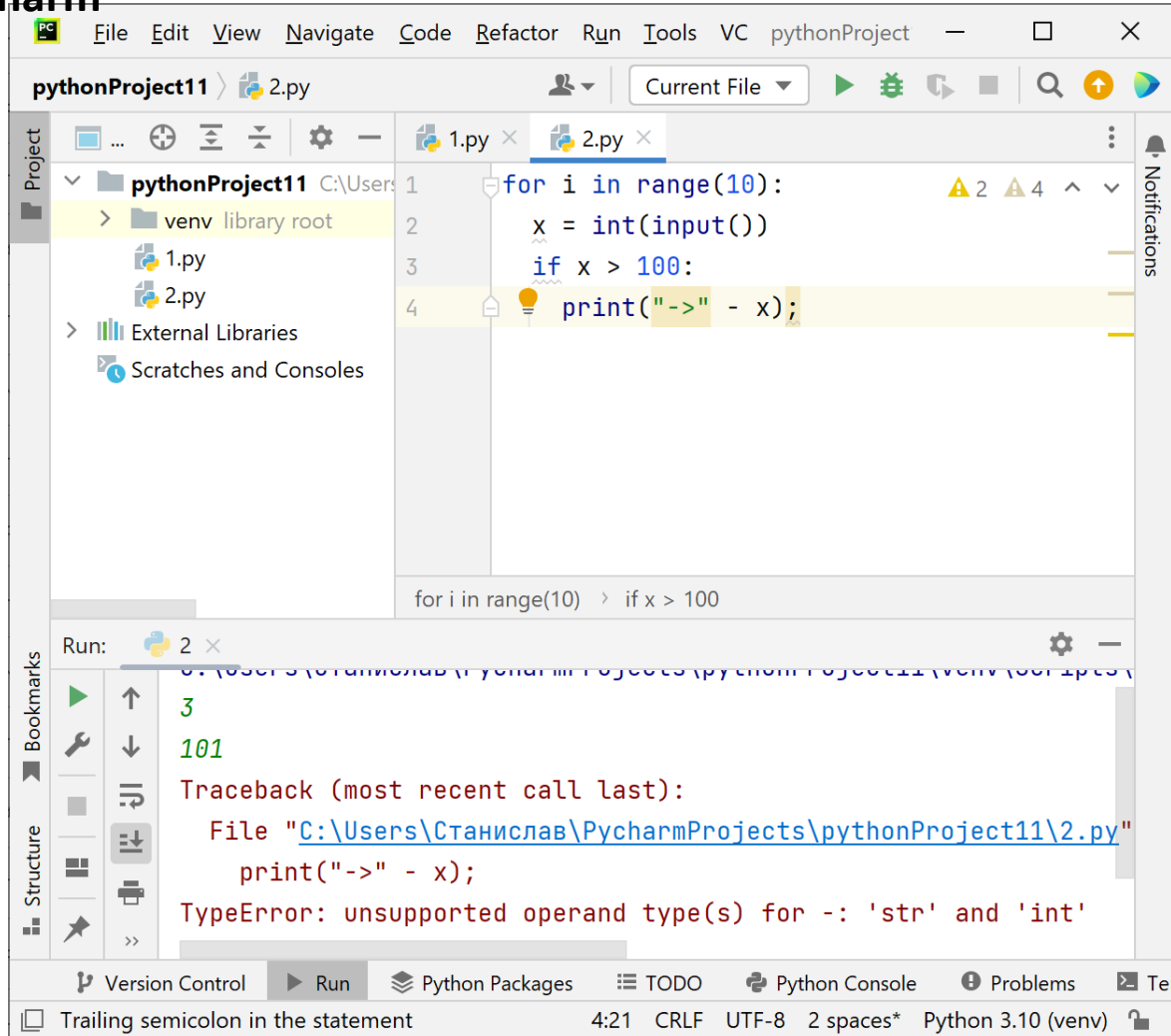
Перед запуском компилятор проверяет соответствие грамматике, осуществляет контроль области видимости переменных и контроль типов. Множество ошибок компиляции препятствует написанию неправильных программ на раннем этапе и приучает к строгой культуре кода (**это хорошо**)

```
##
var a := 1;
if a < 2 then
    Print('Ok')
else if bla(bla)[bla] then begin
    bla[bla:bla:bla, bla:bla];
    Print(b)
end
else if bla.bla().bla then
begin
    bla.bla(bla);
    Print(c)
end
else Prnt('Bad')
```

