



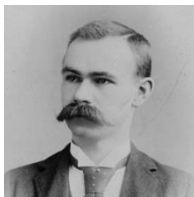
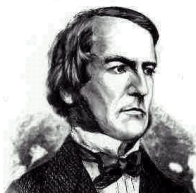
Институт
кибернетики

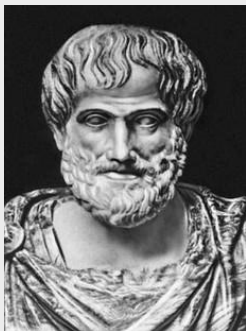
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ



ИНФОРМАТИКА В ЛИЦАХ

Татьяна Александровна Гладкова
программист
Институт кибернетики





Аристотель

(384 - 322 гг. до н.э)
Ученый и философ

Законы мышления

Аристотель Стагирский, один из величайших философов античной Греции, родился в 384 г. до н. э. в Стагире, греческой колонии во Фракии, недалеко от Афона. Рано потеряв родителей, Аристотель отправился сначала в Атарней, в Малой Азии, а затем, в 367 году, – в Афины. Там Аристотель стал учеником [Платона](#) и в течении 20 лет был участником платоновской Академии. В 343 г. Аристотель был приглашен Филиппом (царём Македонии) воспитывать его сына – 13-летнего Александра. В 335 г. Аристотель вернулся в Афины и создал там свою школу (Лицей, или перипатетическую школу).

Он пытался дать ответ на вопрос: «Как мы рассуждаем», изучал правила мышления. Подверг человеческое мышление всестороннему анализу.

Определил основные формы мышления:

- понятие
- суждение
- умозаключение.

Его трактаты по логике объединены в сборнике «Органон».

В книгах «Органона»:

- «Топика»
- «Аналитики»
- «Герменевтике»

и др.

Мыслитель разрабатывает важнейшие категории и законы мышления, создает теорию доказательства, формулирует систему дедуктивных умозаключений.

Дедукция позволяет выводить истинное знание о единичных явлениях, исходя из общих закономерностей.



Мухаммед ибн Муса Хорезми

(ок. 783 - ок. 850) - хорезмийский, центральноазиатский математик, астроном и географ, основатель классической алгебры.

Открытие алгоритма

Аль-Хорезми мы обязаны целыми двумя словами, без которых нам, пожалуй, уже было бы трудно себя представить. Это **Алгебра** и **Алгоритм**.

Алгебра.

В первую очередь аль-Хорезми известен книгой **«Китаб аль-Джебр ва-ль-Мукабала**. Именно от названия этой книги *аль-Джебр* (восстановление) и произошло слово «алгебра». Она называется «Книга восстановлений и противопоставлений». Еще есть вариант перевода «Книга сложения и вычитания». Хотя на самом деле, аль-Мукабала — это вовсе не вычитание, а перенос неизвестных и констант в разные части уравнения и приведение подобных.

Чтобы решить уравнение, Мухаммед аль-Хорезми переносил члены уравнения из одной части в другую с противоположным знаком (эта процедура и называлась «аль-джебр»), затем приводил подобные слагаемые («аль-мукабала») и лишь затем решал уравнение.

Так как в те времена отрицательные числа считались ненастоящими, то действие аль-джебр, как бы превращающее число из небытия в бытие, казалось чудом. Эту науку в Европе долго считали «великим искусством», рядом с «малым искусством» — арифметикой.

Алгоритм.

С алгоритмом оказалось не всё так просто. Само слово «алгоритм» происходит от имени хорезмского учёного Абу Абдуллах Мухаммеда ибн Муса аль-Хорезми (алгоритм - аль-Хорезми). Около 825 года он написал сочинение, в котором впервые дал описание придуманной в Индии позиционной десятичной системы счисления. Аль-Хорезми сформулировал правила вычислений в новой системе и, вероятно, впервые использовал цифру 0 для обозначения пропущенной позиции в записи числа (её индийское название арабы перевели как *as-sifr* или просто *sifr*, отсюда такие слова, как «цифра» и «шифр»). В первой половине XII века книга аль-Хорезми в латинском переводе проникла в Европу. Переводчик, имя которого до нас не дошло, дал ей название *Algoritmi de numero Indorum* («Алгоритмы о счёте индийском»). По-арабски же книга именовалась *Kutab аль-джебр ва-ль-мукабала* («Книга о сложении и вычитании»). Из оригинального названия книги происходит слово Алгебра (алгебра — аль-джебр — восполнение).

Таким образом, мы видим, что латинизированное имя среднеазиатского учёного было вынесено в заглавие книги, и сегодня считается, что слово «алгоритм» попало в европейские языки именно благодаря этому сочинению.



Джон Непер

(1550-1617)
шотландский
математик

Открытие логарифмов

В 1614 году шотландский математик **Джон Непер** изобрел таблицы логарифмов. Таблица логарифмов — небольшая таблица, с помощью которой можно узнать посредством весьма легких вычислений все геометрические размеры и движения. Она по справедливости названа небольшой, ибо по объему превосходит таблицы синусов, весьма легкой, потому что с ее помощью избегают всех сложных умножений, делений и извлечений корня, и все вообще фигуры и движения измеряются посредством выполнения более легких сложения, вычитания и деления на два. Она составлена из чисел, следующих в непрерывной пропорции. Принцип таблицы логарифмов заключался в том, что каждому числу соответствует свое специальное число - логарифм. Логарифмы очень упрощают деление и умножение. Например, для умножения двух чисел складывают их логарифмы. результат находят в таблице логарифмов. В дальнейшем им была изобретена логарифмическая линейка, которой пользовались до 70-х годов нашего века.

Таблицы Непера, расчет которых требовал очень много времени, были позже "встроены" в удобное устройство, чрезвычайно ускоряющее процесс вычисления - логарифмическая линейка. Непер же придумал в 1617 году (год его смерти) другой - не логарифмический - способ перемножения чисел и разработал оригинальный прибор для быстрого умножения — палочки Непера.

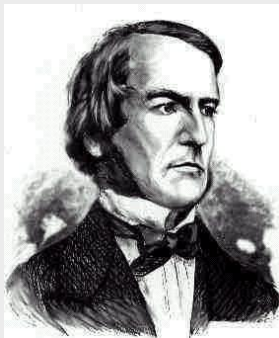
	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	10	12	14	16	18
3	3	6	9	12	15	18	21	24	27
4	4	8	12	16	20	24	28	32	36
5	5	10	15	20	25	30	35	40	45
6	6	12	18	24	30	36	42	48	54
7	7	14	21	28	35	42	49	56	63
8	8	16	24	32	40	48	56	64	72
9	9	18	27	36	45	54	63	72	81

Палочки Непера

	4	9	3	8	
1	4	9	3	8	
2	8	18	6	16	
3	12	27	9	24	
4	16	36	12	32	
5	20	45	15	40	
6	24	54	18	48	
7	28	63	21	56	
8	32	72	24	64	
9	36	81	27	72	

$$\begin{array}{r}
 14814 \\
 + 39504 \\
 + 24690 \\
 \hline
 1901130
 \end{array}$$

Умножение на палочках Непера
(4938 x 385 = 1901130)



Джордж Буль

(1815-1864)

английский
математик

Основоположник булевой алгебры

Считается основоположником математической логики (булевой алгебры).

