

Основы программной инженерии (ПОИТ)
Технологии разработки программного обеспечения (ИСиТ)

Современные системы программирования

План лекции:

- система программирования, язык программирования;
- система программирования Windows;
- система программирования Linux (UNIX-подобная);
- стандарт POSIX;
- стандарты языков программирования;
- парадигмы программирования.

1. На прошлых лекциях:

Система программирования:

комплекс программных средств, предназначенных для автоматизации процесса разработки, отладки ПО и подготовки программного кода к выполнению



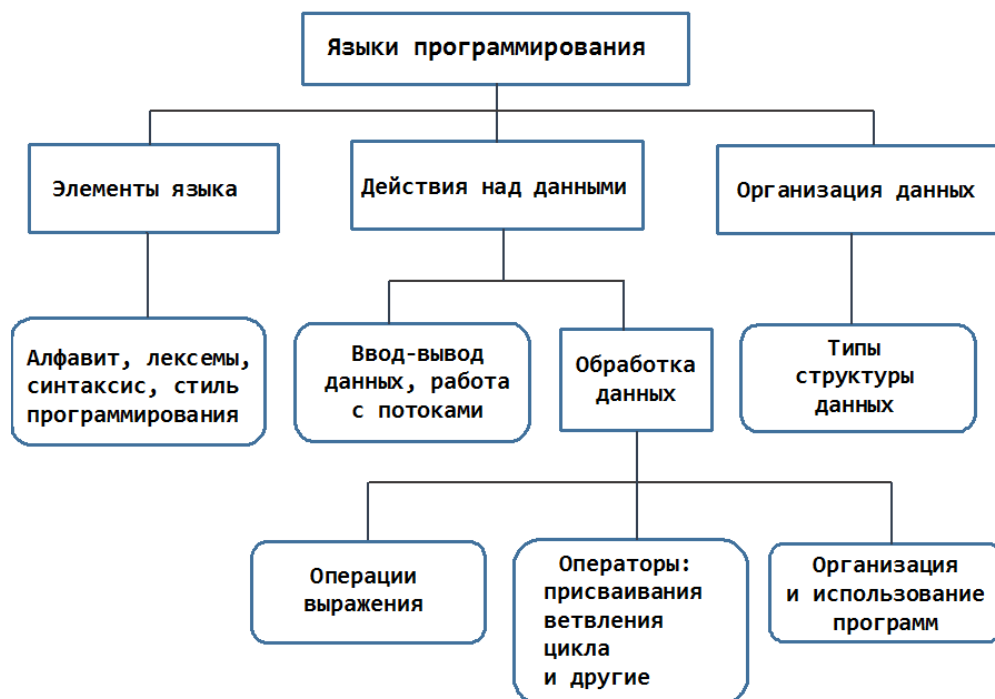
Интегрированная среда разработки:

набор инструментов для разработки и отладки программ, имеющий общую интерактивную графическую оболочку, поддерживающую выполнение всех основных функций жизненного цикла разработки программы.

Примеры IDE (визуальные среды):

Eclipse, Microsoft Visual Studio, NetBeans, Qt Creator, ...

Структура языка программирования:



Программа – алгоритм, записанный на языке программирования.