Этот код
компилятор не
пропустит из-за
многочисленных
ошибок 😊

PascalABC.NET и Python – диагностика ошибок

PyCharm



The screenshot shows the PyCharm IDE with a Python project named 'pythonProject11'. The main editor displays a file '2.py' with the following code:

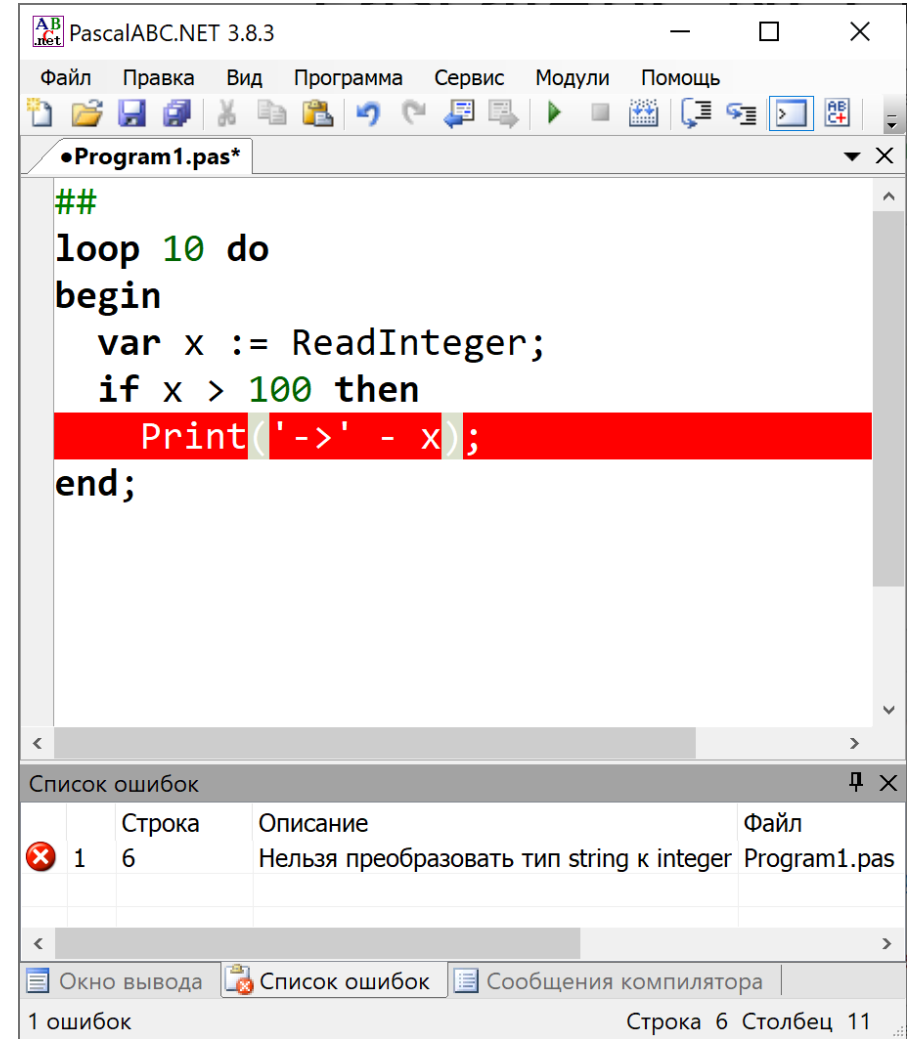
```
1 for i in range(10):
2     x = int(input())
3     if x > 100:
4         print("->" - x);
```

The code is highlighted with a yellow background. The bottom status bar indicates 'Trailing semicolon in the statement'.

The Run window at the bottom shows the execution output:

```
Run: 2 x
3
101
Traceback (most recent call last):
  File "C:\Users\Станислав\PycharmProjects\pythonProject11\2.py"
    print("->" - x);
TypeError: unsupported operand type(s) for -: 'str' and 'int'
```

Ошибка во время выполнения (**поздно!**)



The screenshot shows the PascalABC.NET IDE with a project named 'PascalABC.NET 3.8.3'. The main editor displays a file 'Program1.pas' with the following code:

```
##
loop 10 do
begin
    var x := ReadInteger;
    if x > 100 then
        Print("->" - x);
end;
```

The code is highlighted with a red background. The bottom status bar indicates '1 ошибок' (1 error).