Свои математические исследования Буль начал с разработки операторных методов анализа и теории дифференциальных уравнений, затем занялся математической логикой. В основных трудах Буля "математический анализ логики, являющийся опытом исчисления дедуктивного рассуждения" и "исследование законов мышления, в которых основаны математические теории логики и вероятности" были заложены основы математической логики.

Основное произведение Буля "Исследование законов мышления". Буль предпринял попытку построить формальную логику в виде некоторого "исчисления", "алгебры". Логические идеи Буля в последующие годы получили дальнейшее развитие. Логические исчисления, построенные в соответствии с идеями Буля, находят сейчас широкое применение в приложениях математической логики к технике, в частности к теории релейно-контактных схем. В современной алгебре есть булевы кольца, булевы алгебры — алгебраические системы, в программировании — переменные и константы типа **boolean**. Известно булево пространство, в математических проблемах управляющих систем — булев разброс, булево разложение, булева регулярная точка ядра.

В его работах логика обрела свой алфавит, свою орфографию и свою грамматику.



Ада Лавлейс

(1815-1852)

дочь
английского
поэта Байрона

Первый программист

Графиня **Ада Лавлейс**, дочь поэта Байрона, изучала астрономию, латынь, музыку и математику. Совместно с английским математиком Чарльзом Беббиджем она работала над созданием арифметических программ для его счетных машин. Ее работы в этой области были опубликованы в 1843 году. Однако в то время считалось неприличным для женщины издавать свои сочинения под полным именем и, Лавлейс поставила на титуле только свои инициалы. Поэтому ее математические труды, как и работы многих других женщин-ученых, долго пребывали в забвении.

В материалах Бэббиджа и комментариях Лавлейс намечены такие понятия, как подпрограмма и библиотека подпрограмм, модификация команд и индексный регистр, которые стали употребляться только в 50-х годах нашего века. Сам термин библиотека был введен Бэббиджем, а термины рабочая ячейка и цикл предложила Ада Лавлейс.

Графиню Лавлейс называют первым программистом, в ее честь назван язык программирования АДА.



Норберт Винер

(1894-1964)

доктор
философии,
профессор
математики
(США)

Новое научное направление - кибернетика

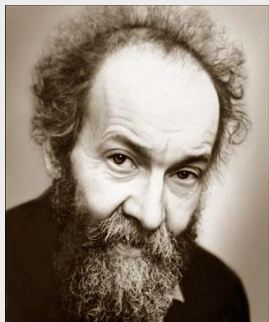
В 1948 году в США и Европе вышла книга Винера "Кибернетика или Управление и связь в животном и машине", ознаменовавшая своим появлением рождение нового научного направления - **кибернетики**.

Норберт Винер родился в городе Колумбия, штат Миссури, в семье выходца из России. В 14 лет по окончании колледжа он получил первую ученую степень - бакалавра искусств. Затем учился в Корнельском и Гарвардском университетах и в 17 лет получил степень магистра искусств, а через год стал доктором философии по специальности "математическая логика". С 1919 года и до своей кончины он работал в Массачусетском технологическом институте в качестве профессора математики.

Круг математических интересов Винера весьма широк. Ему принадлежат работы по теории вероятностей и статистике, по рядам и интегралам Фурье, теории потенциала, теории чисел, обобщенному гармоническому анализу и др.

Винер стал основателем кибернетической философии, основателем собственной школы, и его заслуга в том, что эта философия была передана ученикам и последователям. Именно школе Винера принадлежит ряд работ, которые, в конечном счете, привели к рождению Интернета.

Вместе с К.Шеноном Винер разработал статистические основы современной теории информации и ввел меру количества информации - бит.



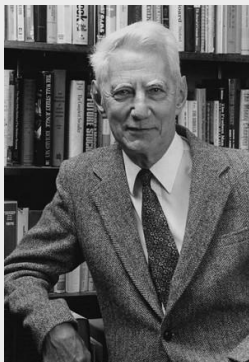
**Алексей Андреевич
Ляпунов**

(1911-1973)

советский
ученый

Новое научное направление - кибернетика

Алексей Андреевич Ляпунов - один из первых отечественных ученых, кто оценил значение кибернетики, внес большой вклад в ее становление и развитие. Еще в середине 50-х годов, когда в СССР считалась "буржуазной лженаукой", ученый активно выступил в защиту этого перспективного научного направления. Под его руководством в стране начались первые исследования в области кибернетики. Общие и математические основы кибернетики, вычислительные машины, программирование и теория алгоритмов, машинный перевод и математическая лингвистика, кибернетические вопросы биологии, философские и методологические аспекты развития современной науки - вот не полный перечень основных направлений науки, получившей интенсивное развитие по инициативе и при участии А.А.Ляпунова. В конце 50-х годов он сформулировал основные направления развития кибернетики, которые на протяжении десятков лет являлись основой теоретических и практических исследований в этой области. А.А.Ляпунову принадлежит разработка математической теории управляющих (кибернетических) систем. Он создал первые учебные курсы программирования и разработал операторный метод - по существу первый язык программирования, отличающийся от языка систем команд ЭВМ и разработанный после появления алгоритмических языков типа АЛГОЛ и другие. Большая роль А.А.Ляпунову принадлежит в распространении идей и методов кибернетики. **В 1958 году** под руководством А.А.Ляпунова начал выходить периодический сборник "Проблемы кибернетики", на страницах которого публиковались результаты отечественных исследований.



Клод Шеннон

(1916-2001)
американский
инженер и
математик

Теория информации и связи

Клод Элвуд Шеннон (англ. Claude Elwood Shannon) - американский инженер и математик. Человек, которого называют отцом современных теорий информации и связи.

В годы войны он занимался разработкой криптографических систем, и позже это помогло ему открыть методы кодирования с коррекцией ошибок. А в свободное время он начал развивать идеи, которые потом вылились в теорию информации. Исходная цель Шеннона заключалась в улучшении передачи информации по телеграфному или телефонному каналу, находящемуся под воздействием электрических шумов. И для того, чтобы решить эту проблему, ему пришлось сформулировать, что такое информация и чем определяется ее количество. В своих работах 1948-49 годов он определил количество информации через энтропию - величину, известную в термодинамике и статистической физике как мера разупорядоченности системы, а за единицу информации принял то, что впоследствии было названо "битом", то есть выбор одного из двух равновероятных вариантов.



Грейс Хоппер

(1906-1992)

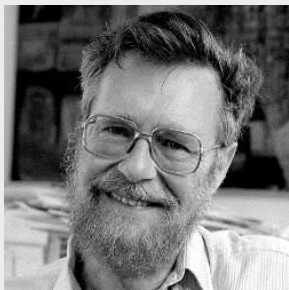
американский
компьютерный
ученый

Теория информации и связи

Грейс Хоппер - американский компьютерный учёный и контр-адмирал флота США.

