



# PascalABC.NET 2025

Михалкович С.С.

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

[miks@sfedu.ru](mailto:miks@sfedu.ru)

Доклад на пятой Всероссийской научно-методической конференции  
«Использование системы программирования PascalABC.NET  
в обучении программированию»  
(Ростов-на-Дону, 28-29 марта 2025 г.)

# Позиционирование PascalABC.NET

**PascalABC.NET (2007 – 2025)** – язык программирования Паскаль нового поколения, развивающий идеи современных языков программирования:

- язык для **обучения современному программированию**
- инструмент для **сферы образования** и **научных исследований**
- **аккумулирует лучшие языковые конструкции и библиотечные методы** из C#, Python, Kotlin, Java, Haskell
- **лаконичный и понятный**, конкурируя выразительности с Python
- **язык компиляционного типа**: синтаксические и семантические ошибки проверяются до запуска программы, что повышает качество разработки
- **эффективный**: программы выполняются в 100 раз быстрее, чем на Python, и лишь на 20-30% медленнее, чем на C++
- обеспечивает **легкий переход на промышленные языки** C#, Java, Python, C++

# Основные идеи при разработке PascalABC.NET

Система программирования **PascalABC.NET** ориентирована прежде всего на обучение и образовательную сферу. Она постоянно развивается, объединяя в себе лучшие черты современных языков программирования:

- Строгость **Delphi** (язык №10 в рейтинге Tiobe, лучший для быстрого создания десктопных приложений под Windows)
- Огромное количество современных конструкций и вся стандартная библиотека – от языка **C#** (язык №5 в рейтинге Tiobe, лучший для Windows, Unity и бэкенда)
- Краткость и лаконичность кода – от языка **Python** (язык №1 в рейтинге Tiobe). При этом скорость работы – в 100 раз быстрее Питон-программ
- Ряд библиотечных конструкций – от языка **Kotlin** (язык №1 при разработке под Андроид – официальная рекомендация Google)
- Некоторые функциональные конструкции и функции – от языка **Haskell** (язык №1 в академическом функциональном программировании)

# PascalABC.NET – в России и для России

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

PascalABC.NET обладает мощной методической поддержкой: в компьютерной школе мехмата ЮФУ разрабатываются современные методики обучения программированию, оказывающие влияние на сам язык и его библиотеки.

PascalABC.NET имеет активное сообщество: постоянно действующий форум поддержки и Telegram-канал на **русском** языке.

Простая **русскоязычная** среда PascalABC.NET включает сообщения об ошибках и справку на **русском** языке, а также поддерживает использование имен переменных и файлов на **кириллице**.

включен в дистрибутивы **российских** Linux-систем:

- Alt Linux Образование – версия 3.9
- Astra Linux Orel Special Edition
- М ОС (Москва) – версия 3.10.3

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

# Области использования PascalABC.NET

- В **общем образовании** – для обучения программированию на **базовом** и **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 – 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 550 работ в РИНЦ: сайт [elibrary.ru](http://elibrary.ru))
- Как средство программирования «**на каждый день**»

# События 2023-25 гг., связанные с PascalABC.NET

# Воскресная компьютерная школа мехмата ЮФУ

Воскресная компьютерная школа мехмата ЮФУ – ключевая площадка для развития системы программирования PascalABC.NET, включая методики и сервисы.

Ежегодно ВКС выпускает около **400 юных программистов** (6–11 классы). **Основным языком** обучения является **PascalABC.NET**, а в качестве дополнительных используются **Python** и **C#**.



# Направление ФИИТ мехмата ЮФУ

**Фундаментальная информатика и информационные технологии** – направление мехмата ЮФУ, на которое каждый год поступает 110 студентов. В первом семестре курс **Основы программирования** ведётся на PascalABC.NET, затем изучаются C# и C++.

Язык Python органично входит в курс машинного обучения.



# IV научно-практическая конференция PascalABC.NET

Более 500 участников, 16 докладов, 2 дня конференции, трансляция в интернет



**Нейронная сеть**  
PascalABC.Net в старшей школе. Per aspera ad astra  
32 просмотра • 12 дней назад



**Воскресная компьютерная школа мехмата ЮФУ**  
Михалкович С.С. «PascalABC.NET 2023 – российский Паскаль»  
276 просмотров • 1 год назад



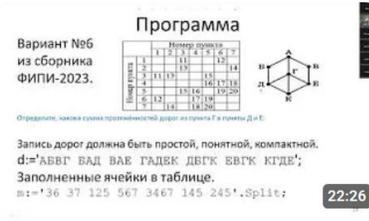
**Приложения с оконным интерфейсом**  
Костин К.К. Приложения с оконным интерфейсом в обучении школьников.  
28 просмотров • 12 дней назад



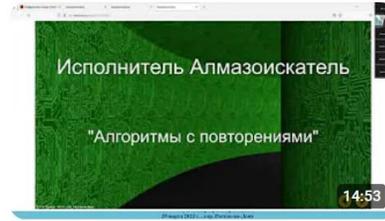
**Алгоритм действий**  
Стриженков Р.В. DL Club – методика обучения программированию младших...  
11 просмотров • 12 дней назад



**Решение 1. Минимум новых возможностей; длинная запись условия**  
Запрещенные методы  
204 просмотра • 1 год назад



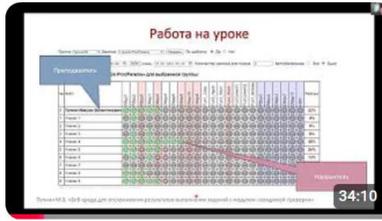
**Программа**  
Решение задачи из ЕГЭ №1 «Города и дороги» на PascalABC.Net с помощью...  
34 просмотра • 12 дней назад



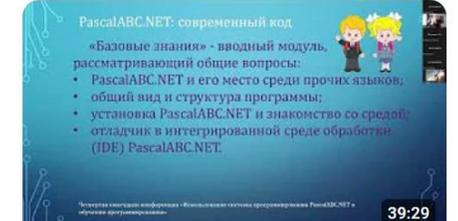
**Исполнитель Алмазоискатель**  
Использование мультимедийной игровой среды разработки блок-схем  
23 просмотра • 12 дней назад



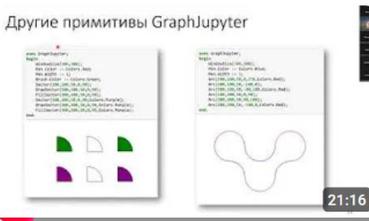
**Особенности нового модуля проверки Light**  
Модуль невидимой автоматической проверки заданий  
119 просмотров • 1 год назад