The List of Errors window at the bottom shows the following error:

Строка	Описание	Файл
6	Нельзя преобразовать тип string к integer	Program1.pas

Ошибка компиляции (**рано!**)

PascalABC.NET – базовый уровень. Обязательные отличия от стандартного Паскаля

Оператор вывода Print

В PascalABC.NET для вывода в основном используется процедура Print, разделяющая элементы вывода пробелами. Это удобно и при выводе отдельных элементов и при выводе в цикле

Пример 1.

```
begin
    Print(1,2,3);
end.
```

1 2 3

Пример 2.

```
begin
    for var i:=1 to 10 do
        Print(i)
    end.
```

1 2 3 4 5 6 7 8 9 10

```
# python
for x in range(10):
    print(x,end = ' ')
```

Автовывод типа при описании переменной

В PascalABC.NET при описании переменной с инициализацией можно не указывать **тип** – он **выводится автоматически** по типу правой части и сокращает программу. Автовывод типа имеется практически во всех современных языках: C#, C++, Kotlin и пр.

Кроме того, такой стиль похож на Python, где тип вообще не указывается явно.

Это упрощает восприятие первых программ начинающими

Пример 1.

```
begin
  var n := 0;           // integer
  var r := 2.5;         // real
  var s := 'PascalABC'; // string
end.
```

```
# python
n = 0
r = 2.5
s = 'Python'
```

Пример 2.

```
begin
  var r := 1/2;         // real
  var s := '*' * 3;     // string
  var a := |1,2,3|;     // array of integer
end.
```

Цикл loop

Цикл loop используется когда надо выполнить некоторые действия известное количество раз и не нужен счетчик цикла.

Цикл loop очень прост для начинающих: **loop 10 do** – повторить 10 раз

Пример 1.

```
begin
  loop 10 do
    Print('*')
  end.
```

```
# python
for x in range(10):
    print('*',end = ' ')
```

Пример 2.

```
uses Robot;

begin
  loop 10 do
    begin
      Paint; Right
    end
  end.
```

Операции += и *=

Операции += и *= есть во многих языках и хорошо читаются:

+= - «увеличить на»

*= - «увеличить в»

Пример 1.

```
begin
  var p := 5;
  p += 2; // 7
end.
```

Пример 2.

```
begin
  var p := 5;
  p *= 2; // 10
end.
```

Множественная инициализация при описании и множественное присваивание

Можно описать сразу несколько переменных и присвоить им значения – их типы автовыведутся.

Можно присвоить значения одновременно нескольким переменным

Пример 1.

```
begin
  var (a,b,c) := (3,5,7);
  (a,b) := (b,a); // поменять местами
  (a,b,c) := (b,c,a);
  // циклический сдвиг влево
end.
```

```
# python
a,b,c = 3,5,7
a,b,c = b,c,a
```

Пример 2.

```
begin
  var (a,b) := (72,66);
  while b > 0 do
    (a,b) := (b,a + b);
    Print(a); // НОД
  end.
```

```
# python
while b>0:
  a,b = b,a + b
```

Функции ввода ReadInteger, ReadReal2, ...

Можно в одну строку описать переменную с автовыводом типа и инициализировать её вводом с клавиатуры. Для ввода одной переменной базового типа используются функции ReadInteger, ReadReal, ..., для ввода двух переменных – ReadInteger2, ReadReal2 и т.д.

Это сокращает программы и позволяет не описывать типы явно.

Пример 1.

```
begin
  var n := ReadInteger;
  var a := ReadReal('Введите a:');
end.
```

```
# python
a = int(input())
```

Пример 2.

```
begin
  var (a,b) := ReadReal2('Введите a,b:');
end.
```

```
# python
a,b = map(int,input().split())
```

Диапазоны и операция in

В PascalABC.NET появились диапазоны вида a..b. Основное их использование для начинающих – проверка попадания в диапазон с помощью операции in:

Пример 1.

```
begin
    Print(5 in 2..10); // True
end.
```

Пример 2.

```
begin
    var x := ReadInteger;
    if x in 2..5 then
        Print('Это оценка');
    end.
```

```
# python
if 2 <= x <= 5:
    print('Это оценка')
```

Внутриблочные описания переменных

Переменные в PascalABC.NET описываются преимущественно внутри блока (как и во многих других языках). Глобальные переменные, описываемые до `begin` основной программы, описываются крайне редко.