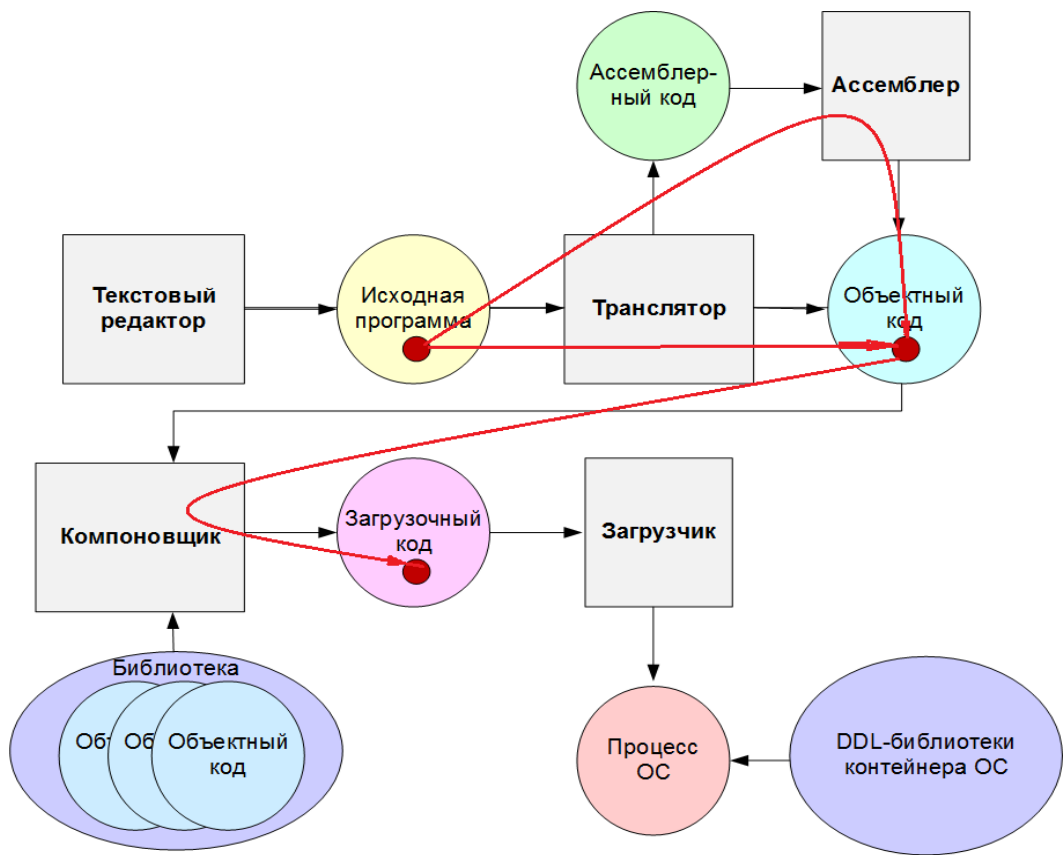
Текст программы (исходный код) – полное законченное и детальное описание алгоритма на языке программирования.

Объектный код: – результат работы транслятора. Один файл объектного кода – объектный модуль.

Объектный модуль – двоичный файл, который может быть объединён с другими объектными файлами при помощи редактора связей (компоновщика) для получения готового исполняемого модуля, либо библиотеки.

Загрузочный код – результат работы компоновщика.
Один файл загрузочного кода – загрузочный модуль.

От исходного кода к исполняемому модулю:



<p>Компилятор (транслятор) – программа, преобразующая исходный код на одном языке программирования в исходный код на другом языке; результат – объектный модуль</p>	<p>Компоновщик (linker, редактор связей) – программа, принимающая один или несколько объектных модулей и формирующая на их основе загрузочный модуль</p>	<p>Загрузчик (loader) – программа, предназначенная для запуска процесса операционной системы на основе загрузочного модуля</p>

2. Система программирования

Система программирования – инструментальное ПО, предназначенное для разработки программного продукта на этапах программирования и отладки. Каждая система программирования должна иметь некоторый встроенный в нее язык программирования, предназначенный для общения разработчика с используемыми инструментами.



Система программирования является основным инструментом программиста.

3. Классическая система программирования

С развитием сервисных средств ОС появились командные процессоры, что позволило объединять последовательность вызовов системных программ в единые командные файлы. Это упростило работу по запуску компонентов систем.

Специализированные командные процессоры (*координатор make* или командный *интерпретатор shell*) представляли собой интерпретаторы, на вход которым подавались файлы, записанные на особом командном языке.

Ядром системы программирования является язык программирования.

4. Интегрированная среда разработки




Интегрированная среда разработки (integrated development environment – IDE): – набор инструментов для разработки и отладки программ, имеющий общую интерактивную графическую оболочку, поддерживающую выполнение всех основных функций жизненного цикла разработки программы.

Примеры IDE: Visual Studio, NetBeans, Eclipse, Embarcadero Delphi и пр.


JDeveloper — бесплатная интегрированная среда разработки ПО. Для разработки на языках программирования Java, JavaScript, BPEL, PHP, SQL, PL/SQL; на языках разметки HTML, XML.

	1998
разработчик	Oracle
написана на	Java
ОС	кроссплатформенная
платформа	Java Virtual Machine
последняя версия	12c (12.2.1.2.0) (19.10.2016)
сайт	http://www.oracle.com/technetwork/developer-tools/jdev/


NetBeans — свободная интегрированная среда разработки ПО на языках программирования Java, Python, PHP, JavaScript, C, C++, Ада и др.