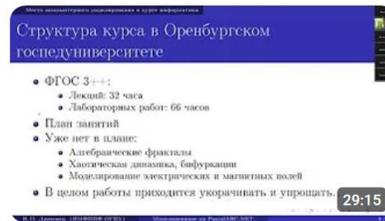
**Работа на уроке**  
Веб-среда для отслеживания результатов выполнения заданий с модулем...  
85 просмотров • 1 год назад



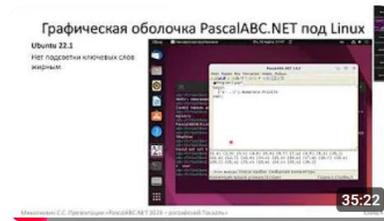
**PascalABC.NET: современный код**  
PascalABC.NET на Stepik  
90 просмотров • 1 год назад



**Графика PascalABC.NET для Jupyter Notebook**  
Графика PascalABC.NET для Jupyter Notebook  
101 просмотр • 1 год назад



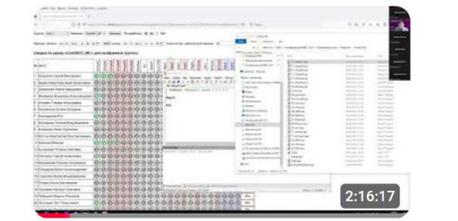
**Структура курса в Оренбургском госуниверситете**  
Моделирование физических и математических задач с использованием...  
152 просмотра • 1 год назад



**Графическая оболочка PascalABC.NET под Linux**  
Михалкович С.С. PascalABC.NET под Linux – что нас ждёт  
529 просмотров • 1 год назад



**А.А. Богданов – Alex Danov**  
Весь ЕГЭ 100% кодом на PascalABC.NET  
406 просмотров • 1 год назад



**Мастер-класс по использованию легковесного задачника LightPT**  
Мастер-класс по использованию легковесного задачника LightPT  
143 просмотра • 1 год 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.

Рейтинг > Компьютеры > Программирование

НАЗВАНИЕ САЙТА
1. Ru.Board - Компьютерный портал <a href="http://forum.ru-board.com">http://forum.ru-board.com</a>
2. Excel для всех <a href="http://www.excel-vba.ru">http://www.excel-vba.ru</a>
3. Мир MS Excel Всё об Excel <a href="http://www.excelworld.ru/">http://www.excelworld.ru/</a>
4. Программист 1С в Минске <a href="http://koder.by/">http://koder.by/</a>
5. Исходники.RU <a href="http://www.sources.ru">http://www.sources.ru</a>
6. MicroTechnics. Микроконтроллеры, программирование <a href="https://microtechnics.ru/">https://microtechnics.ru/</a>
7. Система программирования PascalABC.NET <a href="https://pascalabc.net">https://pascalabc.net</a>

- Посещаемость – **30000** посетителей в месяц (-25% за 2 года). Пользователи из Беларуси, Молдовы, США, Туркменистана
- В рейтинге mail.ru входит в **Топ-7** в категории Программирование (Топ-5 2 года назад)

ГЕОГРАФИЯ	Февраль 2025	ПОСЕТИТЕЛИ
<input checked="" type="checkbox"/> Россия	22,595	70.80%
<input checked="" type="checkbox"/> Беларусь	4,017	12.59%
<input checked="" type="checkbox"/> Республика Молдова	952	2.98%
<input checked="" type="checkbox"/> Украина	692	2.17%
<input checked="" type="checkbox"/> Соединенные Штаты Америки	673	2.11%
<input type="checkbox"/> Туркменистан	442	1.38%
<input type="checkbox"/> Германия	311	0.97%
<input type="checkbox"/> Китай	309	0.97%
<input type="checkbox"/> Финляндия	283	0.89%
<input type="checkbox"/> Казахстан	226	0.71%

# Количество скачиваний PascalABC.NET по данным официальных сайтов

2025 год: **6.5 миллионов**  
**скачиваний** с начала проекта,  
**900** скачиваний в день  
(-40% за 2 года)