Пример 1. n!

```
begin
  var n := 20;
  var p: BigInteger := 1;
  for var i:=1 to n do
    begin
      Print(p);
      p *= i
    end;
end.
```

```
1 1 2 6 24 120 720 5040 40320 362880
3628800 39916800 479001600 6227020800
87178291200 1307674368000 20922789888000
355687428096000 6402373705728000
121645100408832000
```

Пример 2. Сумма чисел, попадающих в диапазон 10..20

```
begin
  var n := ReadInteger('Введите n:');
  var sum := 0;
  loop n do
    begin
      var x := ReadInteger;
      if x in 10..20 then
        sum += x;
      end;
    end;
    Print('Сумма чисел в диапазоне 10..20 =', sum);
end.
```

```
Введите n: 5
7 10 20 47 55
Сумма чисел в диапазоне 10..20 = 30
```

Цикл for с шагом

В PascalABC.NET появился цикл for с шагом, в т.ч. и отрицательным. Это расширяет спектр задач для начинающих.

Пример 1.

```
begin
  for var i:=20 to 1 step -2 do
    Print(i);
end.
```

20 18 16 14 12 10 8 6 4 2

Пример 2.

```
uses GraphWPF;

begin
  var y := 300;
  for var x:=50 to 450 step 50 do
    Circle(x,y,20,RandomColor)
  end.
```



Командный режим записи программ

Командный режим записи программ начинается с **##** и позволяет опускать внешний begin-end. Он служит для **коротких программ** в несколько строк и используется обычно в **демонстрационных целях**

```
##  
var (a,b) := ReadInteger2;  
Print(a + b, a * b);
```

```
##  
function SumSquares(a,b: integer) := a*a + b*b;  
  
Print(SumSquares(3,4));
```

```
##  
Print(BigInteger(2) ** 100);
```

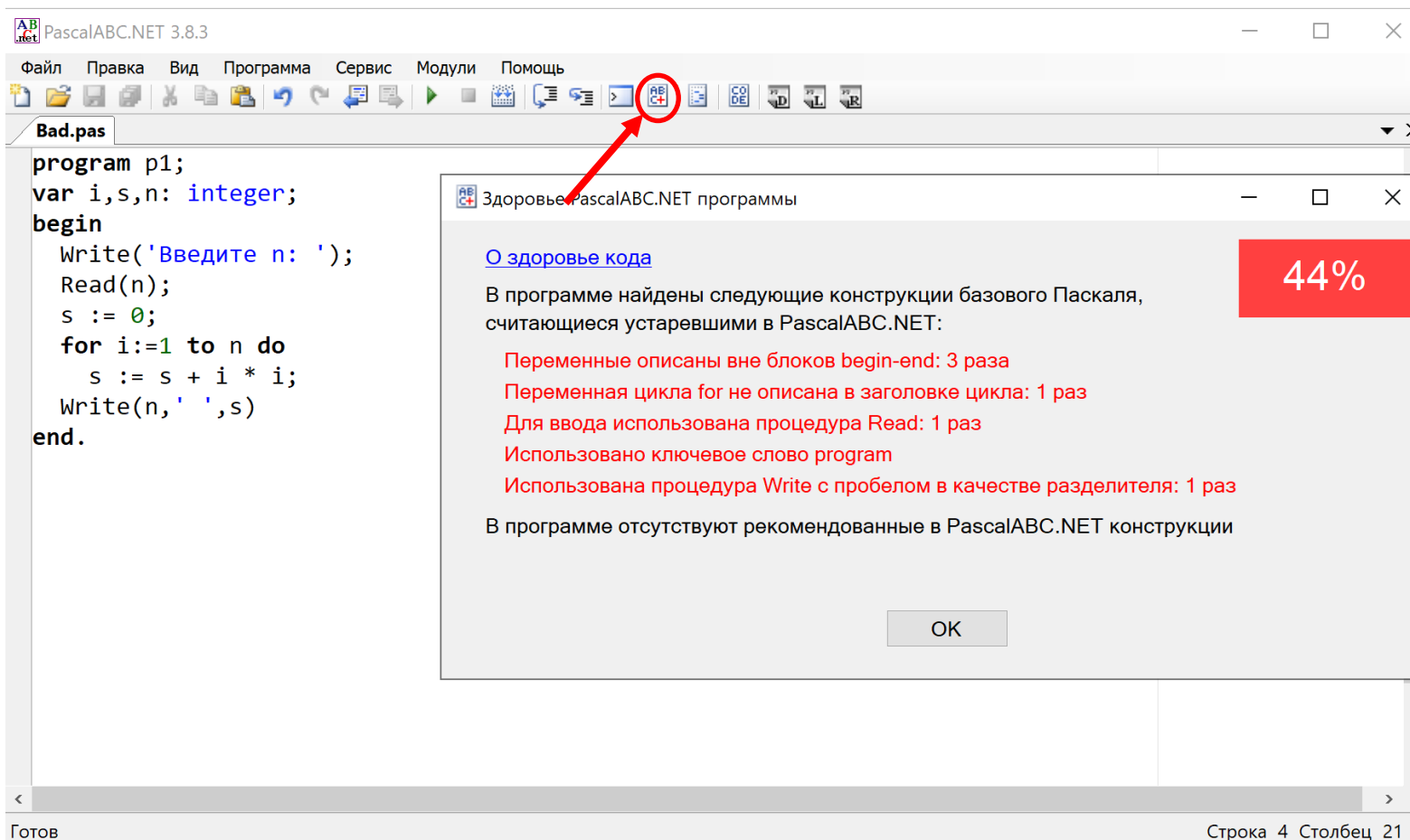
```
##  
uses Graph3D;  
Sphere(0,0,0,2);
```

```
##  
uses GraphWPF;  
Circle(100,100,50,Colors.Green);
```

```
##  
uses School;  
var x := 123456;  
Println(Bin(x));
```