Будучи первооткрывателем в своей области, она была одной из первых, кто писал программы для гарвардского компьютера Марк I. При работе на компьютере Mark-1 Грейс Хоппер и ее группой впервые были введены некоторые приемы, которые стали в дальнейшем широко использоваться в программистской практике. В частности, первыми инструментами, которые экономили труд программистов, были подпрограммы. Так вот, в августе 1944 года для Mark-1 была написана первая подпрограмма для вычисления $\sin(x)$.

Она разработала первый компилятор для компьютерного языка программирования, развила концепцию машинно-независимых языков программирования, что привело к созданию COBOL, одного из первых высокоуровневых языков программирования. Ей приписывается популяризация термина *debugging* для устранения сбоев в работе компьютера (поводом для появления этого слова послужил случай удаления мотылька из компьютера). Из-за многочисленных достижений и высокого чина во флоте её иногда называют «Удивительная Грейс», «Amazing Grace» (возможно, аллюзия на Amazing Grace). В её честь назван эсминец ВМФ США USS Hopper (DDG-70) и суперкомпьютер Cray XE6 «Hopper» Исследовательского вычислительного центра министерства энергетики (NERSC).



Эдсгер Дейкстра

(1930-2002)
голландский
ученый

Теория математической логики

Известность Дейкстре принесли его работы в области применения математической логики при разработке компьютерных программ.

Он активно участвовал в разработке языка программирования Algol и написал первый компилятор Algol-60.

Будучи одним из авторов концепции структурированного программирования, он проповедовал отказ от использования инструкции GOTO. Также ему принадлежит идея применения «семафоров» для синхронизации процессов в многозадачных системах и алгоритм нахождения кратчайшего пути на ориентированном графе с неотрицательными весами ребер, известный как Алгоритм Дейкстры.

В 1972 году Дейкстра стал лауреатом премии Тьюринга.

Дейкстра был активным писателем, его перу (он предпочитал авторучку клавиатуре) принадлежит множество книг и статей, самыми известными из которых являются книги «Дисциплина программирования» и «Заметки по структурному программированию», и статья «О вреде оператора GOTO».

Дейкстра также приобрел немалую известность за пределами академических кругов благодаря своим резким и афористичным высказываниям по актуальным проблемам компьютерной индустрии.



Винтон Серф

(родился
23 июня 1943г.)
американский
ученый

Протоколы TCP/IP

Винтон Серф (Vinton Cerf) — «прародитель» Интернета, создатель легендарного протокола TCP/IP. Именно Серф первым высказал идею объединения сетей в глобальную информационную структуру и предложил термин Интернет (Internet).

В 2000-2007 годах председатель Международной корпорации по распределению имен и номеров в Интернете (ICANN), с 2005 года вице-президент компании Google Inc., одновременно занимающий в ее иерархии должность главного интернет-проповедника.

Находясь в Калифорнийском университете Серф, начал работать в области создания непосредственного предшественника Интернета - сети изначально призванной обеспечить связь между компьютерами около 30 американских университетов. В 1970 году в рамках данного проекта Серф начал сотрудничать с другим будущим "отцом Интернета" Робертом Каном (Robert E. Kahn) . В 1972 году Серф вернулся в Стэнфордский университет, где получил место доцента (assistant professor), на котором он оставался вплоть до 1976 года . Находясь в Стэнфорде, Серф продолжил сотрудничество с Каном и в 1974 году опубликовал вместе с ним статью "A Protocol for Packet Network Intercommunication" ("Протокол для пакетной межсетевой коммуникации"), в которой был описан протокол передачи данных TCP (transmission-control protocol), позволяющий передавать данные между разными сетями. Благодаря этому открытию Серфа и Кана нередко называли "отцами Интернета".

MS DOS

MS-DOS (англ. Microsoft Disk Operating System) — дисковая операционная система для компьютеров на базе архитектуры x86. MS-DOS самая известная ОС среди семейства DOS-совместимых операционных систем и самая используемая среди IBM PC-совместимых компьютеров на протяжении 80-х и до середины 90-х годов, пока её не вытеснили операционные системы с графическим пользовательским интерфейсом, в основном из семейства Microsoft Windows. В 1981 году корпорация IBM разместила запрос на создание операционной системы, которая должна была использоваться в новом семействе компьютеров IBM PC. Microsoft выкупила права на операционную систему 86-DOS у Seattle Computer Products и начала работу по её модификации под требования IBM. Впоследствии IBM лицензировала MS-DOS и выпустила её в августе 1981 года под наименованием PC DOS 1.0. В дальнейшем разработка MS-DOS и PC DOS велась Microsoft и IBM совместно вплоть до 1993 года, когда были выпущены два отдельных релиза — 6.0 от Microsoft и 6.1 от IBM. За время существования MS-DOS вышло восемь основных релизов (1.00, 2.00 и т. д.) и два десятка промежуточных (3.10, 3.30 и т. п.), пока в 2000 году Microsoft не прекратила её разработку.



Питер Нортона

(родился
14 ноября 1943г.)
американский
программист

Norton Utilities

Вся профессиональная жизнь **Питера Нортона** связана с программированием.

В 1982 году он случайно стер исходный файл с жесткого диска компьютера. Можно было потратить несколько дней и занести в компьютер всю информацию по новой, однако Нортона сделал по-другому — быстро написал программу, которая восстанавливает потерянные данные.

Судьба заставила его написать программу, являющуюся прообразом современных утилит. Затем появились и другие утилиты, способствующие облегчению труда программиста.

В 1982 году все разработанные утилиты были собраны воедино и выпущены под единым названием **Norton Utilities**.

Сейчас существует очень большое количество утилит. К их числу относятся, например, различные архиваторы, антивирусные программы, программы для диагностики компьютера, программы для обслуживания дисков и многие другие программные средства.

Небольшая программа Norton Commander была главным инструментом повседневного использования и в итоге стала символом DOS-эпохи в куда большей степени, чем собственно текстовый интерфейс DOS.



**Дмитрий Николаевич
Лозинский**

(родился
27 июня 1939г.)
русский
программист

Первые антивирусные программы

Дмитрий Николаевич Лозинский – один из тех, кто определил развитие отечественного программирования и стоял у истоков первых русских антивирусных решений.

Осенью 1988 г. обнаружил в Госплане вирус Vienna. Написал программу, назвав ее Aidstest. С тех пор пришлось ее совершенствовать, добавляя обработку новых вирусов.

Вклад Лозинского в борьбу с компьютерными вирусами трудно переоценить. Благодаря его упорному многолетнему труду по анализу компьютерных вирусов и совершенствованию программы Aidstest, а также других отечественных антивирусов, в нашей стране в последнее время практически не происходили глобальные эпидемии компьютерных вирусов. Спасенные данные на компьютерах и сэкономленное рабочее время миллионов пользователей принесли государству большой экономический эффект, который, по оценкам экспертов, определяется величиной 100 млн. долл.

Лозинский активно участвовал в разработке русского антивируса Dr.Web. Совместно с автором антивируса, Игорем Даниловым, был создан алгоритм записи процедур обнаружения вирусов и лечения файлов, который лежит в основе внешней вирусной базы Dr.Web и позволяет автоматизировать анализ новых угроз.