Сегодня Вчера Неделя Месяц Квартал Год 10 мар 2025

Сегмент: 1 условие × Выборка: 100%

Визиты, в которых Поведение • Роботность: Только люди + для людей, у кото

## 1. PascalABCNETSetup

Конверсия	46,8 %
Достижения цели	897
Целевые визиты	678

## 2. PascalABCNETMiniSetup

Конверсия	6,42 %
Достижения цели	122
Целевые визиты	93

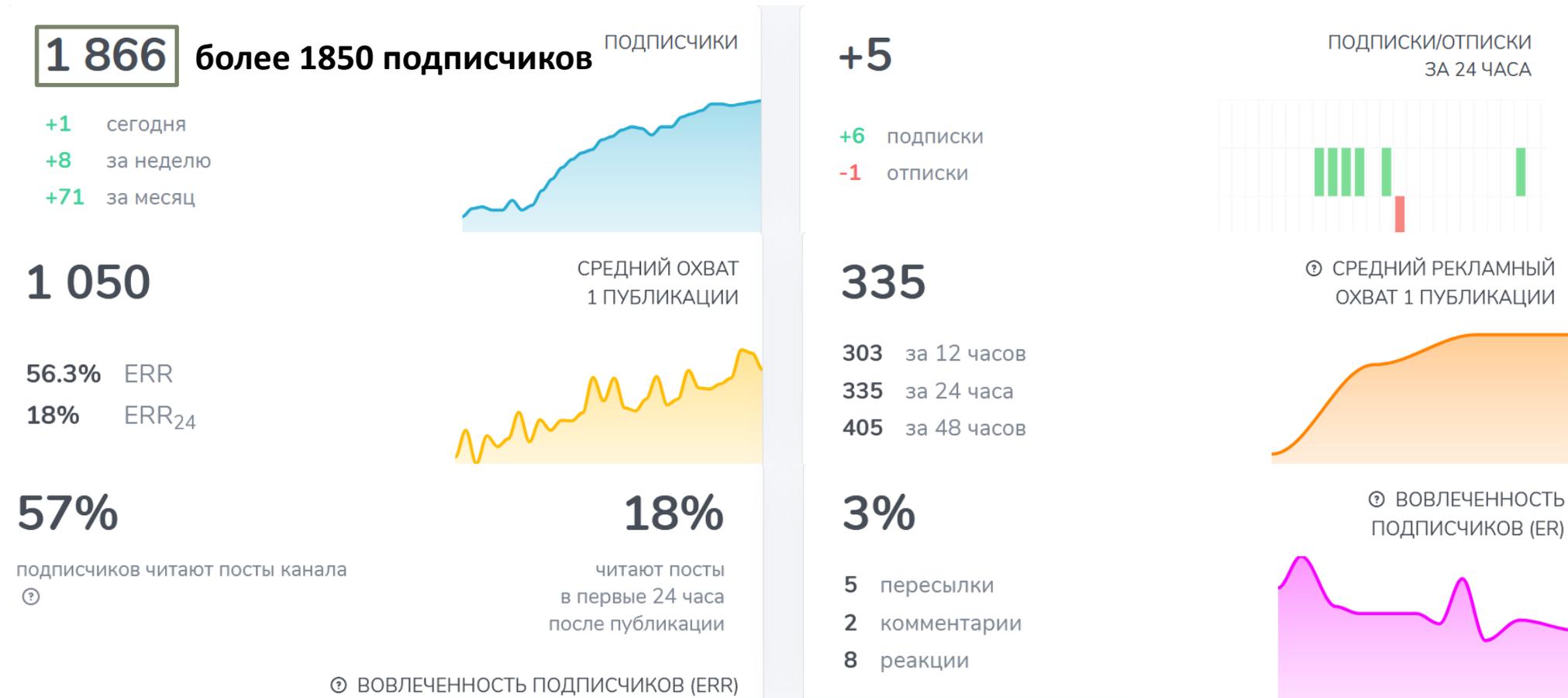
## 3. PascalABCNETLinux

Конверсия	1,86 %
Достижения цели	36
Целевые визиты	27

Данные  
Яндекс-метрики:  
**900** скачиваний в  
день

# Официальный телеграм-канал PascalABC.NET

В сентябре 2022 г. был открыт официальный Телеграм-канал PascalABC.NET [https://t.me/pascalabc\\_official](https://t.me/pascalabc_official). На скриншоте – статистика TGStat сегодня (+85% за 2 года):



# Телеграм-группа для обсуждения PascalABC.NET

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

<https://t.me/PABCofficial>

← PascalABC.NET 331 участник

Конференция "Использование системы программирования PascalABC.NET" 208 сообщений

# Общие вопросы, обсужде... ✓ Вс  
Вы: ➔ Успехи китайского ИИ ...

П Помогите решить задачу! Пн  
Лариса: Или представление чисе...

В Вопросы разработчикам ✓ Вс  
Вы: <https://github.com/pascalabcn...>

Е ЕГЭ и ОГЭ по информатике Вс  
Тренер: Пишите сразу в прокурат...

Д Для учителей и преподавате... Сб  
Сергей: Три года назад в пункт пр...

Б Болталка ✓ Ср  
Вы: Ну, олимпиад всяких и разны...

К Конференция "Ис... ✓ 02.03.2025  
Вы: 15 минут

Р PascalABC.NET для на... 01.03.2025  
ser: точно - работает

П PascalABC.NET под Lin... 03.02.2025  
Nikolay: ➔ [Runtu-screen0.webm](#)

М Мнимые вирусы в Ра... 28.01.2025  
Nikolay: Вроде как галочка появля...

Обсуждение Вики ... ✓ 17.12.2024  
Вы: Начал писать страничку <http...>

**Закреплённое сообщение**  
Фотография

Пятая конференции "Использование системы программирования **PascalABC.NET** в обучении программированию" пройдет 28-29 марта 2025 г. онлайн.

**Примерная программа конференции**

1. **PascalABC.NET 2025** (Михалкович С.С.)
2. Модуль LightPT 2025. Легковесная автоматическая проверка заданий. (Михалкович С.С.)
3. Короткий код - панацея или вред? (Демяненко Я.М.)
4. Модуль Coords и новая Черепаха (Михалкович С.С.)
5. Применение регулярных выражений в задачах КЕГЭ (Ипатов Ю.А.)
6. Решение задачи № 6 ЕГЭ "Исполнитель" на **PascalABC.NET** (Крючков М.В.)
7. Создание Телеграм-ботов на **PascalABC.NET** - новый взгляд (Пучкин М.В.)
8. Задачник РТ - объектно-ориентированное программирование на **PascalABC.NET** (Абрамян М.Э.)
9. **PascalABC.NET** и Python - взгляд из 2025
10. Новые литеральные массивы и новые встроенные множества. Словари и всё что можно с ними делать (Михалкович С.С.)
11. Типы - залог здоровья кода. (Демяненко Я.М.)
12. Решение задачи № 27 ЕГЭ "Кластеризация" на **PascalABC.NET** (Крючков М.В.)
13. Проект SPython (Мовчан Е.).
14. **PascalABC.NET** и машинное обучение
15. **PascalABC.NET** и ChatGPT (Михалкович С.С.)
16. Круглый стол (возможно)

**Мастер-класс.** Модуль LightPT - составление проверяемых задач к занятию.

Олимпиада для школьников **PascalABC.NET** (анонс)

10  
изменено 16:44 ✓

# Новая Wiki PascalABC.NET

Новая wiki  
расположена по  
адресу  
[pascalabc.miraheze.org](http://pascalabc.miraheze.org)  
Любой может  
редактировать её  
страницы

[Заглавная](#) [Обсуждение](#)

[Читать](#) [Править](#) [История](#) ★ [Инструменты](#)

## Добро пожаловать на вики-сайт языка PascalABC.NET!



Сайт для справки, советов и материалов для обучения программированию на языке PascalABC.NET.  
Здесь каждый может внести свой вклад.

### С чего начать?

- Установка PascalABC.NET ([Windows](#) / [Linux](#) / [macOS](#))
- [Первая программа: "Привет, Мир!"](#)

Для быстрого старта рекомендуются материалы:

- [Learn PascalABC.NET in Y minutes](#)
- [Более глубокий быстрый старт](#)
- Если вы хотите попробовать редактирование разметки  
делайте это на специально отведённой для этого странице  
- [Проба](#).

### Справка



На сайт скопирована большая часть содержимого справочной системы среды разработки.