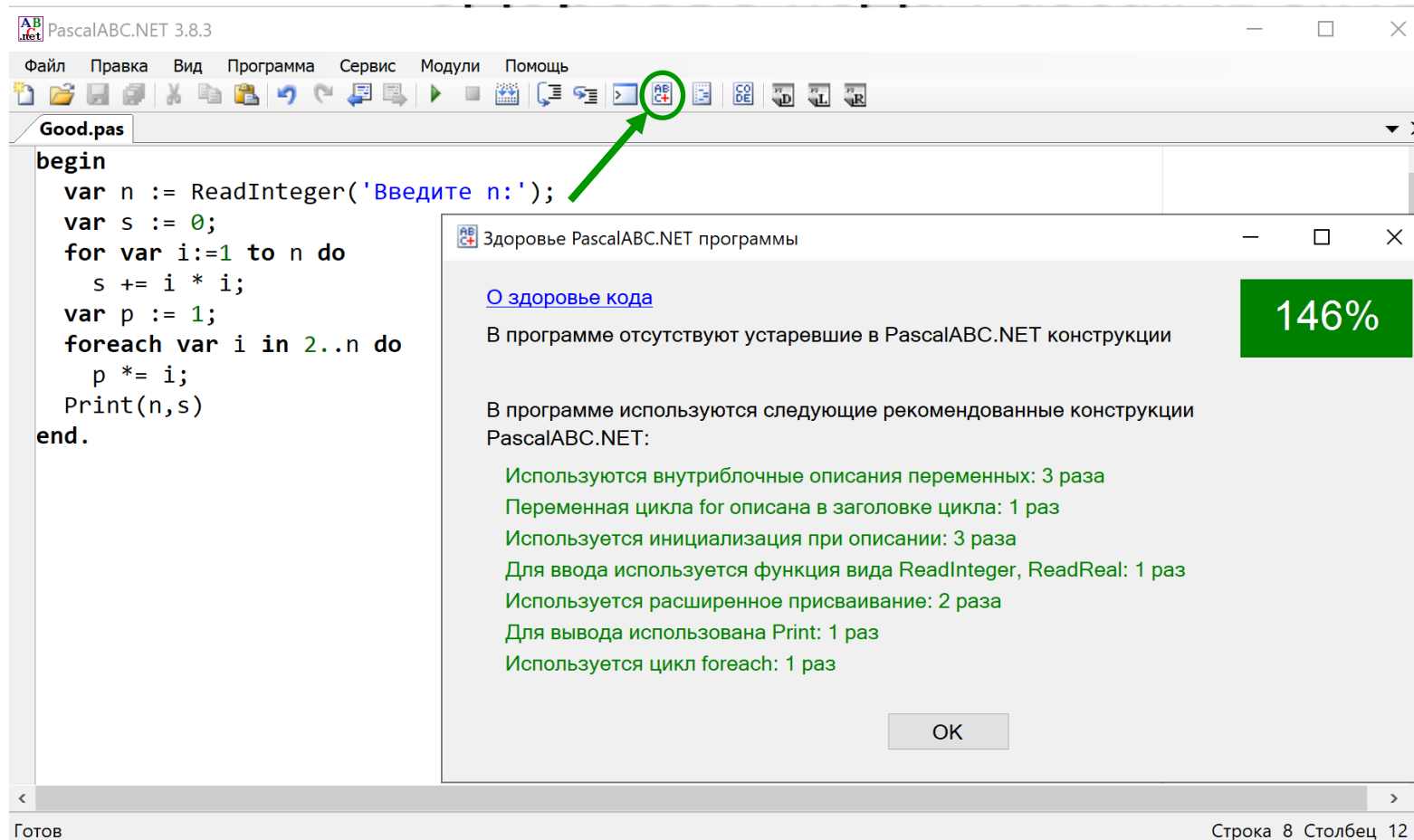
Здоровье кода PascalABC.NET – контроль базовых конструкций