	1997
разработчик	Apache Software Foundation, Oracle и Sun Microsystems
написана на	Java
ОС	Microsoft Windows, Linux, macOS и Solaris
платформа	Java Virtual Machine
последняя версия	12.5 (13 сентября 2021)
сайт	netbeans.apache.org


Eclipse — свободная интегрированная среда разработки модульных кроссплатформенных приложений.

		7 ноября 2001
разработчик	Eclipse Foundation	
написана на	Java	
ОС	GNU/Linux, macOS, Microsoft Windows, Solaris	
платформа	Java Virtual Machine	
последняя версия	4.21.0 (15 сентября 2021)	
сайт	eclipse.org	


Oracle Sun Studio — интегрированная среда разработки для языков программирования Си, C++ и Фортран

		
разработчик	Oracle Corporation	
написана на	SPARC, x86, x86-64	
ОС	Solaris, OpenSolaris, Linux	
последняя версия	Oracle Developer Studio 12.6 (5 июля 2017 года)	
сайт	http://www.oracle.com/technetwork/server-storage/solarisstudio/overview	

Embarcadero Delphi — интегрированная среда разработки ПО для Microsoft Windows, Mac OS, iOS и Android на языке Delphi (раньше Object Pascal). Embarcadero Delphi является частью пакета Embarcadero RAD Studio

		1995
разработчик	Embarcadero Technologies	
написана на	Delphi и Object Pascal	
ОС	Microsoft Windows	
последняя версия	10.4.2 Sydney (24.02.2021)	
сайт	embarcadero.com/ru/product	

Microsoft Visual Studio — линейка продуктов компании Microsoft, включающих интегрированную среду разработки ПО и другие инструменты. Для разработки консольных приложений, игр, приложений с графическим интерфейсом, веб-сайтов, веб-приложений, веб-служб как в нативном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Silverlight.

	1997
разработчик	Microsoft
написана на	C++ и C#
ОС	Microsoft Windows, macOS
последняя версия	16.11.3 (14 сентября 2021)
сайт	visualstudio.microsoft.com

Visual Studio 2019

Быстрое написание кода. Автоматизация работы. Интегрированная среда разработки будущего.


Полный набор инструментов для всех этапов разработки, от начального замысла до финального развертывания


- ✓ Оптимизация работы IntelliSense в файлах C++
- ✓ Локальная разработка с поддержкой множества популярных эмуляторов
- ✓ Упрощенный доступ к тестам в обозревателе решений


Меньше ошибок при написании кода


Используйте рекомендации IntelliSense для быстрого и точного ввода нужных переменных при возникновении затруднений. Сохраняйте высокий темп работы вне зависимости от сложности за счет быстрого перехода к любому файлу, типу, элементу или символу. Используйте значки лампочек, которые рекомендуют действия по улучшению кода, например предлагают переименовать функцию или добавить параметр.


Все функции разработки >


 Разработка


 Анализ

 Отладка

 Тестирование

 Управление версиями

 Совместная работа

 Развертывание

ОС и поддерживаемые платформы

Windows	ARM, IA-64, Itanium, MIPS, DEC Alpha, PowerPC и x86	Microsoft
Linux	DEC Alpha, x86, x86_64, ARM, PowerPC, RISC-V и MIPS	Свободное программное обеспечение
macOS	Motorola 68k, PowerPC, IA-32, x86-64, ARM	Apple Inc.
OS/2	x86	IBM, Microsoft
Unix	Intel x86	
Solaris	SPARC, x86, x86-64, PowerPC	Sun Microsystems

5. Стандарты языков программирования

Стандарт языка программирования: Visual C++ 2017 версия 15.3 – это реализация стандарта C++17 или ISO/IEC 14882:2017.

Standards

ICS > 35 > 35.060

ISO/IEC 14882:2017

Programming languages — C++

THIS STANDARD HAS BEEN REVISED BY ISO/IEC 14882:2020

ABSTRACT

ISO/IEC 14882:2017 specifies requirements for implementations of the C++ programming language. The first such requirement is that they implement the language, so this document also defines C++. Other requirements and relaxations of the first requirement appear at various places within this document.