- Можно посмотреть [описание языка PascalABC.NET](#).
- Все страницы из справки помечены шаблоном  
[Шаблон:Материалы из справки](#)
- Все страницы из справки находятся в отдельной категории  
[Категория:Страницы справки](#)

# Learn PascalABC.NET in Y minutes

На вики-странице [Learn PascalABC.NET in Y minutes](https://learnxinyminutes.com/) в формате сайта <https://learnxinyminutes.com/> впервые представлены все ключевые возможности PascalABC.NET, отражающие 17 лет его эволюции.

## Learn PascalABC.NET in Y minutes

Статья [Обсуждение](#)

[Читать](#) [Править](#) [История](#) ★ [Инструменты](#) ▾

Эта страница написана в формате сайта [Learn X in Y minutes](#) где X = PascalABC.NET.

**PascalABC.NET** — это элегантный и типобезопасный компилируемый язык программирования, основное предназначение которого — образовательная сфера и научные исследования. PascalABC.NET позволяет создавать надёжные приложения на платформе .NET и поддерживает мультипарадигменный подход, объединяя структурный, объектно-ориентированный и функциональный стили программирования.

## Комментарии [\[править\]](#)

```
// Однострочные комментарии начинаются с //
```

```
{  
Многострочные комментарии  
выглядят следующим образом  
}
```

```
(*  
Многострочные комментарии  
можно также делать так  
*)
```

## Лямбда-выражения [\[править\]](#)

**Лямбда-выражение** - это анонимная (безымянная) функция, которая определяется в месте своего использования. Лямбда-выражения можно присваивать функциональным переменным и передавать в другие функции как callback.

```
// Общая форма лямбда-выражения  
параметры -> значение // для лямбда-функций  
параметры -> действие // для лямбда-процедур  
  
// Инициализация функциональной переменной лямбда-выражением  
// Типы параметров лямбда-функции и тип возвращаемого значения выводятся из типа  
// функциональной переменной  
var f: real -> real := x -> x*x;  
var pred: (real,real) -> boolean := (x,y) -> x < y;  
var pr: integer -> () := n -> Print(n);  
var hello: procedure := procedure -> Print('Hello');
```

```
// Вызов через функциональную переменную  
f(5).Print;  
pred(3,2).Print;  
pr(77);  
hello;
```

```
// Передача лямбда-выражения как параметра функции  
var a := Arr(1..10);  
a.Select(x -> x * x).Println; // лямбда-преобразование  
a.Where(x -> x mod 2 = 0).Println; // лямбда-условие
```

## Простейшие программы [\[править\]](#)

## Последовательности [\[править\]](#)

# PascalABC.NET в проекте Rosetta Code

Проект [Rosetta Code](#) — это сайт, который представляет собой коллекцию решений различных задач на множестве языков программирования (более 1000). Основная цель проекта — помочь программистам, преподавателям и учащимся сравнивать и изучать, как одна и та же задача может быть решена на разных языках программирования.

**PascalABC.NET** занимает **63 место** с **603 решениями!**

Fortran  
> FreeBASIC  
FutureBasic  
Go  
J  
Java  
JavaScript  
jq  
Julia  
Lua  
Matlab  
Mathematica /Wolfram Language  
newLISP  
Nim  
OCaml  
Pascal  
**PascalABC.NET**  
Perl  
> Phix  
PHP  
Picat

## PascalABC.NET

```
##
uses School;

function AscendingSeq(n: integer): sequence of integer;
begin
  for var x := n*10 + n mod 10 + 1 to n*10 + 9 do
    yield sequence AscendingSeq(x) + x;
  end;

AscendingSeq(0).Order.Where(n -> n.IsPrime).Print;
```

### Output:

```
2 3 5 7 13 17 19 23 29 37 47 59 67 79 89 127 137 139 149 157 167 179 239 257 269 347 349 359 367 379 389 457 467 479
569 1237 1249 1259 1279 1289 1367 1459 1489 1567 1579 1789 2347 2357 2389 2459 2467 2579 2689 2789 3457 3467 3469
4567 4679 4789 5689 12347 12379 12457 12479 12569 12589 12689 13457 13469 13567 13679 13789 15679 23459 23567 23689
23789 25679 34589 34679 123457 123479 124567 124679 125789 134789 145679 234589 235679 235789 245789 345679 345689
1234789 1235789 1245689 1456789 12356789 23456789
```

## Perl

Library: ntheory

```
use strict;
use warnings;
use ntheory 'is_prime';
```

## Ascending primes

Page Discussion Read View

Generate and show all primes with strictly ascending decimal digits.

Aside: Try solving without peeking at existing solutions. I had a weird idea for generating a prime sieve faster, which needless to say didn't pan out. The solution may be p(r)etty trivial but generating them quickly is at least mildly interesting. Tip: filtering all 7,027,260 primes below 123,456,789 probably won't kill you, but there is at least one significantly better and much faster way, needing a mere 511 odd/prime tests.

# Что – в школах?

- **Переход на язык Питон.** Единый учебник в каждом классе (Поляков 7-9, Поляков 10-11, Босова 8-9).
- На **олимпиадах** по программированию PascalABC.NET вытесняется. Хотя и опережает Java и C# в 3 раза. В некоторых регионах – **25% (Марий-Эл)**. В Москве – 1%. Ситуация стабильна последние 2 года.
- Сообщество DL Club – олимпиады для учащихся 1-5 классов. PascalABC.NET – 85%, количество участников – около 300.
- Олимпиада Компьютерной школы мехмата ЮФУ по PascalABC.NET, использующая легковесную автоматическую проверку – около 100 школьников 7-10 классов.
- Борьба за свежие версии PascalABC.NET на ЕГЭ по информатике 😊 Разработчики рекомендовали в начале учебного года версию 3.10.3.