**Евгений Валентинович
Касперский**

(родился
4 октября 1965г.)
русский
программист

Антивирусные программы

Евгений Валентинович Касперский — российский программист, один из ведущих мировых специалистов в сфере информационной безопасности.

До 1991 года работал в многопрофильном научно-исследовательском институте Министерства обороны СССР.

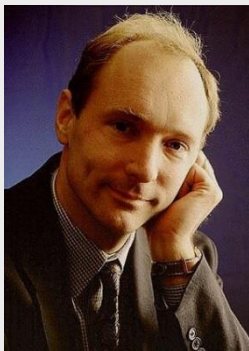
Начал изучение феномена компьютерных вирусов в октябре 1989 года, когда на его компьютере был обнаружен вирус «Cascade».

С 1991 по 1997 год работал в НТЦ «КАМИ», где вместе с группой единомышленников развивал антивирусный проект «AVP» (сейчас «Антивирус Касперского»).

В 1997 году Евгений Касперский стал одним из основателей «Лаборатории Касперского». На сегодняшний день Евгений Касперский – один из ведущих мировых специалистов в области защиты от вирусов.

Он является автором большого числа статей и обзоров по проблеме компьютерной вирусологии, регулярно выступает на специализированных семинарах и конференциях в России и за рубежом.

Евгений Валентинович Касперский – член Организации исследователей компьютерных вирусов (CARO), которая объединяет экспертов в этой области.



Тим Бернес-Ли

(родился
8 июня 1955г.)
английский
ученый,
изобретатель

Создатель World Wide Web

Тим Бернес-Ли — человек, перевернувший представление о всемирной сети — создатель World Wide Web и системы гипертекста.

В 1989 г. выпускник Оксфордского университета, сотрудник Европейского центра ядерных исследований в Женеве (CERN) Бернес-Ли разработал язык гипертекстовой разметки Web-страниц HTML, подарив пользователям возможность просмотра документов на удаленных компьютерах. В 1990 г. Тим изобрел первый примитивный браузер, а его компьютер, естественно, считается первым Web-сервером.

Бернес-Ли не запатентовал свои судьбоносные открытия, что в алчном мире, в общем-то, не редкость (вспомните, к примеру, Дугласа Энгельбарта и его легендарную мышь). В книге *Weaving the Web* («Плетение Паутины») он признался, что в нужное время просто не стал зарабатывать на собственных изобретениях, посчитав (как ни странно) сею идею рискованной. «Место под солнцем» тут же заняли мировые гиганты Microsoft и Netscape. В 1994 г. Бернес-Ли возглавил созданный им Консорциум World Wide Web (W3C), занимающийся разработкой стандартов Интернета. Нельзя сказать, что его имя известно широкому кругу пользователей, тем не менее, за разработки web-технологий Бернес-Ли не раз удостоивался почетных премий и наград.



Билл Гейтс

(родился
28 октября 1955г.)
американский
предприниматель

Один из создателей компании Microsoft

Уильям Генри Гейтс III (англ. William Henry Gates III), более известный как Билл Гейтс (англ. Bill Gates) — американский предприниматель и общественный деятель, филантроп, один из создателей (совместно с Полом Алленом) и бывший крупнейший акционер компании Microsoft. До июня 2008 года являлся руководителем компании, после ухода с поста остался в должности её неисполнительного председателя совета директоров.

В январе 1975 года Пол Аллен прочитал в журнале Popular Electronics статью о новом персональном компьютере Altair 8800. После прочтения статьи Гейтс связался с президентом компании Micro Instrumentation and Telemetry Systems (MITS), разработчиком нового компьютера Эдом Робертсом и сообщил ему, что он и его друг работают над программным обеспечением данного компьютера. Президент MITS пригласил в свой офис Пола и он продемонстрировал рабочий интерпретатор языка BASIC для их компьютера, и через несколько недель Пол и Билл уже работали на MITS. Свою компанию они думали назвать «Аллен и Гейтс», но посчитали что это больше подходит для юридической конторы, и тогда Пол предложил — Micro-Soft, от microprocessors и software. В течение года работы на компанию MITS дефис в названии компании Гейтса и Аллена исчез, и **26 ноября 1976 года** в Окружном секретариате штата Нью-Мехико была зарегистрирована новая торговая марка «Microsoft». В 1980 году компания Microsoft подписала контракт с IBM, по которому должна была разработать для IBM MS-DOS. Сотрудничество с IBM продолжалось и 20 ноября 1985 года появилась новая операционная система Microsoft Windows. Так началась эпоха Windows — операционная система, прославившая и сделавшая богатейшим человеком Гейтса.



Линус Торвальдс

(родился
28 декабря 1969г.)

финно-
американский
программист

Операционная система Linux

Линус Бенедикт Тóрвальдс, или Тóрвальдс (швед. Linus Benedict Torvalds) — финно-американский программист, хакер - создатель известной во всем мире операционной системы.

В начале 1991 года он принялся писать собственную платформу, ориентированную на среднего потребителя, которую можно было бы распространять бесплатно посредством Интернета. Новая система обрела название Linux. За 10 лет Linux стал реальным конкурентом продуктов, выпускаемых Microsoft, способным потеснить монополию этой компании на рынке системного и серверного ПО. За 10 лет Linux прошел путь от игрушки нескольких сотен фанатов и энтузиастов, выполнявшей пару десятку команд в примитивной консоли, до профессиональной многопользовательской и многозадачной 32-разрядной ОС с оконным графическим интерфейсом, по спектру своих возможностей, стабильности и мощности многократно превосходящей Microsoft Windows 95? 98 и NT и способной работать практически на любом современном IBM-совместимом компьютере.

Сегодня Linux – это мощная UNIX-подобная платформа, включающая в себя практически все функции и еще целый комплекс собственных, нигде больше не встречающихся свойств. Благодаря высокой производительности и надежности она стала одной из самых популярных платформ для организации http-серверов.



**Андрей Андреевич
Марков**

(1903 – 1979)

математик, чл.-
корр. АН СССР

Русские ученые

Андрей Андреевич Марков (младший) — математик, чл.-корр. АН СССР, сын выдающегося математика, специалиста по теории вероятности, тоже Маркова Андрея Андреевича (старшего).

Основные труды по топологии, топологической алгебре, теории динамических систем, теории алгоритмов и конструктивной математике. Доказал неразрешимость проблемы гомеоморфизма в топологии, создал школу конструктивной математики и логики в СССР, автор понятия нормального алгоритма.

С 1959 и до конца жизни Андрей Андреевич заведовал кафедрой математической логики мехмата МГУ. Работал во многих областях (теория пластичности, прикладная геофизика, небесная механика, топология и др.), но наибольший вклад внёс в математическую логику (в частности, основал конструктивное направление в математике), теорию сложности алгоритмов и кибернетику. Создал большую математическую школу, его ученики работают сейчас во многих странах. Писал стихи, которые при жизни не публиковались.



**Андрей Николаевич
Колмогоров**

(1903 – 1987)
советский
математик

Русские ученые

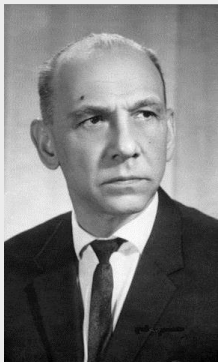
Широта научных интересов и научных занятий **А.Н. Колмогорова** имеет мало precedентов в XX веке, если вообще имеет таковые. Их спектр простирается от метеорологии до стиховедения.