C++ is a general purpose programming language based on the C programming language as described in ISO/IEC 9899:2011 Programming languages ? C (hereinafter referred to as the C standard). In addition to the facilities provided by C, C++ provides additional data types, classes, templates, exceptions, namespaces, operator overloading, function name overloading, references, free store management operators, and additional library facilities.

GENERAL INFORMATION

Status :  Withdrawn	Publication date : 2017-12
Edition : 5	Number of pages : 1605
Technical Committee : ISO/IEC JTC 1/SC 22 Programming languages, their environments and system software interfaces	
ICS : 35.060 Languages used in information technology	

LIFE CYCLE


PREVIOUSLY	NOW	REVISED BY
 WITHDRAWN ISO/IEC 14882:2014	 WITHDRAWN ISO/IEC 14882:2017 Stage: 95.99 ~	 PUBLISHED ISO/IEC 14882:2020



Международная организация по стандартизации (ИСО) одобрила C++ 20, последнюю версию объектно-ориентированного языка программирования. Официальный стандарт опубликован в конце 2020 года.

ISO/IEC 14882:2020
Programming languages — C++

Общая информация

Текущий статус :  Published

Дата публикации : 2020-12

Версия : 6

Технический комитет : **ISO/IEC JTC 1/SC 22** Programming languages, their environments and system software interfaces

ICS : **35.060** Languages used in information technology

Жизненный цикл


Ранее

Withdrawn
ISO/IEC 14882:2017



Сейчас

Published
ISO/IEC 14882:2020

Стандарт, который пересматривается каждые 5 лет
Этап: **90.92** (Будет пересмотрено) 



Будет заменено

Under development
ISO/IEC CD 14882

Новая версия C++ выходит примерно каждые три года, ей присваивают номер года. Стандарт языка C++ 20 является преемником C++ 17.

https://www.ecma-international.org/publications/standards/Ecma-262.htm

ecma INTERNATIONAL **Standards**

What is Ecma | Activities | News | Standards

Standards Index
Standards List
Withdrawn Standards
Tech. Reports Index
Tech. Reports List
Withdrawn Tech. Reports
Mementos

Printer Friendly Version
Back

Standard ECMA-262

ECMAScript® 2019 Language Specification

10th edition (June 2019)

This Standard defines the ECMAScript 2019 general-purpose programming language.

The following files can be freely downloaded:

File name	Size (Bytes)	Content
ECMA-262 edition 10		Browsable HTML
ECMA-262.pdf	14 423 437	Acrobat (r) PDF file

ECMAScript — это встраиваемый расширяемый язык программирования, основа для построения других скриптовых языков. Стандартизирован международной организацией Ecma в спецификации ECMA-262. Расширениями языка являются JavaScript (Netscape), JScript (Microsoft) и ActionScript.

ECMAScript — это официальный стандарт языка JavaScript (Слово JavaScript не могло быть использовано, потому что слово Java является торговой маркой компании Sun)

Это восьмое издание спецификации языка ECMAScript.



[Java SE](#) > Java SE Specifications

Java Language and Virtual Machine Specifications

Java SE 17

Released September 2021 as [JSR 392](#)



The Java Language Specification, Java SE 17 Edition

- [HTML](#) | [PDF](#)
- Preview feature: [Pattern Matching for switch](#)




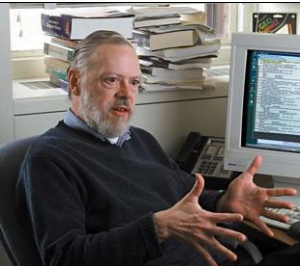
The Java Virtual Machine Specification, Java SE 17 Edition

- [HTML](#) | [PDF](#)

6. Системы программирования Windows и UNIX

Unix («UNIX» – зарегистрированная торговая марка The Open Group) – семейство переносимых, многозадачных и многопользовательских операционных систем, которые основаны на идеях оригинального проекта AT&T Unix, разработанного в 1970-х годах в исследовательском центре Bell Labs Кеном Томпсоном, Деннисом Ритчи и другими.