# Съезд учителей информатики (Сириус, 26-27 августа 2024 г.)

- Около 40% учителей используют Pascal
- Надо учитывать, что на съезд не приглашали учителей с докладами по PascalABC.NET и была большая секция олимпиадников (это C++)



- **Инновации PascalABC.NET** и его опора на одну из самых прогрессивных программных платформ современности .NET и язык C# - **игнорируются**. Для многих это по-прежнему старый Паскаль образца 70-х годов.
- Слово **PascalABC.NET** стараются не произносить, заменяя его словом **Паскаль**

**Что ж, давайте сравнивать ...**

# PascalABC.NET и Python – сравнение

Слайд конференции  
2023 г., измененный

## Python

- Появился в **1990 г.**, за последние 10 лет изменился, но **не в сторону обучения**
- **Лидер** рейтингов языков программирования
- **Отличные** библиотеки и документация
- **Лаконичный синтаксис** (отступы)
- **Интерпретируемый**, низкая скорость выполнения (**в 100 раз медленнее** PascalABC.NET)
- **Нет контроля** семантических ошибок до запуска программы – **плохо** для начинающих
- Динамическая типизация
- Среда и сообщения об ошибках **на английском**, PyCharm – **сложный** для новичков
- **Более сложный переход** на C#, Java, C++ в силу различия языковых конструкций
- Нет встроенных инструментов для обучения

## PascalABC.NET

- Возник в **2007 г.**, нацелен на обучение современному программированию
- Частично **основан на C#** - 5-й язык в рейтинге ЯП
- Простой и **компактный синтаксис** (блоки)
- Компилируемый, **высокая скорость** выполнения
- **Полный контроль ошибок** при компиляции – **хорошо** для начинающих
- Статическая типизация
- Простая и мощная **среда программирования** (Intellisense и автоформатирование)
- 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

**PascalABC.NET позаимствовал у  
Python лаконичность кода.  
Итог – основное преимущество  
Python – растворилось**

# Современный PascalABC.NET – лаконичный код

## Ввод - вывод

```
var a := ReadInteger;  
Println(a, a*a, a*a*a);  
var (x,y) := ReadReal2;  
Println('${x}+{y}={x+y}');
```

## Циклы

```
for var i:=0 to 9 step 2 do  
    Print(i);  
foreach var x in [2,3,5] do  
    Print(x);  
loop 10 do  
    Print('*');
```

## Кортежи

```
var (a,b) := (3,5);  
(a,b) := (b,a);
```

## Длинные целые

```
var x := 2bi ** 1000;  
var p: BigInteger := 1;  
for var i:=2 to 100 do  
    p *= i;
```

## Диапазоны

```
if x in 3..7 then Print(x);  
foreach var x in 3..7 do  
    Print(x);  
(3..7).Select(x → x>4 ? 1 : x).Print;
```

## Массивы

```
var a = [12,3,5,2,41,92,28,35];  
var a1 := Arr(1..9) * 2 + [33,55,77];  
var a2 := ReadArrInteger(10);  
var a3 := ArrRandomReal(10,2,5);  
var a4 := ArrGen(10,i → 2*i+1);  
a.Println;  
a.Where(x → x.IsEven).Println;
```

## Срезы

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

## Матрицы

```
var c: array [,] of integer;  
c := MatrGen(3,3,(i,j)→(i+1)*(j+1));  
c.Println; c.Row(2).Max.Print;
```

## Функции

```
function Add(a,b: integer) := a+b;  
procedure p1 := Print(1);
```

## Списки, множества, словари

```
var L: List<integer> := [];  
L.Add(2); L.AddRange([5,7,9]);  
L.Insert(0,66); L.RemoveAt(1);  
var s: set of integer := SetOf(1,2,3);  
var s1 := SetOf(3,4,5);  
Print(s + s1, s * s1, s - s1,3 in s);  
var d: Dictionary<string,integer>;  
d := Dict('cat' to 2, 'dog' to 3);  
if 'cat' in d then d['cat'] += 1;  
var dd := 'Знание - сила'.EachCount;  
dd.Println;
```

## Комбинаторика

```
var b := ['Петя','Коля','Вася'];  
var g := ['Марина','Таня'];  
b.Cartesian(g).Println;  
'пак'.CartesianPower(4).Println;  
'строка'.Permutations(4).Println;  
'тропа'.Combinations(3).Println;
```

# 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)
```

28 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 75 раз  
быстрее!

Задача. Вычислить  $\sum_{i=1}^n \sum_{j=1}^n \frac{1}{i \cdot j}$

# PascalABC.NET и Python – диагностика ошибок

Ой, а нет русской версии?

```

1 a = []
2 for i in range(10):
3     x = int(input())
4     a.append(x)
5     Print(sum(a))

```

Run: my x

```

8
9
88
77

```

Traceback (most recent call last):

```

File "D:\PyProjects\pythonProject\Lib\my.py",
  Print(sum(a))
  ^^^^^
NameError: name 'Print' is not defined. Did you

```

Как же так получилось, что программа доработала почти до конца и упала? А в PascalABC.NET такого не было...

Ой, а ошибки тоже на иностранном?

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

```

##
var a: List<real> := [];
loop 10 do
  a.Add(ReadReal);
  Print(a.Sum)

```

PascalABC.NET не даст запустить неверную программу пока ученик не исправит все ошибки!

Список ошибок				
	Строка	Описание	Файл	Путь
✖	1	5	Неизвестное имя 'Print'	Program1.pas C:\PABCWork.NET

Окно вывода | Список ошибок | Сообщения компилятора

1 ошибок | Строка 5 Столбец 1

Ошибка компиляции (**рано!**)

# Что нового в PascalABC.NET в 2023-25 гг.

# Новое в языке

# Новые литеральные массивы

## Версия 3.10.3

### Важнейшее изменение.

Выражения вида [1,2,3] в PascalABC.NET трактуется по умолчанию как массив.

Данное поведение реализовано в последних версиях Delphi.

Сочетание использования массивов и множеств представлено в коде.

Основное – пользователь не испытывает сложностей при переходе от старого кода.

```
begin
  var a := [1,2,3]; // по умолчанию - массив
  PrintLn(a,TypeName(a));
  var s: set of byte := [3,4,5]; // множество
  var s1 := SetOf(6,7,8); // множество
  PrintLn(s,TypeName(s));
  // Операнды смешанных типов. Результат - множество
  PrintLn(s + [1,2,3], s * [1,2,3], [1,2,3] - s);
  PrintLn(a + [7,8,9], a * 3); // Это операции с массивами
  // in - операция для массива
  Print(3 in [3,5,7]); // True
  // Сравнение множества и массива - как множества
  Print(SetOf(3,4,5) = [4,3,5]); // True
  // Сравнение массива и массива. Сравняются ссылки
  Print([1,2,3] = [1,2,3]); // False
end.
```

```
[1,2,3] array of integer
{3,4,5} set of byte
{1,2,3,4,5} {3} {1,2}
[1,2,3,7,8,9] [1,2,3,1,2,3,1,2,3]
True True False
```

# Новая реализация стандартных множеств

Версия 3.10.3

## Важнейшее изменение.

Внутренняя реализация стандартных множеств set of T полностью переделана – они **стали работать быстрее** в 10-20 раз.

По умолчанию синтаксис [1,2,3] теперь используется для массивов, поэтому для создания стандартных множеств есть 3 пути:

1. Явно описать тип – set of integer
2. Использовать функцию SetOf(1,3)
3. Использовать преобразование .ToSet

```
begin
  var s: set of byte := [3..7]; // множество
  var s1 := SetOf(6.1,7.2,8.3); // множество
  var s2 := 1234.ToString.ToSet; // множество СИМВОЛОВ
  Println(TypeName(s));
  Println(TypeName(s1));
  Println(TypeName(s2));
end.
```

```
set of byte
set of real
set of char
```

# Инициализатор пустой коллекции []

## Версия 3.10.3

Инициализатор [] пустой коллекции позволяет присваивать его любой коллекции, заменяя вызов конструктора по умолчанию.