Здоровье кода PascalABC.NET - показатель качества кода программы. Если в программе используются устаревшие конструкции, здоровье кода снижается



Здоровье кода PascalABC.NET – контроль базовых конструкций

Здоровье кода PascalABC.NET - показатель качества кода программы. Если в программе используются новые средства PascalABC.NET, здоровье кода увеличивается



PascalABC.NET – углубленный уровень. Проблемы методики

ФГОС для школ

Приказ Министерства просвещения РФ от 31 мая 2021 г. № 287 «Об утверждении федерального государственного образовательного стандарта основного общего образования»

45.5.4. По учебному предмету «Информатика» (на **углубленном** уровне):

7) свободное оперирование понятиями: переменная, **тип данных**, операция присваивания, арифметические и логические операции, включая операции целочисленного деления и остатка от деления; умение создавать программы на современном языке программирования общего назначения, реализующие алгоритмы обработки числовых данных с использованием ветвлений, циклов со счетчиком, циклов с условиями, **подпрограмм** (алгоритмы проверки делимости одного целого числа на другое, проверки натурального числа на простоту, разложение на простые сомножители, выделение цифр из натурального числа, поиск максимумов, минимумов, суммы числовой **последовательности** и т.п.)

Акцентируем внимание!

PascalABC.NET разработан именно для обучения современному программированию на **углубленном** уровне

типы данных

В **PascalABC.NET**, **C++** типы данных можно задавать явно, что методически правильно. Имеется также автоматический вывод типов: свобода выбора методики
В **Python** типы данных – неявные, у ученика остаётся слабое представление о них

подпрограммы

Процедуры, функции: $f(a)$

Методы классов: $x.ToString$

Лямбда-функции (безымянные функции): $x \rightarrow x*x$

последовательности

В **PascalABC.NET** имеется тип последовательности: **sequence of T**.
Массивы, множества, списки – это разновидности последовательностей

Углубленный уровень. С чего начать. Как продолжить

Общие вопросы

- Слишком много новых элементов – трудно выстроить методику
- В интернете – примеров мало, некоторые примеры плохие, сложные, неправильные, непонятные
- Набор методик есть, для этого надо усвоить базовые понятия

Компактность кода – как найти баланс?

Последовательность – важнейшее новое понятие – это набор значений, идущих одно за другим

- Массивы и списки – конкретные разновидности последовательностей
- Есть много **методов последовательностей**. Переходы между **методами и алгоритмами** – разные методики в зависимости от целей и аудитории. Где найти **баланс**

Лямбда-функции (безымянные функции) – важнейший элемент современного программирования. Это – функции, создаваемые на лету – обычно как параметры методов

- Лямбда-функции делятся на **лямбда-преобразования**, **лямбда-условия** и **лямбда-проекции**

$x \rightarrow x * x$

$x \rightarrow x > 5$

$x \rightarrow x.Name$

Короткий код – панацея или проклятье?

В PascalABC.NET многое сделано для того чтобы код стал **лаконичным**. Это – основа популярности Python. Но всегда ли лаконичный код – хорош? Где – граница понятности и **разумности** лаконичного кода?

```
##  
var (a, b) := (0, 1);  
(a, b) := (a + 1, b * 2);
```

```
begin  
  var a := 0;  
  var b := 1;  
  a += 1;  
  b *= 2;  
end.
```

```
var a: integer := 0;  
var b: integer := 1;  
begin  
  a := a + 1;  
  b := b * 2;  
end.
```

```
###  
var a := Arr(1..10);  
var b := a.Wh(x -> x mod 2 = 0).ToA;  
b.Pr  
  
begin  
  var a := Arr(1..10);  
  var b := a.Where(x -> x mod 2 = 0).ToArray;  
  b.Print  
end.
```

```
var a,b: array[0..9] of integer;  
    bn: integer := 0;  
begin  
  for var i:=0 to 9 do  
    a[i] := i + 1;  
  for var i:=0 to 9 do  
    if a[i] mod 2 = 0 then  
      begin b[bn] := a[i]; bn := bn + 1 end;  
  for var i:=0 to bn-1 do  
    Print(b[i])  
end.
```