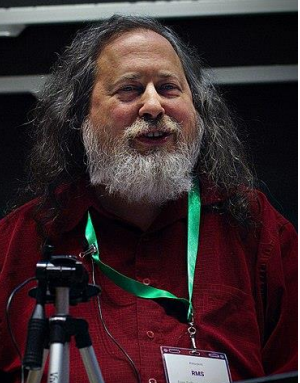
Дизайнеры языков программирования:

	Кеннет Лейн Томпсон – пионер компьютерной науки, известен своим вкладом в создание языка программирования C и операционной системы UNIX. Написал язык программирования B, предшественник языка C, участвует в создании языка программирования Go
	Деннис Ритчи – создатель языка программирования Си. Вместе с Кеном Томпсоном разработал Си для создания операционной системы UNIX. «У Ньютона есть фраза о стоящих на плечах гигантов. Мы все стоим на плечах Денниса», – Брайан Керниган.

Операционные системы семейства Unix характеризуются модульным дизайном, в котором каждая задача выполняется отдельной утилитой, взаимодействие осуществляется через единую файловую систему, а для работы с утилитами используется командная оболочка.

Unix является мультиплатформенной системой. Ядро системы разработано таким образом, что его легко можно приспособить практически под любой микропроцессор.

Linux (GNU/Linux) – семейство Unix-подобных операционных систем на базе ядра Linux, включающих тот или иной набор утилит и программ проекта GNU (проект по разработке свободного программного обеспечения, запущенный известным программистом и сторонником СПО Ричардом Столлманом 27 сентября 1983 года в Массачусетском технологическом институте).

	Ричард Мэттью Столлман – основатель движения свободного программного обеспечения, проекта GNU, Фонда свободного программного обеспечения и Лиги за свободу программирования. Автор концепции «копилефта» (в противоположность традиционному подходу к авторскому праву, при котором ограничивается свобода копирования произведений, копилефт стремится использовать законы об авторском праве для расширения прав и свобод людей).
---	---

Linux-системы реализуются на модульных принципах, стандартах и соглашениях, заложенных в Unix. Подобные системы используют монолитное ядро для управления процессами, сетевыми функциями, периферией и доступом к файловой системе. Драйверы устройств либо интегрированы непосредственно в ядро, либо добавлены в виде модулей, загружаемых во время работы системы.

Linux доминирует на рынке интернет-серверов и смартфонов (операционная система Android имеет в основе Linux ядро). Linux полностью бесплатная система, в основном построенная на открытом программном обеспечении.

Принципы взаимодействия программ в Windows.

Интерфейс вызовов функций в Windows – доступ к системным ресурсам осуществляется через целый ряд системных функций. Совокупность таких функций называется прикладным программным интерфейсом или **API** (Application Programming Interface). Для взаимодействия с Windows приложение запрашивает функции API, с помощью которых реализуются все необходимые системные действия, такие как выделение памяти, вывод на экран, создание окон и т.п.

Функции API содержатся в библиотеках динамической загрузки (Dynamic Link Libraries, или DLL), которые загружаются в память только в тот момент, когда к ним происходит обращение, т.е. при выполнении программы.

Многозадачность в Windows

В Windows два типа многозадачности: основанный на процессах и основанный на потоках.

Процесс – это программа, которая выполняется. При многозадачности такого типа две или более программы могут выполняться параллельно.

Поток – это отдельная часть исполняемого кода. В многозадачности данного типа отдельные потоки внутри одного процесса также могут выполняться одновременно.