```
function Divisors(n: integer): List<integer>;
begin
  Result := []; // Преобразуется в List<integer>
  for var i:=2 to n do
    if n.Divs(i) then
      Result.Add(i);
  end;
begin
  var old: set of integer := [];
  var s: sequence of integer := [];
  var a: array of integer := [];
  var lst: List<(real,real)> := [];
  var d: Dictionary<string,integer> := [];
  var st: Stack<string> := [];
  var q: Queue<string> := [];
  var hs: HashSet<integer> := [];
  var ss: SortedSet<integer> := [];

  var aa: array of array of integer := [[1],[],[2,3]];
  Divisors(2025).Println;
end.
```

# Новый тип faststring

## Версия 3.10.3

Тип faststring реализован на основе StringBuilder, включает неявное преобразование в string и обратно.

**Основное** – операция изменения символа `fs[i] := 'g'` работает крайне быстро.

Также для `s: faststring` реализованы методы:

`s.IndexOf(s1)` – индекс подстроки

`s.Replace(s1, s2: string; n: integer)` – возвращает faststring с заменой `n` вхождений `s1` на `s2`.

```
// 6.0 sec PascalABC.NET vs 11.4 sec Python
begin
  var mx := 0;
  for var n := 4 to 9999 do
    begin
      var s: faststring := '4' + '1' * n;

      while ('411' in s) or ('1111' in s) do
        s.Replace('411', '14', 1)
          .Replace('1111', '1', 1);

      mx := max(mx, s.ToString.Sum(d -> d.todigit));
    end;
  Print(mx, Milliseconds / 1000);
end.
```

# Разделители \_

## Версия 3.10.0

В целых или вещественных можно использовать разделители \_

Это особенно удобно для длинных чисел

```
begin
  var n := 10_000_100;
  Println(n);
  var m := 1_234.567_8;
  Println(m);
  var b := 1_000_000_000_000bi; // BigInteger
  Println(b);
end.
```

# Многострочные строковые литералы '''

## Версия 3.10.0

Появились многострочные строковые литералы, которые чуть ранее появились в Delphi, C#, Java

Строки с открывающими и закрывающими кавычками ''' не должны содержать более ничего.

Во всех строках игнорируется отступ, равный отступу последней строки.

Метод `s.ToLines` преобразует многострочный строковый литерал в массив строк.

```
begin
```

```
  var s := '''
```

Многострочная строка

Открывающие и закрывающие кавычки

должны быть каждая на своей строке

Лидирующие пробелы слева от закрывающих кавычек

удаляются во всех строках многострочной строки

Внутри могут быть апострофы ' и ''

но не три апострофа подряд

```
''';
```

```
Write(s, '<<<');
```

```
s.ToLines.PrintLines(line -> line + '!')
```

```
end.
```

# Новое в библиотеках

# Внешние Zip и Cartesian

## Версия 3.10.0

В стандартной библиотеке реализованы внешние функции Zip и Cartesian с количеством последовательностей от 2 до 5 и необязательной функцией проекции

```
begin
  Zip(1..5,2..6).Println;
  Zip(1..5,2..6,(x,y) -> x+y).Println;
  Zip(1..5,2..6,3..7).Println;
  Zip(1..5,2..6,3..7,(x,y,z) -> x+y+z).Println;
  Cartesian(1..4,3..5).Println;
  Cartesian(1..4,3..5,(x,y) -> x*y).Println;
  Cartesian(1..2,4..5,6..7).Println;
  Cartesian(1..2,4..5,6..7,(x,y,z) -> x*y*z).Println;
  foreach var (a,b) in Cartesian(1..2,3..5) do
    Print('${a}+{b}')
```

end.

```
(1,2) (2,3) (3,4) (4,5) (5,6)
3 5 7 9 11
(1,2,3) (2,3,4) (3,4,5) (4,5,6) (5,6,7)
6 9 12 15 18
(1,3) (1,4) (1,5) (2,3) (2,4) (2,5) (3,3) (3,4) (3,5) (4,3) (4,4)
(4,5)
3 4 5 6 8 10 9 12 15 12 16 20
(1,4,6) (1,4,7) (1,5,6) (1,5,7) (2,4,6) (2,4,7) (2,5,6) (2,5,7)
24 28 30 35 48 56 60 70
1+3 1+4 1+5 2+3 2+4 2+5
```

# Улучшился TypeName

## Версия 3.10.0

TypeName выводит правильный паскалевский тип в ряде непростых ситуаций

```
begin
  var q := Arr(1..10);
  Println(TypeName(q.Select(x -> x + 0.5)));
  Println(TypeName(q.Where(x -> x.IsEven)));
  Println(TypeName(q.Pairwise));
  Println(
    TypeName(q.Select(x -> x + 0.5).Numerate));
  Println(TypeName(q.Cartesian(q)));
  Println(TypeName(q.Cartesian(q).ToList));
  Println(TypeName(q.EachCount));
end.
```

```
sequence of real
sequence of integer
sequence of (integer, integer)
sequence of (integer, real)
sequence of (integer, integer)
List<(integer, integer)>
Dictionary<integer, integer>
```

# Преобразование в вещественное с разделителем

## Версия 3.10.0

Реализован параметр для преобразования строк в вещественные с другим разделителем целой и дробной части

```
begin
  '2,7 3,2 4,3 5,8 3,3'.ToReals(',').Println;
  var s := '2,4';
  s.ToReal(',').Print;
  StrToReal(s, ',').Print;
end.
```

```
2.7 3.2 4.3 5.8 3.3
2.4 2.4
```

# Новые функции, методы и операции для словарей

## Версия 3.9

Операция **to** создает пару `KeyValuePair`. То же делает функция `Pair`.

Словарь может инициализироваться функцией `Dict` с набором пар, массивами [ключи], [значения].

Метод `Update` обновляет значения в одном словаре по другому словарю.

Можно вычитать из словаря ключ или последовательность ключей.