Пример 1. Длины слов

Задача. Дан массив слов. Получить последовательность длин слов

Комментарий. Задача – стандартная – на преобразование элементов массива

Алгоритм:

```
var a := Arr('холодный', 'белый', 'плохой', 'злой', 'спелый');  
foreach var word in a do  
    Print(word.Length);
```

Словесное описание. Мы видим цикл по массиву, преобразование каждого слова массива в его длину

Анализ. У массива есть стандартный метод преобразования **Select**, его параметром служит функция преобразования элемента, записываемая в виде лямбда-выражения `word -> word.Length`. Для печати используем метод последовательности `Print`:

```
a.Select(word -> word.Length).Print
```

Методика. Можно начинать с алгоритма и просить написать по нему метод.

Можно наоборот писать код вызова метода и просить написать по нему алгоритм (цикл)

Пример 1а. Минимальная длина слова

Задача. Дан массив слов. Найти слово минимальной длины.

Комментарий. Две методики: написать алгоритм и по нему составить код на основе стандартных методов. И наоборот: привести код с цепочкой методов и попросить по нему реализовать алгоритм

Алгоритм:

```
var a := Arr('холодный', 'белый', 'плохой', 'злой', 'спелый');  
var minlen := integer.MaxValue; // это значение изменится  
foreach var word in a do  
    if word.Length < minlen then  
        minlen := word.Length;
```

Анализ. Алгоритм, хотя и стандартный, становится ощутимо сложнее. Для решения с помощью методов используем предыдущий код и вместо вывода последовательности длин слов вычисляем её минимальный элемент с помощью метода Min:

```
a.Select(word -> word.Length).Min
```

Вывод. Методы проще и гибче.

Методика. Один раз показать алгоритм, затем закрепить его, а потом вызывать методы для решения сходных задач.

Пример 16. Минимальная длина слова в строке

Задача. Дана строка слов. Найти слово минимальной длины.

Комментарий. Эта задача минимально отличается от предыдущей. Надо как-то разбить строку на массив слов. Сделать это алгоритмом в рамках школьных занятий сложно и не нужно. Поэтому применяем стандартный метод разбиения на слова `s.ToWords`

Цепочка методов:

```
var s := 'холодный белый плохой злой спелый';  
s.ToWords.Select(word -> word.Length).Min.Print
```

Анализ. Минимальное число изменений по сравнению с предыдущим кодом. Метод `ToWords` невозможно реализовать в рамках школьного урока.

Алгоритм.

```
var s := 'холодный белый плохой злой спелый');  
var a := s.ToWords;  
var minlen := integer.MaxValue; // это значение изменится  
foreach var word in a do  
    if word.Length < minlen then  
        minlen := word.Length;
```

Пример 2. Фильтрация элементов

Задача. Дан массив чисел. Отфильтровать в нем элементы от 2 до 5.

Комментарий. Задача – стандартная, но проблемная. Если результат записывать в массив, то непонятен размер этого массива. Поэтому выведем все элементы на экран.

Алгоритм:

```
foreach var x in a do
  if x in 2..5 then
    Print(x)
```

Словесное описание. Мы видим цикл по массиву, условие отбора элемента и вывод элемента, соответствующего условию отбора.

Анализ кода. У массива есть стандартный метод фильтрации **Where**, его параметром служит лямбда функция-условие $x \rightarrow x \text{ in } 2..5$. Затем мы печатаем получившуюся последовательность с помощью метода `Print`

```
a.Where(x -> x in 2..5).Print;
```

Методика. Мы вновь можем начать с кода цепочки методов и попросить реализовать по нему алгоритм. Можем наоборот попросить составить алгоритм, а потом вспомнить метод фильтрации, понять его параметр и написать код с цепочкой методов

Пример 3. Сумма отфильтрованных элементов

Задача. Дан массив. Отфильтровать в нем элементы от 2 до 5 и просуммировать их

Алгоритм. Изменим алгоритм предыдущей задачи, используя вместо вывода накопление суммы:

```
var sum := 0;  
foreach var x in a do  
  if x in 2..5 then  
    sum += x
```