7. Стандарт POSIX

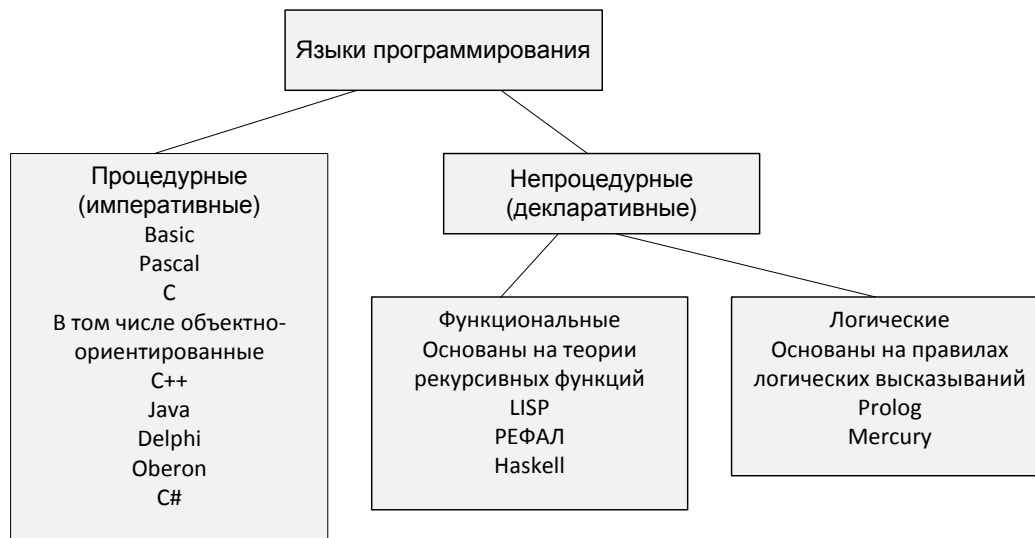
Стандарт POSIX – переносимый интерфейс операционных систем) — набор стандартов, описывающих интерфейсы между операционной системой и прикладной программой (системный API), библиотеку языка С и набор приложений и их интерфейсов.

Традиционно говорят о двух мирах, двух системах мировоззрения, присущих пользователям операционных систем Windows и UNIX.

В 1985 году принят стандарт POSIX. Это стандарт на интерфейсы UNIX-подобных ОС.

Однако никто не запрещает системам, не являющимся клонами (потомками) UNIX, поддерживать стандарт POSIX.

8. Парадигмы (стили) программирования



Язык программирования строится в соответствии с базовой моделью вычислений и парадигмой программирования (пример парадигмы: объектно-ориентированное программирование).

Парадигма программирования — это совокупность идей и понятий, определяющих стиль написания компьютерных программ (подход к программированию).

Императивное программирование:

Программа = последовательность действий, дающих указания компьютеру о том, **как** получить решение.

Конструкции языка:

последовательность действий;

условная конструкция;

цикл.

Пример:

Функция с именем `add`, принимает на вход массив и возвращает сумму всех его элементов.

Псевдокод:

```
function add(array)
{
    let result = 0
    for (let i = 0; i < array.length; i++)
        result += array[i];
    return result;
}
```


Декларативное программирование:

Программа = описание действий, которые необходимо выполнить компилятору для получения результата.

Отвечает на вопрос **что** надо выполнить.

Пример:

Функция с именем `add`, принимает на вход массив и возвращает сумму всех его элементов.

Псевдокод:

```
function add(array)
{
    return array.reduce((prev, current) => prev +
        current, 0)
}
```

Функциональное программирование:

Программа = система определений и функций, описывающих что нужно вычислить, а как это сделать – решает транслятор; последовательность действий не прослеживается.

Раздел дискретной математики. Основой функционального программирования является лямбда-исчисление

Объектно-ориентированное программирование:

Программа = несколько взаимодействующих объектов + функциональность (действия и данные распределяются между этими объектами).

Распределённое (параллельное) программирование:

Программа = совокупность описаний процессов, которые могут выполняться как параллельно (при наличии нескольких процессоров), так и в псевдопараллельном режиме (при наличии одного процессора).

Логическое программирование:

Программа = система определений вида «условие => новый факт». Программа представляет собой описание фактов и правил вывода в некотором логическом исчислении. Результат, (который часто записывается как вопрос), получается системой путем логического вывода. Раздел математической логики.

Распределённое (параллельное) программирование:

Программа = совокупность описаний процессов, которые могут выполняться как параллельно (при наличии нескольких процессоров), так и в псевдопараллельном режиме (при наличии одного процессора).

Визуальное программирование:

Программа = способ создания программы для ЭВМ путём манипулирования графическими объектами вместо написания её текста.

Визуальное программирование позволяет программировать на уровне алгоритмов, а не программного кода.

Пакет визуального программирования генерирует, написанный на языках программирования (1GL, 2GL, 3GL), на основании составленной программистом «блок-схемы» в автоматическом режиме.

Аспектно-ориентированное программирование:

Программа = к уже существующему коду добавляется дополнительного поведение, так называемой сквозной функциональности.

Пример функции возведения в квадрат в некоторых языках программирования.