```
begin
  // Новая операция 1 to 2 - конструктор пары
  var d := Dict('cat' to 'кошка', 'dog' to 'собака');
  // Новая функция Pair - конструктор пары
  var d0 := Dict(Pair('cat', 'кот'), Pair('camel', 'верблюд'));
  var dd := Dict('Иванов' to [5, 3, 3], 'Петров' to [4, 4]);
  var d1 := Dict(d); // Копия словаря
  var d2 := Dict([11, 22], [333, 444]); // [ключи], [значения]
  d.Update(d0); // Обновление значений и добавление новых
  d += d0; // То же
  // Слияние словарей
  var d3 := d + Dict('world' to 'мир', 'cloud' to 'облако');
  Println(d3);
  d3 -= Seq('cat', 'camel'); // Вычитание пар с данными ключами
  Println(d3);
  d3 := d3 - 'world';
  Println(d3);
end.
```

```
{(cat, кот), (dog, собака), (camel, верблюд), (world, мир), (cloud, облако)}
{(dog, собака), (world, мир), (cloud, облако)}
{(dog, собака), (cloud, облако)}
```

# BinFormat в модуле School

## Версия 3.9.0

Добавлена функция BinFormat, позволяющая переводить целое десятичное число в двоичную систему счисления и возвращать отформатированный результат.

Вызов функции:

BinFormat(ДесятичноеЧисло, Длина\_в\_байтах, Разделитель\_байтов).  
По умолчанию нужное количество байт выбирается автоматически.

```
uses School;  
begin  
  PrintLn(BinFormat(13));  
  PrintLn(BinFormat(12345678));  
  PrintLn(BinFormat(12345678, 4));  
  PrintLn(BinFormat(12345678, 2, '.'));  
  PrintLn(BinFormat(0, 2));  
  PrintLn(BinFormat(12453674856bi));  
end.
```

```
00001101  
10111100 01100001 01001110  
00000000 10111100 01100001 01001110  
10111100.01100001.01001110  
00000000 00000000  
00000010 11100110 01001011 11111111 01101000
```

# CalcIP в модуле School

## Версия 3.9.0

В модуль School добавлен класс CalcIP, позволяющий решать простейшие задачи с IP-адресами.

(№ 6839) В терминологии сетей TCP/IP маской сети называют двоичное число, которое показывает, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу узла в этой сети. Адрес сети получается в результате применения поразрядной конъюнкции к заданному адресу узла и маске сети. Сеть задана IP-адресом 192.168.32.160 и маской сети 255.255.255.240. Сколько в этой сети IP-адресов, для которых сумма единиц в двоичной записи IP-адреса чётна? В ответе укажите только число.

```
## uses School;  
var net := new  
    CalcIP('192.168.32.160', '255.255.255.240');  
net  
    .GenAddrBin  
    .Where(t -> t.CountOf('1') mod 2 = 0)  
    .Count  
    .Print;
```

8

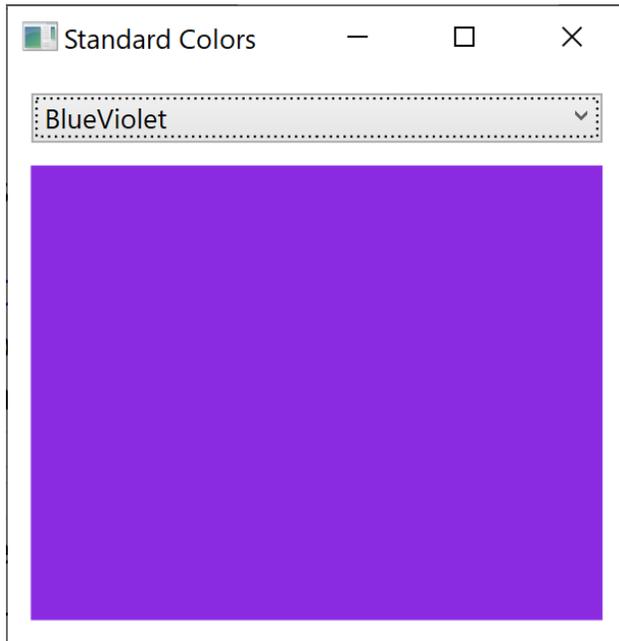
# Новые модули и библиотеки

# Модуль WPF

## Версия 3.10.0

Новый модуль WPF позволяет создавать оконные приложения без оберток для каждой компоненты

В нем имеется ряд методов, позволяющих сократить код создания интерфейса



```
uses WPF;

begin
  var colorNames := typeof(Colors).GetProperties
    .ToDictionary(pi -> pi.Name,
                 pi -> GBrush(Color(pi.GetValue(nil,nil))));
  MainWindow.Title := 'Standard Colors';
  MainWindow.WindowStartupLocation
    := WindowStartupLocation.CenterScreen;
  var mainpanel := CreateStackPanel(Margin := 10,
    Horizontal := True).AsMainContent;
  var panel1 := CreateStackPanel(Width := 250).AddTo(mainpanel);
  var cb := CreateComboBox.AddTo(panel1);
  cb.ItemsSource := colorNames.Keys;
  var rect := CreateRectangle(Height := 200,
    Margin := [0,10]).AddTo(panel1);
  rect.Fill := Brushes.Blue;

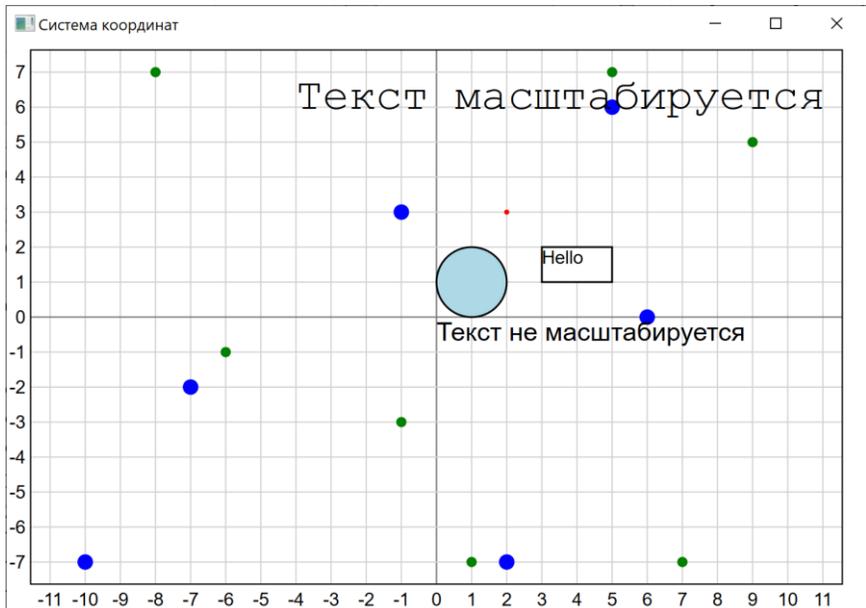
  cb.SelectionChanged += procedure(o,e) ->
    (rect.Fill := colorNames[cb.SelectedValue as string]);