В известной хрестоматии Ван Хейеноорта «От Фреге до Геделя», посвященной математической логике, можно найти английский перевод двадцатидвухлетнего Колмогорова статьи, которую автор хрестоматии охарактеризовал как «первое систематическое изучение интуиционистской логики». Статья была первой отечественной статьей по логике, содержащей собственно математические результаты.

Колмогоров заложил основы теории операций над множествами. Ему принадлежит существенная роль в превращении теории информации Шеннона в строгую математическую науку, а также построение теории информации на принципиально ином, отличном от шенноновского, фундаменте.

Он является одним из основоположников теории динамических систем, ему принадлежит определение общего понятия алгоритма.

В математической логике он внес выдающийся вклад в теорию доказательств, в теории динамических систем — в развитие так называемой эргодической теории, куда он достаточно неожиданно сумел внести и успешно применить идеи теории информации.



**Анатолий Алексеевич
Дородницын**

(1910 – 1994)
советский
математик

Русские ученые

Анатолий Алексеевич Дородницын широко известен своими выдающимися научными трудами по математике, аэродинамике и метеорологии, определяющей ролью в создании вычислительной гидродинамики.

Многое в нем определялось природной одаренностью и незаурядным трудолюбием, личными склонностями, преданностью науке и любовью к вычислениям, которые он до конца жизни выполнял самостоятельно.

Если все это и позволяет угадывать истоки формирования личности ученого, то основы широты тематики его научных исследований остаются загадкой.

А. А. Дородницын опубликовал труды по обыкновенным дифференциальным уравнениям, алгебре, метеорологии, теории крыла (эллиптические уравнения), пограничному слою (параболические уравнения), сверхзвуковой газовой динамике (гиперболические уравнения), численному методу интегральных соотношений (для уравнений всех этих типов), методу малого параметра для уравнений Навье-Стокса, а также по различным вопросам информатики.



**Андрей Петрович
Ершов**

(1931 – 1988)
советский
ученый

Русские ученые

Андрей Петрович Ершов - один из зачинателей теоретического и системного программирования, создатель Сибирской школы информатики. Его существенный вклад в становление информатики как новой отрасли науки и нового феномена общественной жизни широко признан в нашей стране и за рубежом.

Фундаментальные исследования А. П. Ершова в области схем программ и теории компиляции оказали заметное влияние на его многочисленных учеников и последователей. Книга А. П. Ершова "Программирующая программа для электронной вычислительной машины БЭСМ" была одной из первых в мире монографий по автоматизации программирования.

Работы Ершова по технологии программирования заложили основы этого научного направления в нашей стране.

Более 20 лет тому назад он начал эксперименты по преподаванию программирования в средней школе, которые привели к введению курса информатики и вычислительной техники в средние школы страны и обогатили нас тезисом "программирование - вторая грамотность".

Первые универсальные языки

Обратимся к истокам развития вычислительной техники. Вспомним самые первые компьютеры и программы для них. Это была эра программирования непосредственно в машинных кодах, а основным носителем информации были перфокарты и перфоленты. Программисты обязаны были знать архитектуру машины досконально. Программы были достаточно простыми, что обуславливалось, во-первых, весьма ограниченными возможностями этих машин, и, во-вторых, большой сложностью разработки и, главное, отладки программ непосредственно на машинном языке. Вместе с тем такой способ разработки давал программисту просто невероятную власть над системой. Становилось возможным использование таких хитроумных алгоритмов и способов организации программ, какие и не снились современным разработчикам. Например, могла применяться (и применялась!) такая возможность, как самомодифицирующийся код. Знание двоичного представления команд позволяло иногда не хранить некоторые данные отдельно, а встраивать их в код как команды. И это далеко не полный список приемов, владение хотя бы одним из которых сейчас сразу же продвигает вас до уровня «гуру» экстра-класса.

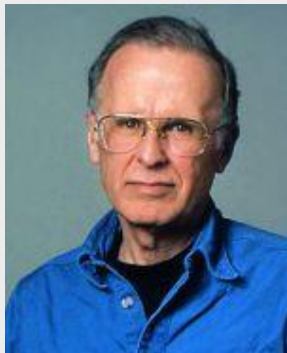
Ассемблер

Первым значительным шагом представляется переход к языку ассемблера.

Ассемблер (от англ. assembler — сборщик) — транслятор исходного текста программы, написанной на языке ассемблера, в программу на машинном языке.

Не очень заметный, казалось бы, шаг — переход к символическому кодированию машинных команд — имел на самом деле огромное значение. Программисту не надо было больше вникать в хитроумные способы кодирования команд на аппаратном уровне. Более того, зачастую одинаковые по сути команды кодировались совершенно различным образом в зависимости от своих параметров. Появилась также возможность использования макросов и меток, что также упрощало создание, модификацию и отладку программ. Появилось даже некое подобие переносимости — существовала возможность разработки целого семейства машин со сходной системой команд и некоего общего ассемблера для них, при этом не было нужды обеспечивать двоичную совместимость.

Вместе с тем, переход к новому языку таил в себе и некоторые отрицательные (по крайней мере, на первый взгляд) стороны. Становилось почти невозможным использование всяческих хитроумных приемов сродни тем, что упомянуты выше. Кроме того, здесь впервые в истории развития программирования появились два представления программы: в исходных текстах и в откомпилированном виде. Сначала, пока ассемблеры только транслировали мнемоники в машинные коды, одно легко переводилось в другое и обратно, но затем, по мере появления таких возможностей, как метки и макросы, дизассемблирование становилось все более и более трудным делом. К концу ассемблерной эры возможность автоматической трансляции в обе стороны была утеряна окончательно. В связи с этим было разработано большое количество специальных программ-дизассемблеров, осуществляющих обратное преобразование, однако в большинстве случаев они с трудом могут разделить код и данные. Кроме того, вся логическая информация (имена переменных, меток и т.п.) теряется безвозвратно. В случае же задачи о декомпиляции языков высокого уровня примеры удовлетворительного решения проблемы и вовсе единичны.



Джон Бэкус

(1924-2007)

американский
ученый в
области
информатики

Язык программирования Фортран (FORTRAN)

Бэкус (John Backus) был руководителем команды, разработавшей первый высокоуровневый язык программирования ФОРТРАН, изобретателем формы Бэкуса - Наура, одной из самых универсальных нотаций, используемых для определения синтаксиса формальных языков.

В 1950 году Джон Бэкус, уже имея степень магистра математики, пришел в фирму IBM. Ему было тогда 25 лет. Вскоре он возглавил группу, разрабатывавшую интерпретатор «Быстрый кодировщик» для компьютера IBM 701, а позже принял участие в создании усовершенствованного варианта этой машины, модели IBM 704. В 1953 году Бэкус предложил создать для компьютера IBM 704 язык, позволяющий записывать команды почти в обычной алгебраической форме, и компилятор для него. Данное предложение быстро нашло поддержку, так как в фирме искали пути, обеспечивающие увеличение сбыта своих компьютеров, и потому стремились сделать их более привлекательными, «дружественными» пользователю.