Переход к методам. Мы знаем, что к массиву нужно применить метод фильтрации Where с параметром – лямбда-условием `x -> x in 2..5`. После этого мы должны просуммировать отфильтрованные элементы стандартным методом Sum:

```
var sum := a.Where(x -> x in 2..5).Sum;  
Print(sum);
```

Комментарий. Код для решения с методом изменился совсем незначительно – добавился метод .Sum, а алгоритм пришлось ощутимо менять

Пример 3. Гибридное решение

Задача. Дан массив. Отфильтровать в нем элементы от 2 до 5 и просуммировать их

Решение. Совместим в решении метод Where с нахождением суммы с помощью алгоритма:

```
var sum := 0;  
foreach var x in a.Where(x -> x in 2..5) do  
    sum += x
```

Решение 2. Чтобы заголовок цикла foreach не был таким длинным, поместим отфильтрованные данные в отдельную переменную:

```
var sum := 0;  
var filtered := a.Where(x -> x in 2..5);  
foreach var x in filtered do  
    sum += x
```

Решение 3. Методами – в 2 этапа.

```
var filtered := a.Where(x -> x in 2..5);  
Print(filtered.Sum,filtered.Average,filtered.Min);
```

Пример 3. One Liner

Задача. Дан массив. Отфильтровать в нем элементы от 2 до 5 и просуммировать их

Решение в одну строку:

```
a.Where(x -> x in 2..5).Sum.Print
```

Комментарий. Если чувствуешь себя уверенно в методах и цель – как можно скорее получить результат, то это – хорошее решение.

Комментарий. В PascalABC.NET для всех элементарных типов определен метод .Print.

Полный код

```
##  
var a := ArrRandomInteger(10,1,10);  
a.Println;  
a.Where(x -> x in 2..5).Sum.Print
```

```
5 5 3 4 3 9 4 8 10 6  
24
```

Что дальше? Мы можем решать более сложные задачи!

Освоив какую-то технику, следует двигаться дальше.

Использование методов не только укорачивает код, но и позволяет эффективно решать более сложные задачи. На основе прошлого примера можно решить:

- Чего в массиве больше - четных или нечетных?
- **Что больше - сумма четных или сумма нечетных?**
- Среди четных найти сумму первого и последнего

Решим для примера вторую задачу, используя методы:

```
begin
  var a := ArrRandomInteger(10);
  a.Println;
  var filteredOdds := a.Where(x->x mod 2 <> 0);
  var filteredEvens := a.Where(x->x mod 2 = 0);
  if filteredOdds.Sum > filteredEvens.Sum then
    Print('сумма нечетных больше')
  else Print('сумма четных больше')
end.
```

Выводы

1. PascalABC.NET – новый язык программирования общего назначения, разработанный в России для обучения современному программированию
2. PascalABC.NET максимально адаптирован к школьному образованию: **базовый** и **углубленный** уровни обучения информатике в школах, **олимпиадное** программирование, использование в **ЕГЭ**
3. PascalABC.NET имеет просто устанавливаемую графическую оболочку под Windows и Linux
4. PascalABC.NET основан на платформе .NET и языке C#, входящим в Top-5 популярных языков программирования
5. PascalABC.NET – многогранный **развивающийся** язык с большим количеством возможностей.

Важные ссылки

[Книги Осипова А.В. по PascalABC.NET](#)

[Курсы на Stepik](#)

[Статьи, примеры на PascalABC.NET](#)

[Официальный Телеграм-канал PascalABC.NET](#)

[Телеграм-группа для обсуждения PascalABC.NET](#)

[Сравнение Python и PascalABC.NET](#) (доклад 2021 г.)

[PascalABC.NET 2021](#) (доклад 2021 г.)

[Вся правда о лямбда-выражениях](#) (доклад 2021 г.)

[Динамические массивы в языке PascalABC.NET](#) (доклад 2021 г.)

[Презентация курса на Stepik «PascalABC.NET: современный код»](#) (доклад 2021 г.)

[Модуль School базовых алгоритмов информатики](#) (доклад 2021 г.)

[PascalABC.NET Notebooks – интерактивные блокноты и их возможности](#) (доклад 2021 г.)

[Методы расширения в PascalABC.NET](#) (доклад 2020 г.)

[Графика... Графика? Графика!](#) (доклад 2020 г.)



Михалкович С.С.

*Южный федеральный университет,
Институт математики, механики и компьютерных наук*

miks@sfedu.ru