  MainWindow.SizeToContent := SizeToContent.WidthAndHeight;
end.
```

# Модуль Coords

## Версия 3.10.3

Новый модуль Coords позволяет визуализировать точки, круги, прямоугольники, линии, текст с отображением системы координат и возможностью масштабирования колёсиком мыши и перемещения центра левой мышью.



```
uses Coords;
```

```
function RandomPoint: Point
```

```
:= Pnt(Random(-13,13),Random(-10,10));
```

```
begin
```

```
DrawPoints(ArrGen(10,i -> RandomPoint),PointRadius := 4);
```

```
DrawPoints(ArrGen(10,i -> RandomPoint),PointRadius := 6);
```

```
DrawPoint(2,3,Colors.Red);
```

```
DrawCircle(1,1,1,Colors.LightBlue);
```

```
DrawRectangle(3,2,2,1);
```

```
DrawText(3,2,'Hello');
```

```
DrawTextUnscaled(0,0,'Текст не масштабируется',  
Size := 20, Color := Colors.Red);
```

```
DrawText(-4,7,'Текст масштабируется',
```

```
FontName := 'Courier New', Size := 34);
```

```
end.
```

# Библиотека MathNet.Numerics.dll

## Версия 3.10.0

В PascalABC.NET интегрирована известная свободная библиотека математических вычислений MathNet.Numerics.dll

Справа представлена программа, решающая систему линейных алгебраических уравнений.

```
{ $reference Mathnet.Numerics.dll }
uses Mathnet.Numerics.LinearAlgebra;

begin
  var A := Matrix<double>.Build.DenseOfArray(Matrc(
    [3.0, 2, -1],
    [2.0, -2, 4],
    [-1, 0.5, -1]));
  var b := Vector<double>.Build.Dense([1.0, -2, 0]);
  var x := A.Solve(b);
  Print(x.ToArray);
end.
```

```
[1, -2, -2]
```

# Оптимизация

# Ускорение кортежного присваивания

## Версия 3.10.0

Кортежное присваивание разбивается также на множество индивидуальных присваиваний.

Вычисляется оптимальное количество и последовательность индивидуальных присваиваний.

Для двух переменных кортежное присваивание сравнялось по скорости с вызовом Swap.

```
uses Utils;

procedure BubbleSortTuple(a: array of integer);
begin
  for var i := 1 to a.Length-1 do
    for var j := a.Length-1 downto i do
      if a[j] < a[j-1] then
        (a[j], a[j-1]) := (a[j-1], a[j])
      end;
    end;
  end;

procedure BubbleSortSwap(a: array of integer);
begin
  for var i := 1 to a.Length-1 do
    for var j := a.Length-1 downto i do
      if a[j] < a[j-1] then
        Swap(a[j], a[j-1]);
      end;
    end;
  end;

begin
  var n := 50000;
  var a := ArrRandomInteger(n, 0, integer.MaxValue-1);
  var b := Copy(a);
  Benchmark(procedure -> BubbleSortSwap(b), 2).Println;
  b := Copy(a);
  Benchmark(procedure -> BubbleSortTuple(b), 2).Println;
end.
```

2556

2448

# Выводы

# Выводы

- 1. PascalABC.NET 2025 – современный язык программирования общего назначения, разработанный в России специально для обучения программированию.** PascalABC.NET сфокусирован на образовательных целях, что делает его более подходящим для обучения основам программирования и алгоритмического мышления.
- 2. PascalABC.NET идеально адаптирован для школьного образования.** Он поддерживает как базовый, так и углубленный уровни обучения информатике. Кроме того, PascalABC.NET за счет наличия богатой стандартной библиотеки позволяет просто решать задачи олимпиадного программирования задачи ЕГЭ, что подчеркивает его значимость в образовательной среде.
- 3. PascalABC.NET предоставляет простую и удобную графическую среду разработки,** которая работает на всех популярных платформах: Windows, Linux и MacOS.
- 4. PascalABC.NET основан на мощной платформе .NET и языке C#, который входит в топ-5 самых популярных языков программирования.** Это обеспечивает высокую производительность и современные возможности. Строгая типизация PascalABC.NET помогает избежать многих ошибок, характерных для динамически типизированного Python, что особенно важно для начинающих.
- 5. PascalABC.NET – это многогранный и активно развивающийся язык с огромным потенциалом.** PascalABC.NET продолжает развиваться, предлагая новые функции и возможности, специально ориентированные на образовательные задачи. Это делает его перспективным выбором для обучения современному программированию.

# Важные ссылки

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

[Курсы на Stepik](#)

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

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

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

[PascalABC.NET 2023 – российский Паскаль](#)

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

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

[Модуль невидимой автоматической проверки заданий \(доклад 2023 г.\)](#)

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

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

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

[PascalABC.NET под Linux – что нас ждёт \(доклад 2023 г.\)](#)



Михалкович С.С.

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

[miks@sfedu.ru](mailto:miks@sfedu.ru)





# **PascalABC.NET – базовый уровень. Слайды 2023 г.**

# Оператор вывода Print

Слайд конференции  
2023 г.

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

# Цикл loop

Слайд конференции  
2023 г.

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.
```

Пример 2.

```
uses Robot;

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

# Операции += и \*=

Слайд конференции  
2023 г.

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

Слайд конференции  
2023 г.

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.
```

Пример 2.

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

# 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.
```

Пример 2.

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

Слайд конференции  
2023 г.

# Диапазоны и операция in

Слайд конференции  
2023 г.

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.
```

# 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;
    Print('Сумма чисел в диапазоне 10..20 =', sum);
  end.
```

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

Слайд конференции  
2023 г.

# 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.
```