Фортра́н (Fortran) - первый язык программирования высокого уровня, имеющий транслятор. Создан в период **с 1954 по 1957 год**. Название Fortran является сокращением от FORmula TRANslator (переводчик формул). Фортран широко используется в первую очередь для научных и инженерных вычислений. Одно из преимуществ современного Фортрانا - большое количество написанных на нём программ и библиотек подпрограмм.

Язык программирования Фортран (FORTRAN)

Пример

```
101 FORMAT (F10.2, F10.8, F10.2)
102 FORMAT (I5, F10.2, F10.2, F10.2)
READ (1,101) BAL, RATE, PMT N = 0
RATE = RATE/12
2 N = N + 1
XINT = RATE*BAL
PRIN = PMT - XINT
IF (PRIN) 3, 3, 4
3 STOP
4 BAL = BAL - PRIN
IF (BAL) 5, 5, 6
6 WRITE (3, 102) N, XINT, PRIN, BAL
GO TO 2
5 PRIN = PRIN + BAL
BAL = 0.0
WRITE (3, 102) N, XINT, PRIN, BAL
STOP
END
```



Джон Маккарти

(1927-2011)

выдающийся
американский
информатик

Язык программирования Лисп (LISP)

1958 год - Лисп (LISP, от англ. LISt Processing language — «язык обработки списков»; современное написание: Lisp) — семейство языков программирования, программы и данные в которых представляются системами линейных списков символов. Создатель Лиспа Джон Маккарти занимался исследованиями в области искусственного интеллекта и созданный им язык по сию пору является одним из основных средств моделирования различных аспектов искусственного интеллекта.

Является вторым в истории (после Фотрана) используемым по сей день высокоуровневым языком программирования, а также первым из сохранившихся в использовании языков, использующих автоматическое управление памятью и сборку мусора.

Является языком системного программирования для так называемых лисп-машин, производившихся в 1980-годы, например, фирмой Symbolics.

Пример программы на Lisp

```
(DEFUN FACTORIAL (N)
  (LOOP FOR I FROM 1 TO N
        FOR FAC = 1 THEN (* FAC I)
        FINALLY (RETURN FAC)))
```

Язык программирования Алгол-60 (Algol 60)



**Группа разработчиков
Алгол-60**

Алгол был разработан в 1958 году как универсальный язык программирования для широкого круга применений, а затем в **1960 году** доработан комитетом, созданным Международной федерацией по обработке информации (IFIP). В комитет вошёл ряд ведущих европейских и американских учёных и инженеров-разработчиков языков.

Среди них были:

- Джон Бэкус — один из создателей Фортрана,
- Джозеф Уэгстен — впоследствии возглавлял комитет по разработке языка Кобол,
- Джон Маккарти — автор языка Лисп разработанного одновременно с Алголом,
- Петер Наур — впоследствии доработал «нормальную форму Бэкуса», завершив разработку БНФ,
- Эдсгер Дейкстра — нидерландский учёный, впоследствии получивший широкую известность как один из создателей структурного программирования и сторонник математического подхода к программированию, будущий лауреат Премии Тьюринга.

Этот язык дал начало целому семейству Алгол-подобных языков (важнейший представитель — Pascal). В 1968 году появилась новая версия языка. Она не нашла столь широкого практического применения, как первая версия, но была весьма популярна в кругах теоретиков. Язык был достаточно интересен, так как обладал многими уникальными на тот момент характеристиками.

Кобол (Cobol)

Пример

```
1 IDENTIFICATION DIVISION.
2 PROGRAM-ID. SUP-OF-PRICES.
3 AUTHOR. T-FRATT.
4 ENVIRONMENT DIVISION.
5 CONFIGURATION SECTION.
6 SOURCE-COMPUTER. SURL.
7 OBJECT-COMPUTER. SURL.
8 INPUT-OUTPUT SECTION.
9 FILE-CONTROL.
10 SELECT IMP-DATA ASSIGN TO INPUT.
11 SELECT RESULT-FILE ASSIGN TO OUTPUT.
12 DATA DIVISION.
13 FILE SECTION.
14 FD IMP-DATA LABEL RECORD IS OMITTED.
15
16 01 ITEM-PRICE.
17 03 PRICE PICTURE 9999V99.
18 WORKING-STORAGE SECTION.
19 77 TOT PICTURE 9999V99. VALUE 0. USAGE IS COMPUTATIONAL.
20 01 SUR-LINE.
21 02 FILLER VALUE = SUR-PICTURE N(12).
22 02 SUR-OUT PICTURE $$,555.559.99.
23 03 COUNT-OUT PICTURE ZZZ9.
24 ... DUFFYE SIGNME.
25 PROCEDURE DIVISION.
26 START.
27 OPEN INPUT IMP-DATA AND OUTPUT RESULT-FILE.
28 READ-DATA.
29 READ IMP-DATA AT END GO TO PRINT-LINE.
30 ADD PRICE TO TOT.
31 ADD 1 TO COUNT.
32 MOVE PRICE TO PRICE-OUT.
33 MOVE ITEM TO ITEM-OUT.
34 WRITE RESULT-LINE FROM ITEM-LINE.
35 GO TO READ-DATA.
36 PRINT-LINE.
37 MOVE TOT TO SUR-OUT.
38 ... DUFFYE OPERATOR.
39 CLOSE IMP-DATA AND RESULT-FILE.
40 STOP RUN.
```

В 1960 году был создан язык программирования **Cobol**.

Он задумывался как язык для создания коммерческих приложений, и он стал таковым. На Коболе написаны тысячи прикладных коммерческих систем.

Отличительной особенностью языка является возможность эффективной работы с большими массивами данных, что характерно именно коммерческих приложений. Популярность Кобола столь высока, что даже сейчас, при всех его недостатках (по структуре и замыслу Кобол во многом напоминает Фортран) появляются новые его диалекты и реализации. Так недавно появилась реализация Кобола, совместимая с Microsoft .NET, что потребовало, вероятно, внесения в язык некоторых черт объектно-ориентированного языка.





слева направо:
Томас Курц и
Джон Кемени

Язык программирования Бейсик (Basic)

Двумя яркими представителями, в разработке языков программирования, были сотрудники математического факультета Дартмутского колледжа Томас Курц (Thomas Kurtz) и Джон Кемени (John Kemeny).

В 1963 году в ими был создан язык программирования **BASIC** (Beginners' All-Purpose Symbolic Instruction Code — многоцелевой язык символических инструкций для начинающих).

Язык задумывался в первую очередь как средство обучения и как первый изучаемый язык программирования. Он предполагался легко интерпретируемым и компилируемым. Надо сказать, что BASIC действительно стал языком, на котором учатся программировать (по крайней мере, так было еще несколько лет назад; сейчас эта роль отходит к Pascal). Было создано несколько мощных реализаций BASIC, поддерживающих самые современные концепции программирования (ярчайший пример — Microsoft Visual Basic).