Императивный C :	Функциональный Scheme :
<pre>int square(int x) { return x * x; }</pre>	<pre>(define square (lambda (x) (* x x)))</pre>
Конкатенативный Joy :	Конкатенативный Factor :
<pre>DEFINE square == dup * .</pre>	<pre>: square (x -- y) dup *;</pre>

9. TIOBE определила кандидатов на «язык года»

Голландская компания TIOBE Software BV известна как автор регулярно рассчитываемого рейтинга популярности языков программирования.

Индекс TIOBE — индекс, который оценивает популярность языков программирования, основываясь на количестве поисковых запросов, содержащих название языка. Расчет индекса происходит ежемесячно.

Очередной выпуск рейтинга языков программирования TIOBE показал стабильность среди лидеров списка

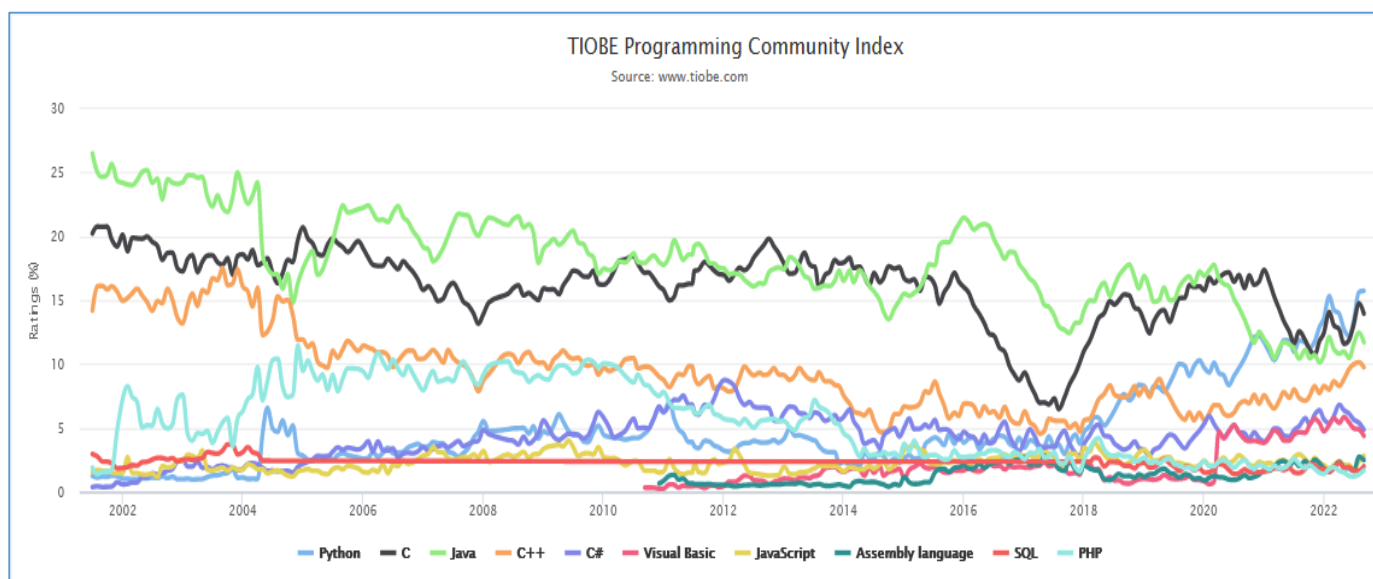


Иллюстрация: TIOBE, сентябрь 2022 года.

Четвёрка лидеров осталась неизменной: Python, C, Java и C++.

!! Языком 2020 и 2021 года является Python.

Заголовок сентября: Julia приближается к топ-20 индекса TIOBE

«Язык программирования Julia всего на 0,05% от топ-20. Язык Julia предназначен для численного анализа и вычислительной науки. В этой области существует множество конкурирующих языков: Так чем же выделяется Julia? Julia превосходит Matlab, потому что он намного современнее и его можно использовать бесплатно. Julia превосходит Python и R, потому что он намного быстрее. Поскольку спрос на работу с цифрами и модельным бизнесом огромен, у Julia есть серьезные шансы войти в топ-20 уже в ближайшем будущем. Отметим, что язык Rust тоже уже довольно давно стучится в двери топ-20, но пока безуспешно. Время покажет, постигнет ли Julia та же участь.» - Пол Янсен, генеральный директор TIOBE Software