Пример

Программа, вычисляющая сумму $12 + 22 + \dots + 102$

```
100 REMARK S IS SUM; I IS INDEX
200 LET S=0
300 FOR I=1 TO 10
400 LET S = S +I*I
500 NEXT I
600 REMARK NEXT IS END OF LOOP
700 PRINT "SUM IS ", S
800 STOP
900 REMARK OTHER STATEMENTS: IF S>400 THEN 200 - (BRANCH TO 200)
1000 REMARK OTHER STATEMENTS: DIM A(20) - A ARRAY OF 20
1100 REMARK OTHER STATEMENTS: GOSUB 100: RETURN - SUBROUTINES
1200 REMARK OTHER STATEMENTS: READ A-INPUT
```

Язык программирования ПЛ/1 (PL/1)

В 1961 г. в IBM осознали, что если они хотят сохранить первенство, то нужно сделать гигантский скачок в технологии программирования. Результатом этого решения была System/360, впечатляющая серия универсальных компьютеров.

Эти машины проектировались так, чтобы в равной степени удовлетворить потребности как в аналитических вычислениях, так и в обработке данных в сфере бизнеса, а также в широком диапазоне специальных приложений. Естественно, компьютер с таким широким спектром возможностей требовал мощного языка. Непростительно поздно поняв это, IBM лишь за полгода до окончания работ над аппаратным обеспечением начала заниматься языком.

Ассоциация SHARE, оценив перспективность этого дела, помогла IBM создать «комитет по разработке передового языка». В комитет вошли представители фирм Lockheed, Union Carbide, Standard Oil из Калифорнии, а также специалисты из отделов программирования и разработки языков самой IBM. Комитет возглавил специалист по компиляторам из фирмы IBM Джордж Радин (George Radin).

В 1964 году корпорация IBM создала язык PL/1, который был призван заменить Cobol и Fortran в большинстве приложений. Язык обладал исключительным богатством синтаксических конструкций. В нем впервые появилась обработка исключительных ситуаций и поддержка параллелизма. Надо заметить, что синтаксическая структура языка была крайне сложной. Пробелы уже использовались как синтаксические разделители, но ключевые слова не были зарезервированы. В частности, следующая строка — это вполне нормальный оператор на PL/1:

IF ELSE=THEN THEN THEN; ELSE ELSE

В силу таких особенностей разработка компилятора для PL/1 была исключительно сложным делом. Язык так и не стал популярен вне мира IBM.

Пример

```
NOO: PROCEDURE (X,Y) DECIMAL FIXED (5,0);  
DO WHILE (X<=Y);  
  IF X=Y THEN S=X*Y;  
  ELSE Y=Y-X;  
END;  
RETURN (X);  
END NOO;
```




Жан Ишбия

(1940г. – 2007г.)

французский ученый

Языки ADA и ADA 95

Жан Давид Ишбия (фр. Jean David Ichbiah)— французский учёный и главный разработчик (в 1977—1983 годы) языка программирования Ada. В то время он был членом научно-исследовательского подразделения по программированию французской компании по производству компьютерной техники Bull в Лувесьене. Ранее он уже разрабатывал экспериментальный язык — LIS (1972—1974) на основе Паскаля и Симулы и был одним из членов-учредителей Международной федерации по обработке информации..

В 1983 году под эгидой Министерства Обороны США был создан язык Ada. Язык замечателен тем, что очень много ошибок может быть выявлено на этапе компиляции. Кроме того, поддерживаются многие аспекты программирования, которые часто отдаются на откуп операционной системе (параллелизм, обработка исключений). В 1995 году был принят стандарт языка Ada 95, который развивает предыдущую версию, добавляя в нее объектно- ориентированность и исправляя некоторые неточности. Оба этих языка не получили широкого распространения вне военных и прочих крупномасштабных проектов (авиация, железнодорожные перевозки). Основной причиной является сложность освоения языка и достаточно громоздкий синтаксис (значительно более громоздкий, чем Pascal).



Никлаус Вирт

(родился
15 февраля 1934г.)
швейцарский
инженер и
исследователь мира
программирования

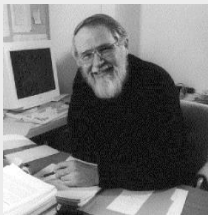
Язык программирования Паскаль (Pascal)

В 1970 году Никлаусом Виртом был создан язык программирования **Pascal**. Н. Вирт был одним из первых, кто ввел в практику принцип пошагового уточнения как ключевого для систематического создания программ. Помимо Паскаля, создал и другие алгоритмические языки (в их числе Modula-2 и Oberon). Вирт является одним из самых авторитетных в мире ученых в области компьютерных наук, его книга «Алгоритмы + структура данных = программы» считается одним из классических учебников по структурному программированию. Язык Pascal замечателен тем, что это первый широко распространенный язык для структурного программирования (первым, строго говоря, был Алгол, но он не получил столь широкого распространения). Впервые оператор безусловного перехода перестал играть основополагающую роль при управлении порядком выполнения операторов. В этом языке также внедрена строгая проверка типов, что позволило выявлять многие ошибки на этапе компиляции.

Отрицательной чертой языка было отсутствие в нем средств для разбиения программы на модули. Вирт осознал это и разработал язык Modula-2 (1978), в котором идея модуля стала одной из ключевых концепций языка. В 1988 году появилась Modula-3, в которую были добавлены объектно-ориентированные черты. Логическим продолжением Pascal и Modula являются язык Oberon и Oberon-2. Они характеризуются движением в сторону объектно- и компонентно-ориентированности.

Пример

```
PROGRAM TEST1;  
VAR X : INTEGER; Y : REAL;  
BEGIN  
  WRITE('ВВЕДИТЕ ЧИСЛО 1:'); READLN(X);  
  IF X = 5 THEN WRITELN('НЕ ОПРЕДЕЛЕНО')  
    ELSE BEGIN  
      IF X > 5 THEN Y:=X*(X - 5)  
        ELSE Y:=5*(X - 5);  
    WRITELN(Y)      END  
END.
```



Брайан Керниган
(родился 1 января 1942г.)



Деннис Ритчи
(1941г. – 2011г.)

С – подобные языки

Брайан Уилсон Керниган (англ. Brian Wilson Kernighan) — канадский учёный в области компьютерных технологий.

Деннис Макалестэйр Ритчи (англ. Dennis MacAlistair Ritchie) — компьютерный специалист, известен по участию в создании языков программирования BCPL, B, C, расширения ALTRAN для языка программирования FORTRAN, участием в разработке операционных систем Multics и UNIX.

В 1972 году Керниганом и Ритчи был создан язык программирования C.

Он создавался как язык для разработки операционной системы UNIX. С часто называют «переносимым ассемблером», имея в виду то, что он позволяет работать с данными практически так же эффективно, как на ассемблере, предоставляя при этом структурированные управляющие конструкции и абстракции высокого уровня (структуры и массивы). Именно с этим связана его огромная популярность и поныне. И именно это является его ахиллесовой пятой. Компилятор C очень слабо контролирует типы, поэтому очень легко написать внешне совершенно правильную, но логически ошибочную программу.

Пример

```
LONG SOME_FUNCTION();
/* INT */ OTHER_FUNCTION();

/* INT */ CALLING_FUNCTION()
{
    LONG TEST1;
    REGISTER /* INT */ TEST2;

    TEST1 = SOME_FUNCTION();
    IF (TEST1 > 0)
    ELSE TEST2 = 0;
    TEST2 = OTHER_FUNCTION();
    RETURN TEST2;
}
```



**Бьерн
Страуструп**

программист из Дании
родился
30 декабря 1950г.

С – подобные языки

Бьерн Страуструп (дат. Bjarne Stroustrup (Орхус, Дания)) — программист, автор языка программирования **C++** (P++).

В 1986 году Бьерн Страуструп создал первую версию языка **C++**, добавив в язык **C** объектно-ориентированные черты, взятые из Simula, и исправив некоторые ошибки и неудачные решения языка.

C++ продолжает совершенствоваться и в настоящее время, так в 1998 году вышла новая (третья) версия стандарта, содержащая в себе некоторые довольно существенные изменения.

Язык стал основой для разработки современных больших и сложных проектов. У него имеются, однако же, и слабые стороны, вытекающие из требований эффективности.

В 1999–2000 годах в корпорации Microsoft был создан язык C#. Он в достаточной степени схож с Java (и задумывался как альтернатива последнему), но имеет и отличительные особенности. Ориентирован, в основном, на разработку многокомпонентных Интернет-приложений.

Пример

```
100 REMARK S IS SUP; I IS INDEX
CLASS STACK { // ПЕРЕОПРЕДЕЛЕННЫЙ КЛАСС ANY
// СОДЕРЖИТ ФУНКЦИЮ NEXT
ANY*V;
VOID PUSH (ANY* P)
{
P->NEXT = V;
V = P;
}
ANY*POP ()
{
IF(V==0)RETURN ERROR_OBJ;
ANY*R=V;
V=V->NEXT;
RETURN R;
}
};
```



Джеймс Гослинг

программист из Канады
родился
19 мая 1955г.

Язык программирования Java

Джеймс Гослинг (англ. James Gosling) — автор объектно-ориентированного и кросс-платформенного языка программирования **Java**. Дата официального выпуска - **23 мая 1995 года**.

Приложения Java обычно транслируются в специальный байт-код, поэтому они могут работать на любой виртуальной Java-машине вне зависимости от компьютерной архитектуры. Достоинством подобного способа выполнения программ является полная независимость байт-кода от операционной системы и оборудования, что позволяет выполнять Java-приложения на любом устройстве, для которого существует соответствующая виртуальная машина.

Другой важной особенностью технологии Java является гибкая система безопасности, в рамках которой исполнение программы полностью контролируется виртуальной машиной.

Изначально язык назывался Oak («Дуб») разрабатывался Джеймсом Гослингом для программирования бытовых электронных устройств. Впоследствии он был переименован в Java и стал использоваться для написания клиентских приложений и серверного программного обеспечения. Назван в честь марки кофе Java, которая, в свою очередь, получила наименование одноимённого острова (Ява), поэтому на официальной эмблеме языка изображена чашка с дымящимся кофе. Существует и другая версия происхождения названия языка, связанная с аллюзией на кофе-машину как пример бытового устройства, для программирования которого изначально язык создавался.

Пример

Программа выводит на экран строку "Hello, world!"

```
CLASS HelloWorld {
    PUBLIC STATIC VOID MAIN
    (STRING ARGV[]) {
        SYSTEM.OUT.PRINTLN ("HELLO,
        WORLD"); } }
```

Скриптовые языки

В последнее время в связи развитием Интернет-технологий, широким распространением высокопроизводительных компьютеров и рядом других факторов получили распространение так называемые скриптовые языки. Эти языки первоначально ориентировались на использование в качестве внутренних управляющих языков во всякого рода сложных системах. Многие из них, однако же, вышли за пределы сферы своего изначального применения и используются ныне в совсем иных областях. Характерными особенностями данных языков являются, во-первых, их интерпретируемость (компиляция либо невозможна, либо нежелательна), во-вторых, простота синтаксиса, а в-третьих, легкая расширяемость. Таким образом, они идеально подходят для использования в часто изменяемых программах, очень небольших программах или в случаях, когда для выполнения операторов языка затрачивается время, несопоставимое со временем их разбора. Было создано достаточно большое количество таких языков, перечислим лишь основные и наиболее часто используемые.

Скриптовые языки

JavaScript

Язык был создан в компании Netscape Communications в качестве языка для описания сложного поведения веб-страниц. Первоначально назывался LiveScript, причиной смены названия получили маркетинговые соображения. Интерпретируется браузером во время отображения веб-страницы. По синтаксису схож с Java и (отдаленно) с C/C++. Имеет возможность использовать встроенную в браузер объектную функциональность, однако подлинно объектно-ориентированным языком не является.

VBScript

Язык был создан в корпорации Microsoft во многом в качестве альтернативы JavaScript. Имеет схожую область применения. Синтаксически схож с языком Visual Basic (и является усеченной версией последнего). Так же, как и JavaScript, исполняется браузером при отображении веб-страниц и имеет ту же степень объектно-ориентированности.

Perl

Язык создавался в помощь системному администратору операционной системы Unix для обработки различного рода текстов и выделения нужной информации. Развился до мощного средства работы с текстами. Является интерпретируемым языком и реализован практически на всех существующих платформах. Применяется при обработке текстов, а также для динамической генерации веб-страниц на веб-серверах.

Хронология (Популярные языки)

Название	Год	Вид	Автор	География создания	Организация	Стандарт
Фортран (FORTRAN)	1954	A	Джон Бэкус	Америка	IBM	ISO 1539:1997
Лисп (LISP)	1958	F	Джон Маккарти	Америка	MIT	-
Алгол-60 (Algol 60)	1960	A	Питер Наур+	Межд.	IFIP	-
Кобол (COBOL)	1960	A	+	Межд.	CODASYL Committee	ISO 1989:1985
Симула (Simula)	1962	B	Кристен Нигаард+	Европа	-	-
Бейсик (BASIC)	1963	A	Джон Кемени+	Америка	Dartmouth College	ISO 10279:1991
ПЛ/1 (PL/1)	1964	A	Джордж Радин	Америка	IBM	ISO 6160:1979
Алгол-68 (Algol 68)	1968	A	Аад ван Вайнгартен+	Межд.	IFIP	-
Паскаль (Pascal)	1970	C	Никлаус Вирт	Европа	ETH	ISO 7185:1990
Форт (FORTH)	1970	A*	Чарльз Мур	Америка	Mohasco Industries	ISO 15145:1997
Си (C)	1972	C*	Деннис Ритчи	Америка	AT&T Bell Labs	ISO 9899:1999
Пролог (Prolog)	1973	E	Алан Кольмеро+	Европа	Univ. of Aix-Marseille	ISO 13211:1995
Ада (Ada)	1980	H*	Джин Ишбиа+	Америка	CII Honeywell	ISO 8652:1995
Си++	1984	H*	Бьорн Страуструп	Америка	AT&T Bell Labs	ISO 14882:1998
Java	1995	H	Джеймс Гослинг	Америка	Sun Labs	-

Спасибо за внимание!